

Tuuli Kankaala

RANSAC-SYLINTERIT PUUNRUNGON MALLINNUKSESSA PISTEPILVIDATASTA

Diplomityö
Tekniikan ja luonnontieteiden tiedekunta
Tarkastajat: Pasi Raunonen
Pietari Mönkkönen
Joulukuu 2023

TIIVISTELMÄ

Tuuli Kankaala: RANSAC-sylinterit puunrunгон mallinnuksessa pistepilvidatasta

Diplomityö

Tampereen yliopisto

Diplomi-insinööri

Joulukuu 2023

Random Sample Consensus eli RANSAC on satunnaisotoksiin perustuva menetelmä mallin sovittamiseksi havaintodataan niin, että datan sisältämien poikkeavien havaintojen vaikutus muodostuvaan malliin tulee minimoiduksi. RANSAC on laajasti sovellettu ja siitä on kehitetty erilaisia versioita, joista tässä työssä on toteutettu MATLAB-ohjelmistolla seitsemän. Toteutettujen RANSAC-versioiden suoriutumista on vertailtu tarkastelemalla niiden tuottamien mallien tarkkuudesta ja niiden suoritustehokkuudesta kertovia mittareita.

Toteutettuja menetelmiä on sovellettu sylinterimallien sovittamiseksi osittaiseen kolmiulotteiseen pistepilvidataan, joka sisältää kohinaa ja poikkeavia havaintoja. Tällainen sovitustilanne on käsillä mallinnettaessa puunrunkoa sarjana peräkkäisiä sylintereitä laserkeilaamalla muodostetusta pistepilvidatasta, sillä olosuhteet luonnossa tuottavat puiden laserkeilausdataan aukkoja ja poikkeavia havaintoja. Työssä on käytetty sekä keinotekoisesti generoitujen yksittäisten sylinterien pistepilviä että keskikokoisesta tammirungosta laserkeilaamalla tuotettua pistepilvidataa, jonka resoluutiota ja käytettyjen laserskannausten määrää on varioitu. Yksittäisiä sylintereitä mallinnettaessa malliparametreille ei asetettu rajoitteita, mutta puunrunkoa mallinnettaessa hyväksyttävien sylinterimallien akselien suuntaa ja säteen suuruutta rajoitettiin.

Kaikki seitsemän RANSAC-versiota soveltivat mediaanitulavuudeltaan alle 10 % poikkeavia runkomalleja runkopistepilveen, jonka resoluutio oli 0,5 datapistettä per neliösenttimetri ja joka sisälsi vain yhden laserskannauksen tuottamat datapisteet. Resoluution nostaminen moninkertaiseksi vähensi tulosten hajontaa mallinnusta toistettaessa. Sen sijaan toisen laserskannauksen lisääminen ensimmäisen rinnalle oletusten vastaisesti heikensi mallinnustuloksia, mikä on voinut johtua joko toisen laserskannauksen tuottaman pistepilven heikommasta laadusta tai kahden skannausten tuottamien pisteiden epäonnistuneesta yhteensovituksesta samaan koordinaatistoon.

MSAC-menetelmä tuotti puunrunkoja mallinnettaessa toteutetuista menetelmistä tarkimpia tuloksia, mutta vaati eniten RANSAC-iteraatiokierroksia. Kt-uudelleensovitusmenetelmän tulokset eivät generoituja yksittäisiä sylintereitä mallinnettaessa olleet kehuttavia, mutta puunrunkoja mallinnettaessa se suoriutui vähintään RANSACin standardiversion tasoisesti ja välillä sitä paremmin. Pienimmän neliösumman korvaaminen pienimmällä neliömediaanilla (LMS) Gauss-Newton-optimoinnin kustannusfunktiona hypoteesimallien sovittamisessa tuotti generoitua sylinteridataa käsiteltäessä paljon hajontaa tuloksiin, mutta runkomalleja muodostettaessa LMS suoriutui kaikilla mittapuilla RANSACin standardiversion kanssa samantasoisesti ja kaikkien toteutettujen menetelmien kesken keskitasoisesti. $T_{d,d}$ -testin käyttö lisäsi puunrunkoja mallinnettaessa käytettyjen iteraatioiden määrää ja tulosten hajontaa. Inner-RANSAC-uudelleensovitusmenetelmä ja sen yhdistäminen Kt-uudelleensovitusmenetelmään eivät vaikuttaneet parantavan tuloksia tai vähentävän tarvittavien iteraatioiden määrää tarpeeksi kompensoidakseen näiden menetelmien vaatimista ylimääräisistä mallisovitusoperaatioista koituvia kustannuksia laskenta-ajassa.

Avainsanat: RANSAC, sylinteri, puunrunko, pistepilvi, matemaattinen mallinnus

Tämän julkaisun alkuperäisyys on tarkastettu Turnitin OriginalityCheck -ohjelmalla.

ABSTRACT

Tuuli Kankaala: Modelling tree trunks from point cloud data as RANSAC cylinders
Master's thesis
Tampere University
Master of Science (Engineering)
December 2023

RANSAC (Random Sample Consensus) is a method for fitting a model into observational data in the presence of possible measuring errors, called outliers, and measuring noise. The goal of its use is to minimise the effect of the outliers on the fitted model. RANSAC is widely applicable and has been modified into many different versions, of which seven are implemented in this thesis work using MATLAB software. The performance of the seven versions of RANSAC is studied by analysing markers of the fit of the produced models and of the required computational costs.

The implemented RANSAC methods have been applied to fitting cylinder models into three-dimensional point cloud data, which covers only a part of the cylinder surface and contains outliers and noise. This kind of modelling situation is relevant when modelling a tree trunk as a series of subsequent cylinders from LiDAR (Light Detection and Ranging) point cloud data, since environmental factors often cause formation of gaps and outlier clusters in the point clouds produced from trees. Both artificially generated cylinder point cloud data and real LiDAR point cloud data from a medium-sized oak trunk were used in this thesis. The density and the number of scanning directions used in the LiDAR point clouds were varied. Restrictions on model parameters were not implemented when modelling single cylinders from the artificially generated data, but during the tree modelling phase limits were set for an acceptable cylinder radius and cylinder axis angle.

The median volumes of the fitted tree trunk models deviated less than 10 % for all seven implemented RANSAC-versions, when a point cloud from only one scan with the density of 0,5 points per square centimeter was used. Multiplying the point density reduced the dispersion of the results, when models were regenerated multiple times. Contrary to expectations, adding points from a second laser scan from a different direction weakened the modelling results, which may have been a consequence of the second point cloud being of inferior quality or an unsuccessful registration of two LiDAR scans to the same coordinate system.

Of the seven methods, MSAC produced the most accurate tree trunk volumes but required the most RANSAC-iterations. Refitting the model iteratively by scaling the used error tolerance did not impress with the results when processing the generated cylinder point clouds, but with the LiDAR data the method slightly outperformed the standard RANSAC. Replacing least sum of squares with least median of squares (LMS) as the cost function of Gauss-Newton optimisation heightened dispersion of the results when fitting singular cylinders to the generated point cloud data. However, for the tree trunk LiDAR point clouds this was not the case and LMS-modified RANSAC performed at the same level with the standard version. The use of $T_{d,d}$ test increased the dispersion of the results and the amount of used iterations when modelling tree trunks. Refitting tree models with inner-RANSAC and combining inner-RANSAC with iterative rescaled error tolerance did not appear to improve the results or to reduce the amount of iterations enough to compensate for the time consuming additional model refitting operations that these methods perform.

Keywords: RANSAC, cylinder, tree trunk, point cloud, mathematical modelling

The originality of this thesis has been checked using the Turnitin OriginalityCheck service.

ALKUSANAT

Olen opiskeluni aikana saanut paljon apua ja tavannut yliopistossa eri yhteyksissä ison joukon kiinnostavia ihmisiä, joiden osaaminen, persoona tai molemmat ovat tehneet vaikutuksen. Kiitokset heille kaikille näistä vuosista ja kokemuksista. Haluan esittää erityiskiitokset ohjaajalleni Pasi Raumoselle, jonka matemaattinen ammattitaito ja ystävällinen opastus ovat tämän työn kivijalka, sekä työn toiselle tarkastajalle Pietari Mönkköselle, jonka hyödylliset ja kannustavat kommentit olivat todella tervetulleita. Erityis-erityiskiitokset kuuluvat perheelleni ja ystäväilleni, joista jokainen toivottavasti tiedostaa korvaamattomuutensa elämässäni. Erityis-erityis-erityiskiitokset Huugolle ja Vilpulle, jotka jättäessäni läppärini ajamaan koodia itsekseen valvoivat prosessia tarkkaavaisesti. Suojasin läppärin pyykkikorilla heidän läsnäollessaan, mutta ihan vain muista sattumanvaraisista syistä, ei sillä ettenkö olisi luottanut pieniin pörröisiin prosessinvalvojiin.

Tampereella, 7. joulukuuta 2023

Tuuli Kankaala



Kuva 1. Prosessi käynnissä ja prosessinvalvoja paikalla.

SISÄLLYSLUETTELO

1.	Johdanto	1
2.	Sovellusongelma.	3
2.1	Puunrunko mallinnuksen kohteena	3
2.2	Puunrungon pistepilven tuottaminen laserkeilaamalla	4
3.	Kirjallisuuskatsaus RANSAC-menetelmiin	6
3.1	Muita lähestymistapoja	6
3.2	RANSAC	7
3.2.1	Satunnaisotokset	11
3.2.2	Kustannusfunktio	13
3.2.3	Konsensusjoukon muodostaminen	15
3.2.4	Virhetoleranssitaso	16
3.2.5	Iteraatioiden määrä ja lopetuskriteerit	17
3.2.6	Mallin sovittaminen ja uudelleensovittaminen	21
4.	Menetelmät	23
4.1	Sovelletut RANSAC-menetelmät.	23
4.1.1	RANSAC	23
4.1.2	MSAC	24
4.1.3	$T_{d,d}$ -testi	24
4.1.4	Inner-RANSAC, Kt-menetelmä ja niiden yhdistelmä	24
4.1.5	LMS	25
4.2	Sylinterien parametrusointi	25
4.3	Menetelmien toteutus MATLAB-ohjelmistolla	25
4.4	Menetelmien arviointiin käytetty data	30
4.5	Arvioitavat muuttujat	34
5.	Tulokset	36
5.1	Menetelmien suoriutuminen generoidulla sylinteridatalla	36
5.1.1	RANSAC-versioiden tuottamat sylinteriparametrit	37
5.1.2	RANSAC-tunnusluvut sylinteridatalla	41
5.2	RANSAC-versiot puunrungon pistepilvidatalla	45
5.2.1	Menetelmien suoriutuminen yhden laserskannauksen tuottamasta pistepilvestä	46
5.2.2	Menetelmien suoriutuminen kahden laserskannauksen tuottamasta pistepilvestä	50
6.	Päätelmät ja yhteenveto	55

Lähteet 60

LYHENTEET JA MERKINNÄT

mad	engl. mean absolute distance, pistepilven pisteiden $d \in D$ keskiarvoetäisyys sylinterimallin M_i pintaan
\vec{a}	sylinterin akselin suuntavektori, kolmiulotteinen yksikkövektori
an	anomalioiden lukumäärä, edellisestä sylinteristä tai sylinterisovituksesta huomattavasti poikkeavien sylinterien lukumäärä mallinnettaessa puunrunkoa RANSAC-sylintereinä
C_i	i . RANSAC-hypoteesimallin konsensusjoukko
C_{max}	suurin RANSAC-hypoteesimallien konsensusjoukoista
D	RANSAC-menetelmässä käytettävä havaintodatajoukko, johon halutaan sovittaa malli
ϵ_e	arvio hyvien havaintopisteiden $d \in H$ osuudesta havaintodatajoukossa D
ϵ	hyvien havaintopisteiden $d \in H$ osuus havaintodatajoukossa D
F	kustannusfunktion arvo
H	hyvien havaintopisteiden eli mallin mukaisten havaintopisteiden joukko, joukon D osajoukko
k	RANSAC-iteraatiokierrosten eli satunnaisotosten S_i lukumäärä
l	sylinterin pituus
LMS	pienimmän neliömediaanin menetelmä (engl. Least Median of Squares)
M_i	i . RANSAC-satunnaisotokseen S_i sovitettu hypoteesimalli
M_{max}	suurimman RANSAC-konsensusjoukon C_{max} tukema hypoteesimalli
m	RANSAC-satunnaisotosten S_i otoskoko
MLESAC	Maximum Likelihood Estimator Sample Consensus -menetelmä
MSAC	M-estimator Sample Consensus -menetelmä
p_a	sylinterin akselin \vec{a} piste sylinterin pohjassa
P	poikkeavien havaintopisteiden joukko, joukon D osajoukko
r	sylinterin säde

RANSAC	Random Sample Consensus -menetelmä
S_{max}	RANSAC-satunnaisotos S_i , johon sovitettu hypoteesimalli M_i on suurimman konsensusjoukon C_{max} tukema malli M_{max}
S_i	i . RANSAC-satunnaisotos, joukon D osajoukko
V	tilavuus

1. JOHDANTO

Eräs insinööritieteissä yleinen ongelma on mittausdataa kuvaavan mallin sovittaminen dataan. Ongelma on sitä vaikeampi, mitä enemmän epäpuhtauksia eli kohinaa ja virheellisiä mittapisteitä data sisältää. Jos lisäksi mittausdataa on saatavilla mittauskohteen matkalta vain osittain, jolloin mallin muodostaminen vaatii ekstrapolointia, vaaditaan sovitusten menetelmältä paljon. RANSAC on suosittu ja monikäyttöinen satunnaisotoksiin perustuva mallien sovitusten menetelmä, jonka on tarkoitus vastata edellä kuvattuihin haasteisiin. RANSACilla on mallinnusmenetelmänä sekä hyviä että huonoja puolia, joiden pohjalta siitä on kehitetty monia erilaisia versioita.

Tässä työssä RANSAC-menetelmästä toteutetaan seitsemän erilaista helppokäyttöistä versiota, joiden suoriutumista vertaillaan. Sovelluskohteena on sylinteripintojen mallintaminen kolmiulotteisesta pistepilvidatasta, ja menetelmien testaamiseen käytetään sekä simuloituja sylinteripistepilviä että puunrungoista laserkeilaamalla tuotettua pistepilvidataa. Puunrunkojen pistepilvistä pyritään mallintamaan kokonaisia puunrunkoja approksimoimalla runkoa sarjana peräkkäisiä sylindereitä. Sekä generoitu sylinteridata että aito puunrunkojen laserkeilausdata kuvaavat osittaisia sylinteripintoja ja sisältävät kohinaa sekä poikkeavia havaintoja, minkä on tarkoitus heijastaa puiden laserkeilaukseen liittyviä haasteita.

Seitsemää toteutettua RANSAC-versiota arvioidaan sekä niiden muodostamien mallien laadun että suoritusajasta kertovien muuttujien näkökulmasta. Mallien laatua arvioidaan tarkastelemalla, kuinka lähellä todellisia arvoja menetelmien tuottamat sylinterimallien parametrit ja puunrunkomallien tilavuudet ovat. Suoritusajasta kertoo menetelmien käyttämien RANSAC-satunnaisotosten määrän lisäksi se, millä kaikilla tavoilla menetelmät käsittelevät kutakin satunnaisotosta. Puunrunkon mallintamisessa tutkitaan sitä, millaisia puunrunkomalleja pelkän yhden laserskannauksen tuottamasta pistepilvestä saadaan ja miten toisen laserskannauksen lisääminen ensimmäisen rinnalle vaikuttaa tuloksiin. Lisäksi tarkastellaan pistepilven resoluution nostamisen vaikutusta syntyviin puunrunkomalleihin. Oletuksena havaintodatamäärän kasvattaminen, eli sekä resoluution nostaminen että toisen laserskannauksen tuottamien datapisteiden lisääminen pistepilveen, parantaa mallinnuksen tuloksia. Työn tavoitteet voidaan muotoilla kahdeksi tutkimuskysymykseksi:

- Miten toteutetut RANSAC-versiot eroavat toisistaan tehokkuudessa ja tuotettujen mallien laadussa?
- Miten puunrungosta laserkeilaamalla tuotetun pistepilven resoluution nostaminen ja toisaalta kahden laserskannauksen suorittaminen yhden sijasta vaikuttaa mallinustuloksiin?

Ensimmäiseksi työssä kuvaillaan käsillä oleva puiden laserkeilauksen sovellusongelma luvussa 2. Tämän jälkeen luvussa 3 kerrotaan tarkemmin RANSACin toiminnasta ja sen hyvistä ja huonoista puolista sekä joistakin keskeisistä tavoista, joilla RANSACia on kehitetty eteenpäin sen erilaisiksi versioiksi. Luku 4 käsittelee ne versiot, jotka on valittu käytettäväksi tässä työssä ja kuvailee, miten ne on toteutettu. Luvussa 4 myös kuvaillaan tarkemmin menetelmien testaamiseen käytetty data. Sovellettujen menetelmien tuottamat tulokset esitellään luvussa 5, ja lopuksi työn tuloksia kokonaisuutena pohditaan vielä luvussa 6.

2. SOVELLUSONGELMA

Työn keskiössä ovat RANSAC-menetelmät, ja näkökulmana on niiden soveltaminen puunrunkon kolmiulotteisen pinnan mallintamiseen. Tässä luvussa määritellään lyhyesti, miten puunrunkoa tässä työssä tarkastellaan ja millaista on laserkeilausdata, johon puunrunkon malli pohjautuu.

2.1 Puunrunko mallinnuksen kohteena

Työssä mallinnuksen kohteena on yksittäisen puun runko. Puun oksat ja lehvistö jätetään mallinnuksen ulkopuolelle, ja rungon mallinnusdataan päätyvät oksien ja lehtien osat luokitellaan mallinnuksessa häiriötekijöiksi. Puunrunkon mallintamisen tavoitteena on saada selville rungon säde ja tilavuus siltä matkalta, jolta rungon päähaara voidaan määrittää ennen sen jakautumista puun yläosassa latvushaaroiksi.

Puiden mallintamisen haasteena on, että puiden geometriset ominaisuudet voivat olla lajista ja ulkoisista kasvutekijöistä riippuen hyvin moninaiset [1]. Tässä työssä mallinnuksen kohteena olevalle puunrungolle on asetettu joitakin reunaehtoja, joiden avulla pyritään saavuttamaan mielekkäämpiä mallinnustuloksia. Ensiksi, puunrunkoa mallinnetaan työssä sarjana peräkkäisiä sylintereitä, joiden tunnuslukuja ovat sylinterin säde, pituus ja akselin suunta. Sylinterit ovat poikkileikkaukseltaan ympyröitä, jolloin puunrunkon poikkileikkaus oletetaan ympyräksi. Toinen työssä tehtävä oletus koskee mallinnettavan puun kasvusuuntaa, jossa ei oleteta tapahtuvan huomattavan suuria yhtäkkisiä muutoksia puun eri osien välillä. Runkomallin muodostamisessa sovitettaville sylinterille asetetaan rajoitteita, joilla pyritään vähentämään sellaisia sylinterimalleja, joiden akselin suunta poikkeaa merkittävästi sitä edeltävän sylinterin akselin suunnasta tai joiden säteen suuruus poikkeaa merkittävästi edeltävän sylinterin säteen suuruudesta. Runkomallien muodostamisen jälkeen näitä rajoitteita rikkovat sylinterisovitukset oletetaan lähtökohtaisesti epäonnistuneiksi.

Todellisuudessa puunrunkon poikkileikkauksen tarkka muoto saattaa vaihdella eri puiden ja yksittäisen puunrunkon osien välillä, ja puunrunkon kasvusuunnassa saattaa myös esiintyä huomattavia mutkia tai kulmia. Puunrunkomalli voi sopeutua paremmin kasvusuunnan muutokseen, jos mallinnettavien sylinterien pituuden annetaan määräytyä joustavasti sen mukaan kuinka pitkiä samansuuntaisia kappaleita mallinnusmenetelmä löytää run-

gosta. Tässä työssä kaikesta tarkastellusta datasta pyritään kuitenkin mallintamaan va-
kiopituisia sylintereitä. Tämän johdosta poikkeaviksi luokiteltuja sylintereitä ei karsita run-
gon muodostamisen yhteydessä pois, mutta niiden lukumäärää kuitenkin tarkastellaan
runkomallinnuksen tulosten analysoinnin yhteydessä. Parempiin mallinnustuloksiin pyrit-
täessä tehdyt oletukset ja niiden pohjalta asetetut rajoitteet saattavat heikentää mallin-
nusmenetelmän yleistettävyyttä ja rajata sovelluskohteiden ulkopuolelle esimerkiksi geo-
metrisesti monimuotoista puustoa, jossa on paljon pahkoja, käyriä runkoja ja muita poik-
keamia.

2.2 Puunrunгон pistepilven tuottaminen laserkeilaamalla

Laserkeilaus on epäinvasiivinen mittaussuunnitelma, joka mahdollistaa objektien kuvanta-
misen kolmiulotteisena pistepilvenä. Käytetystä mittausrakenteesta ja mittaussuhteis-
ta riippuen laserkeilattujen objektien geometrisia ominaisuuksia pystytään tallentamaan
jopa millimetrien tarkkuudella. Maanpinnalla toteutettua laserkeilausta (engl. terrestrial la-
ser scanning, lyh. TLS, myös Terrestrial Light Detection and Ranging, lyh. TLiDAR [1])
on käytetty muiden sovellusten, kuten esimerkiksi rakennustekniikan, ohella sekä yksit-
täisten puiden että puustojen kuvantamiseen, ja tekniikan suosio on 2010-luvun aikana
lisääntynyt. [2]

Laserkeilauksen haasteina voidaan pitää muun muassa laserskannerin ja kuvattavan koh-
teen väliin työntyviä objekteja, jotka voivat tuottaa ylimääräistä mittaustietoa ja vaikeut-
taa mittaustietojen saamista tosiasiallisesta kuvauksen kohteesta tai vaikeuttaa siitä saa-
dun mittaustietojen tunnistamista. Laserkeilaamalla kuvattuja puita ympäröivä muu kas-
vusto vaikeuttaa usein tällä tavalla puiden kuvaamista. Seurauksena esimerkiksi yksittäi-
sen puun laserkeilaustietojen voi olla aukkoja, jos viereisestä puusta kasvava oksa on
työntynyt mittaustilanteessa lasersignaalin eteen, tai puunrungon alaosa voi olla vaikea
mallintaa, jos se on aluskasvillisuuden peitossa. Koska laserkeilaus tuottaa tietoa vain
laserskannerille näkyvästä osasta kuvauksen kohdetta, on puu myös kuvattava useasta
suunnasta sen ympäriltä, jos se halutaan mallintaa joka puolelta. Useasta eri suunnasta
kuvaaminen ja syntyneiden pistepilvien yhdistäminen auttaa täyttämään yhdessä suun-
nassa eteen työntyneiden objektien, kuten viereisten puiden oksien, synnyttämiä aukkoja
kuvaustietojen saamisessa. Puuta ympäröivän muun kasvuston vaikutuksen lisäksi kuvausta voi han-
kaloittaa myös puun oksia liikutteleva tuuli. [1]

Lasersignaalin voimakkuus eri mittauspisteissä voi vaihdella, ja signaalien voimakkuutta
voidaan käyttää hyödyksi kohteen tutkimisessa. Tämä vaatii kuitenkin tietoa mittalaitteen
ja mittauksen kohteen etäisyydestä ja mittauskulmista. [3] Laserskannerin ja mittauksen
kohteen etäisyyden perusteella voidaan myös arvioida lasersäteiden halkaisijoita ja näin
ollen mittaustietojen kohinan tasoa. Tässä työssä signaalien voimakkuutta, laite- tai mittaussu-
hteita ei tunneta eikä käsitellä, vaan mallintamisessa hyödynnetään pelkästään

laserkeilausdatan geometrisiä ominaisuuksia eli mittauspisteiden koordinaatteja kolmiulotteisessa avaruudessa.

Työssä sovellettavia menetelmiä analysoidaan sekä simuloidun sylinteridatan että keskikokoisesta tammirungosta saadun laserkeilausdatan avulla. Sekä laserkeilausdata että generoidut sylinterien pistepilvet pyrkivät kuvastamaan puiden laserkeilauksen haasteita osittaisen näkyvyyden ja rungon pistepilvidatan seassa olevien oksankappaleista peräisin olevien mittapisteiden avulla. Aidosta puunrungosta laserkeilaamalla saatu data on esikäsitelty segmentoimalla pois suurin osa oksista, mutta siinä on jäljellä jonkin verran rungoista lähtevien oksien tynkiä sekä muiden oksien kappaleita, jotka työssä käytettävien mallinnusmenetelmien on tarkoitus identifioida poikkeaviksi havainnoiksi ja jättää huomiotta runkomallin muodostamisessa. Datasta kerrotaan tarkemmin luvussa 4.4.

3. KIRJALLISUUSKATSAUS RANSAC-MENETELMIIN

Tässä kappaleessa esitellään joitakin menetelmiä pintamallin sovittamiseksi kolmiulotteiseen pistepilvidataan tilanteessa, jossa mittauksen tulokset sisältävät kohinaa ja aukkoja sekä poikkeavia havaintoja. Kaikki käsitellyt menetelmät pyrkivät löytämään malliparametrit, jotka tuottavat havaintodataan parhaiten sopivan mallin havaintodatan epäpuhtauksista huolimatta. RANSAC-menetelmän perusmekanismin esittelyn lisäksi tuodaan esiin joitakin sen variaatioita. Ensimmäiseksi kerrotaan kuitenkin lyhyesti joistakin vaihtoehtoisista lähestymistavoista, sillä vaikka RANSAC-viitekehys on suosittu, se ei kuitenkaan ole ainoa menetelmä, jolla työssä käsiteltävän kaltaisia ongelmia on pyritty ratkomaan.

3.1 Muita lähestymistapoja

RANSAC-viitekehyyksen lisäksi työssä käsitellyn sovellusongelman ratkaisuun on olemassa useita muitakin lähestymistapoja. Alkuperäinen motiivi RANSACin kehittämiseksi 80-luvulla oli poikkeavien havaintojen paremman sietokyvyn tavoittelu lineaariseen regressioon, kuten pienimmän neliösumman menetelmään, verrattuna [4]. Perinteistä pienimmän neliösumman regressiomenetelmää on pyritty tässä suhteessa parantelemaan tilastotieteessä esimerkiksi totaalisen pienimmän neliösumman (lyh. TLS, engl. total least squares) [5] ja osittaisen pienimmän neliösumman (lyh. PLS, engl. partial least squares) [6] menetelmillä sekä korvaamalla neliösumma neliömediaanilla, jolloin kyseessä on pienimmän neliömediaanin menetelmä (lyh. LMS tai LMEDS, engl. least median of squares) [7]. Toisin kuin RANSACin kohdalla [8][9], LMS-menetelmän sietämä poikkeavien havaintojen osuus on rajoitettu ja on suurimmillaan 50 % havaintodatasta [7], mutta Fangin ym. mukaan [5] sen total least median of squares -variaatio on kilpailukykyinen RANSACiin verrattuna. Myös RANSACin kanssa samoihin aikoihin kehitetyn M-estimaattorin poikkeavien havaintojen sietokyky on LMS-menetelmän tavoin RANSACia huonompi [7]. Tässä työssä sekä LMS-menetelmää että M-estimaatiota käytetään yhdessä RANSACin kanssa, kun RANSACin perinteinen pienimmän neliösumman sovituskäytännön korvata LMS-funktiolla (ks. luku 3.2.6) ja RANSACin kustannusfunktiosta modifioidaan MSAC-kustannusfunktio (ks. luku 3.2.2).

Pienimmän neliösumman ja sen variaatioiden ohella mallien muodostamiseen pistepilvi-

datasta on käytetty esimerkiksi klusterianalyysiä (engl. clustering tai cluster analysis) ja pääkomponenttianalyysiä (lyh. PCA, engl. principle component analysis), myös puunrun-
kojen osien tunnistamiseen ja niiden suunnan määrittämiseen [10]. Hough-muunnos on
RANSAC-menetelmien rinnalla erityisesti konenäön sovellusalalla usein käytetty ja mai-
nittu menetelmä kaksi- ja kolmiulotteisten objektien tunnistamiseen. Schnabelin ym. mu-
kaan sitä kuitenkin käytetään suuren muistinkulutuksen vuoksi useimmiten kaksiulotteis-
ten mallien muodostamiseen kompleksisempien kolmiulotteisten mallien sijaan. [8] Ragu-
ram ym. esittelevät vuoden 2013 artikkelissaan [9] ja Schnabel ym. artikkelissaan vuodel-
ta 2007 [8] useita erityisesti konenäön osa-alueella sovellettuja menetelmiä, jotka tarjoa-
vat vaihtoehdon RANSACille kukin omine vahvuuksineen ja heikkouksineen.

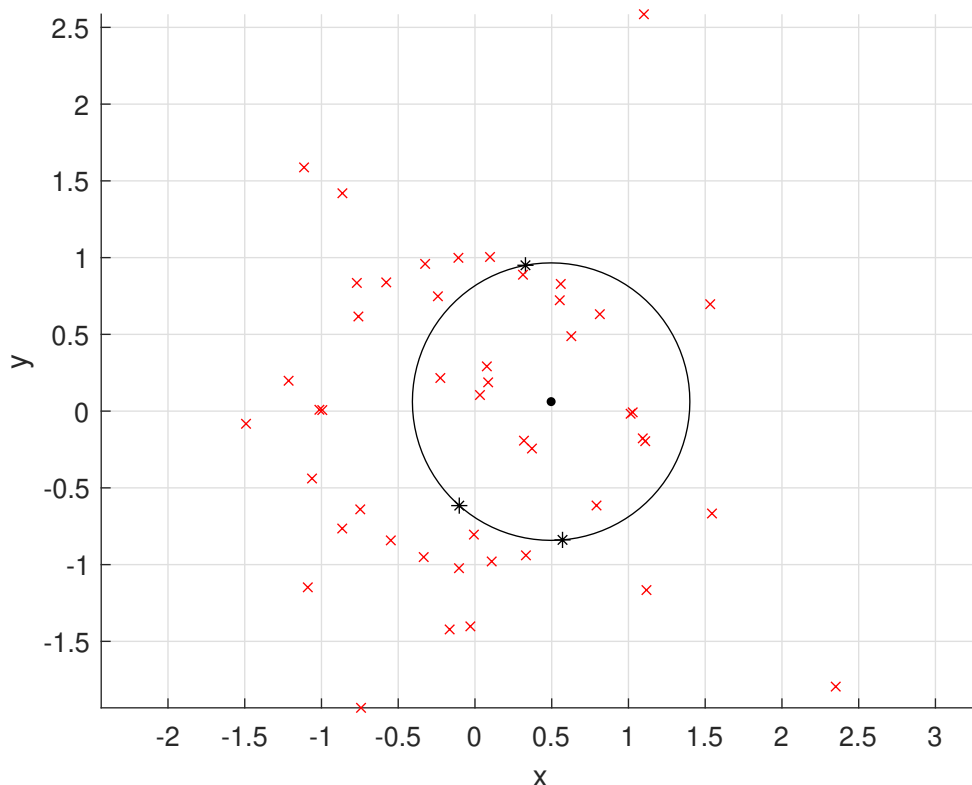
3.2 RANSAC

RANSAC (engl. Random Sample Consensus) on iteratiivinen menetelmä, jossa k ite-
raatiokierroksen ajan jokaisella i . iteraatiokierroksella havaintodatajoukosta D poimitaan
satunnaisotos $S_i \subset D$. Satunnaisotokseen S_i sovitetaan hypoteesimalli M_i , minkä jäl-
keen lasketaan malliin sopivien eli mallia tukevien havaintopisteiden d määrä koko mit-
tausdatasta D . Havaintopisteen d katsotaan tukevan mallia M_i , jos sen etäisyys malliin
on korkeintaan ennalta määritellyn virhetoleranssitason t verran. Mallia tukevien pisteiden
joukkoa kutsutaan mallin konsensusjoukoksi C_i (engl. consensus set) ja mallia tukevien
pisteiden määrä on yhtä kuin tämän joukon kardinaliteetti $|C_i|$. Iteroinnin päätteeksi paras
hypoteesimalli M_i on se, jonka konsensusjoukko C_i on suurin eli jolle $|C_i| = \max\{|C_j| \mid 1 \leq j \leq k\}$. Tällaisella mallilla katsotaan olevan suurin havaintodatan tuki. [4] Tässä
työssä suurimmasta konsensusjoukosta käytetään jatkossa merkintää C_{max} , sen tuke-
masta eli parhaasta hypoteesimallista merkintää M_{max} ja tämän parhaan mallisovituksen
pohjalla olleesta satunnaisotoksesta merkintää S_{max} .

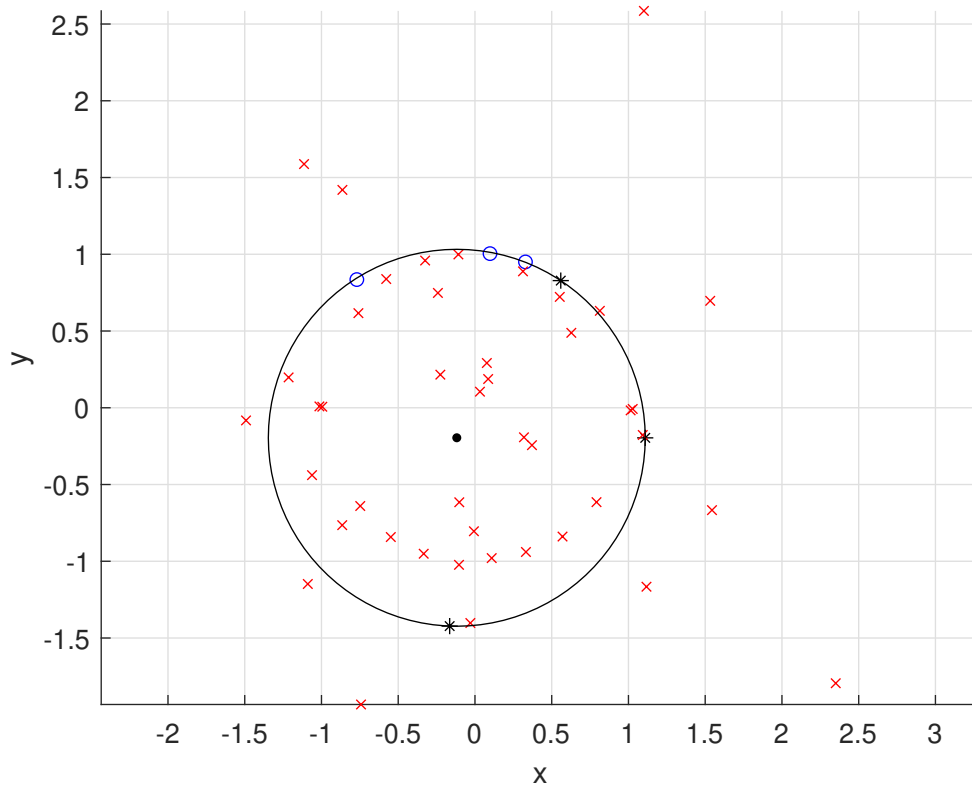
Menetelmän tuloksena havaintodatajoukko D tulee luokiteltua malliin M_{max} sopiviin ha-
vaintoihin (engl. inlier) ja mallista poikkeaviin havaintoihin (engl. outlier) [11]. RANSAC-
menetelmän tuottaman parhaan mallin M_{max} oletetaan lähtökohtaisesti kuvaavan hyvin
sitä datajoukkoa D , jota se mallintaa, ja malliin M_{max} sopivien mittapisteiden oletetaan
olevan hyviä eli kohinaa lukuunottamatta ei-virheellisiä ja peräisin todellisesta mittauk-
sen kohteesta. Näistä malliin M_{max} sopivia mittapisteitä kutsutaan tässä työssä hyväiksi
havainnoiksi, ja niiden joukosta käytetään merkintää H . Mallista M_{max} vähintään virheto-
leranssitason t verran poikkeavat havainnot oletetaan mittavirheiksi, jotka voivat olla pe-
räisin jostain muusta kuin tarkoitetusta mittauksen kohteesta. Tästä poikkeavien havain-
tojen joukosta käytetään merkintää P . Sekä hyvien että poikkeavien havaintojen joukot
ovat havaintodatajoukon D aitoja osajoukkoja, eli $H \subset D$ ja $P \subset D$. Usein ollaan kiin-
nostuneita hyvien havaintojen $d \in H$ osuudesta havaintodatajoukossa D , ja tätä osuutta
merkitään yleisesti kirjallisuudessa ja myös tässä työssä epsilonilla ϵ .

Esimerkki 3.1. Havainnollistetaan RANSAC-käsitteitä sovittamalla ympyrä kaksiulotteiseen pistepilvidataan. Esimerkkiä varten on generoitu datajoukko D , jossa on 50 koordinaattipistettä. Niistä 20 kappaletta mukailee kutakuinkin origokeskeisen, 1-säteisen ympyrän kaarta, mutta niihin on lisätty normaalijakautunutta kohinaa. Tämä on hyvien havaintojen joukko H , ja hyvien havaintojen osuus ϵ datassa D on siis 0,4. Pisteistä 30 kappaletta jakautuu satunnaisesti ympyrän sisälle ja sen ympärille, ja näiden mittapisteiden joukko on datan poikkeavien havaintojen joukko P .

Kuvassa 3.1 ja kuvassa 3.2 on kaksi esimerkkiä meneillään olevasta RANSAC-iteroinnista. Molemmissa kuvissa datasta D on poimittu $m = 3$ pisteen kokoinen satunnaisotos S_i , satunnaisotokseen on sovitettu hypoteesimalli M_i ja mallille on muodostettu konsensusjoukko C_i virhetoleranssitasolla $t = 0,015$. Otosten S_i pisteet on merkitty kuviin mustilla tähdillä, konsensusjoukkojen C_i pisteet sinisillä ympyröillä ja muut pisteet punaisilla rasteilla. Hypoteesimalliympyrän M_i käyrä näkyy kuvissa mustana ympyränä ja sen keskipiste mustana pisteenä. Kummankaan kuvan hypoteesimalli ei selvästi ole etsitty yksiköympyrä. Kuvan 3.1 hypoteesimallille ei ole löytynyt yhtään konsensusjoukon pistettä, ja kuvan 3.2 hypoteesimallille niitä on löytynyt 3 kappaletta.



Kuva 3.1. RANSAC-iterointi on kesken. Pistepilvestä on poimittu kolmen pisteen suuruinen RANSAC-otos, johon on sovitettu hypoteesimallina M_i ympyrän kaari. Konsensusjoukon C_i koko on 0.



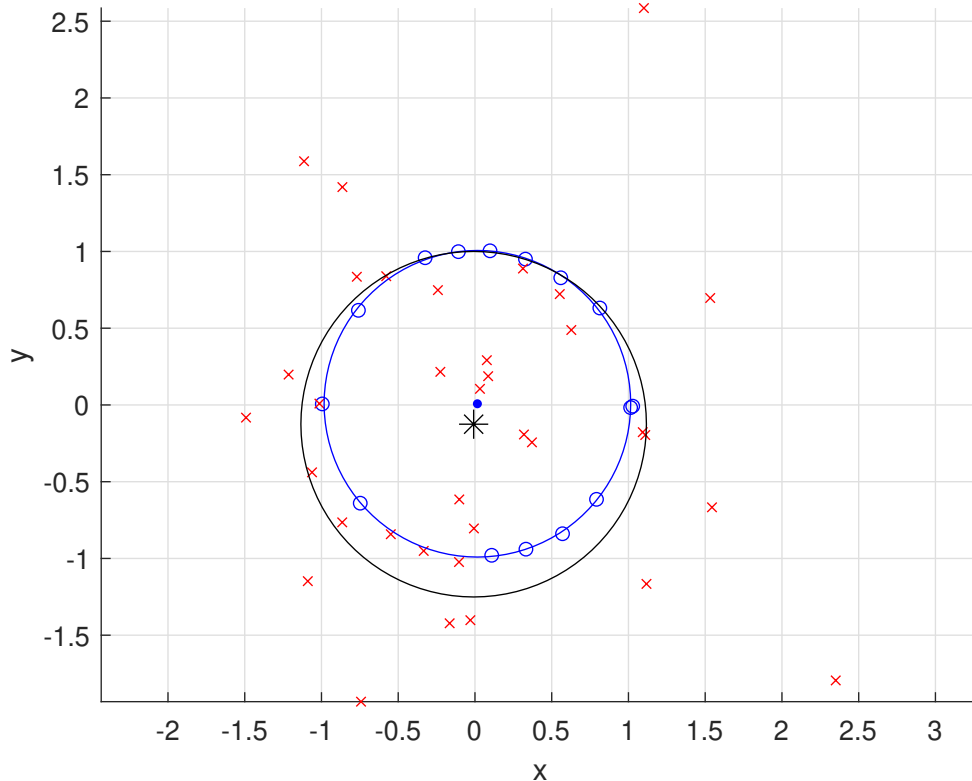
Kuva 3.2. Toinen esimerkki meneillään olevasta RANSAC-iteroinnista. Kolmen pisteen suuruiseen RANSAC-otokseen on sovitettu hypoteesimallina M_i ympyrän kaari, jonka konsensusjoukon C_i koko on 3.

RANSAC esiteltiin vuonna 1981 parannuksena siihen asti yleisesti käytetyille sovitusyökaluille, kuten pienimmän neliösumman menetelmälle, jotka sietivät huonosti poikkeavia havaintoja. Näissä aiemmissä menetelmissä tyypillisesti sovittiin ensin malli M koko havaintodataan D , minkä jälkeen mallin kanssa yhteensopimattomat havainnot luokiteltiin poikkeaviksi ja poistettiin. Tämän jälkeen mallia voitiin hienosäätää sovittamalla se uudelleen jäljelle jääneeseen havaintodataan $D \setminus P$, joka oletettiin laadukkaaksi poikkeavien havaintojen karsiuduttua pois. Jo suhteellisen pieni määrä poikkeavia havaintoja alkuperäisessä havaintodatajoukossa D saattoi kuitenkin vääristää tällaisilla menetelmillä saatavia tuloksia. [4]

Esimerkki 3.2. Esimerkin 3.1 pistepilvidataan on kuvassa 3.3 sovitettu ympyrämallit sekä RANSACilla että pelkästään pienimmän neliösumman menetelmää käyttämällä. RANSACin tuottaman mallin M_{max} ympyräkaari on kuvassa sininen ja sen keskipiste on merkattu kuvaan sinisellä pisteellä. Keskipiste $(0,02; 0,01)$ on lähes täsmälleen origossa, ja ympyrän säde r on $0,99$. Pistepilven D kuvatessa yksikköympyrää on RANSAC-sovitus siis hyvin onnistunut. RANSAC-sovitetun ympyrän kaarelle on merkitty näkyviin sinisillä palloilla RANSACin parhaalle mallille M_{max} tuottaman konsensusjoukon C_{max} pisteet, joita on 15 kappaletta. Kun datajoukon D suuruus on 50, on RANSACin löytämä hyvien

havaintojen osuus ϵ tällöin 0,3, mikä on vähemmän kuin esimerkissä 3.1 ilmoitettu todellinen osuus 0,4. RANSAC ei siis aina löydä kaikkia hyviä havaintopisteitä.

Pienimmän neliösumman menetelmän tuottaman ympyrän kaari on piirretty kuvaan 3.3 mustalla. Sen keskipiste on merkitty kuvaan mustalla asteriskilla. Keskipiste $(-0,01; -0,13)$ poikkeaa silminnähden origosta ja ympyrän säde $r = 1,13$ on myös selvästi suurempi kuin 1, joten poikkeavat havainnot ovat vääristäneet sovituksen tuloksia.



Kuva 3.3. RANSAC-sovitetun pienemmän ympyrän kaari näkyy kuvassa sinisenä ja pienimmän neliösumman menetelmällä sovitetun suuremman ympyrän kaari mustana.

RANSAC:in lähestymistapa sen sijaan on sovittaa hypoteesimallit M_i mahdollisimman pieniin satunnaisotoksiin, jolloin riski poikkeavien havaintojen sisällyttämiseen pienenee huomattavasti. Tarvittavalla iteraatiomäärällä k voidaan taata tietyllä luottamustasolla η , että suurimman konsensusjoukon C_{max} löytyessä sen pohjana ollut satunnaisotos S_{max} ei sisältänyt poikkeavia havaintoja. Poikkeavia havaintoja sisältävät satunnaisotokset S_i tuottavat virheellisiä hypoteesimalleja M_i , joille muun havaintodatan tuki on heikkoa ja konsensusjoukko C_i jää pieneksi. [4] RANSAC-menetelmän murtumispisteen (engl. breakdown point) katsotaankin olevan hyvä: RANSACin katsotaan tuottavan mielekkäitä tuloksia, vaikka jopa yli puolet havaintodatasta koostuisi poikkeavista havainnoista eli vaikka hyvien havaintojen osuus ϵ koko havaintodatasta D olisi alle 0,5 [8].

Sen lisäksi, että RANSAC on sovitusmenetelmänä usein käyttökelpoinen haastavienkin

aineistojen kanssa työskenneltäessä, sen toimintaperiaate on helppo ymmärtää ja se on myös laajasti sovellettavissa erilaisiin käyttökohteisiin [8]. Muun muassa näistä syistä se on paljon käytetty erityisesti konenäön ja tilastotieteen aloilla [9]. Sen alkuperäinen versio on kuitenkin laskennallisesti raskaskäyttöinen [9]. Kuten luvussa 3.2.5 nähdään, alkuperäisen RANSAC-algoritmin vaatima laskentakapasiteetti kasvaa eksponentiaalisesti poikkeavien havaintojen $d \in P$ osuuden kasvun myötä havaintodatassa D eli hyvien havaintojen osuuden ϵ laskiessa [11]. Muun muassa näiden heikkouksien ja vahvuuksien pohjalta RANSAC-menetelmää on pyritty kehittämään edelleen. Konkreettisten kehitysideiden pohjalla on ollut muun muassa seuraavia kysymyksiä:

- Miten satunnaisotokset muodostetaan?
- Miten hypoteesimallit sovitetaan dataan?
- Miten konsensusjoukko muodostetaan, eli mikä on hypoteesimallia tukevan ja siitä poikkeavan havaintopisteen välinen raja?
- Onko konsensusjoukon suuruus ainoa mahdollinen kustannusfunktio, vai voiko maljeja vertailla keskenään jollakin muulla perusteella?
- Milloin iterointi voidaan lopettaa?

Näitä RANSAC-menetelmän aspekteja käydään seuraavaksi läpi omissa aliluvuissaan. Luvuissa esitellään joitain muunnelmia, joita alkuperäisestä RANSAC-algoritmista on kehitetty sen eri osa-alueiden optimoimiseksi ja tulosten laadun, algoritmin tehokkuuden tai molempien parantamiseksi.

3.2.1 Satunnaisotokset

RANSAC-menetelmässä pyritään käyttämään minimaalisia satunnaisotoksia, jotta virheellisten mittapisteiden valikoituminen otokseen olisi mahdollisimman pieni. Minimaalinen otos on pienin joukko havaintoja, jolla sovitettavan mallin parametrit pystytään määrittämään yksikäsitteisesti. Esimerkiksi sovitettaessa ympyrää 2-ulotteiseen koordinaatistoon minimaalinen otoskoko on 3 koordinaattipistettä, sillä 3 koordinaattipistettä on pienin määrä pisteitä, joka määrittää yksikäsitteisen kaksiulotteisen ympyrän. Alkuperäisessä RANSAC-menetelmässä otokset S_i valitaan satunnaisesti koko havaintodatan D joukosta niin, että kaikilla havaintopisteillä d on yhtä suuri todennäköisyys tulla valituksi otokseen. [4]

Tehokkuuden ja tarkkuuden parantamiseksi alkuperäisestä menetelmästä on kuitenkin kehitetty versioita, joissa otokset S_i eivät ole täysin satunnaisia vaan niiden valintaa ohjaillaan. RANSAC-menetelmän GroupSAC-variaatiossa havaintodatapisteitä $d \in D$ luokitellaan keskenään samankaltaisten pisteiden luokkiin $L \subset D$ datasta saatavilla olevan a priori -tiedon avulla. Taustalla on ajatus siitä, että parhaan mallin M_{max} pohjana olevien eli hyvien havaintojen joukko H koostuu jollain tavalla keskenään samankaltaisista

havaintopisteistä, ja H voidaan löytää nopeammin tutkimalla satunnaisten otosten sijaan keskenään samankaltaisista havainnoista koostuvia dataluokkia L . [9]

NAPSAC-menetelmän (N-adjacent Points Sample Consensus) voidaan ajatella olevan erityistapaus GroupSAC-menetelmästä: siinä pisteiden samankaltaisuuden mittarina toimii niiden spatiaalinen läheisyys. Otoksen S_i ensimmäinen havaintopiste d valitaan satunnaisesti koko havaintodatasta D , mutta otos täydennetään haluttuun kokoon valitsemalla pisteitä, jotka ovat tietyn spatiaalisen säteen r sisällä ensimmäisenä valitusta pisteestä. NAPSACia on käytetty tilanteissa, joissa havaintodata-avaruuden ulottuvuus n on suuri. Muun muassa Schnabel ym. [8] ovat edelleen kehittäneet otosten valikointia pisteiden läheisyyden perusteella, mutta myös päinvastaisia, satunnaisotoksen pisteiden hajauttamiseen perustuvia otantatapoja on esitetty degeneraatio-ongelmien välttämiseksi. [9] Degeneraatio on jätetty tämän työn käsittelyn ulkopuolelle.

PROSAC-menetelmässä (Progressive Sample Consensus) satunnaisotokset S_i asetetaan parhausjärjestykseen, jolloin lupaavimmat otokset voidaan käsitellä ensimmäiseksi ja paras mallihypoteesi M_{max} löytää huomattavasti nopeammin. Otosten parhausjärjestys määrittyy havaintodatapisteiden $d \in D$ laatuerojen perusteella, ja se voi sovelluksesta riippuen perustua esimerkiksi havaintodatan tuottavan mittaussignaalin voimakkuuseroihin eri havaintopisteiden kohdalla tai datapisteet hierarkkiseen järjestykseen asettamaan funktioon. GroupSAC-menetelmän tavoin tämänkin menetelmän käyttö vaatii, että havaintodatasta on saatavilla jonkinlaista a priori -tietoa, jonka perusteella luokittelua voidaan tehdä. [11] Luvussa 3.2.2 esiintyvistä MLESAC-menetelmästä on kehitetty Guided-MLESAC -niminen versio, jossa a priori -tietoa käytetään avuksi otosten valikoimisessa Monte Carlo -menetelmää hyödyntäen [12].

Kun otos S_i on valikoitu, sen laatu voidaan tarvittaessa tarkastaa ennen seuraavia vaiheita eli hypoteesimallin M_i sovittamista ja sen tuen laskemista eli konsensusjoukon C_i muodostamista. Tämä voi olla kustannustehokasta, jos käytettävissä on jokin nopea testi, jonka perusteella otos S_i voidaan tarvittaessa suoraan hylätä, ja jos testin käyttäminen vaatii vähemmän laskentakapasiteettia kuin mallin M_i sovittaminen ja sen tuen C_i laskeminen turhaan huonon otoksen pohjalta. [9] Tällainen testi voi riippua sovelluskohtaisista sovitusongelman rajoitteista, jotka ratkaisun tulee täyttää. Kyseessä voi myös olla esimerkiksi sen tarkistaminen, onko otos uniikki vai onko sama otos käsitelty jo aikaisemmin. Voi kuitenkin tulla laskennallisesti halvemmaksi esimerkiksi käsitellä silloin tällöin sama otos kahdesti kuin tarkistaa jokaisella iteraatiokierroksella, onko samaa otosta käsitelty aikaisemmin [12].

Tässä työssä otokset valikoidaan alkuperäisen RANSAC-algoritmin tavoin satunnaisesti. Työn sovellusongelman näkökulmasta laserkeilauksessa käytettävien signaalien intensiteettiä eli mittauksen kohteesta takaisin heijastuvien lasersäteiden voimakkuutta on mahdollista käyttää a priori -tietona, jonka perusteella mittapisteitä ja niistä koostuvia otoksia

voitaisiin luokitella. Tässä työssä tällaista tietoa ei käytetä, vaan havaintodata sisältää pelkästään mittauspisteiden avaruudelliset koordinaatit. Myös otospisteiden läheisyyteen ja toisaalta hajauttamiseen perustuvien menetelmien hyötyjen ja haittojen arviointi ja testaus jätetään työn ulkopuolelle.

3.2.2 Kustannusfunktio

RANSAC-menetelmää voidaan pitää optimointina, jonka kustannusfunktiona toimii havaintodatan D tuki hypoteesimallille M_i eli hypoteesimallin M_i konsensusjoukon C_i koko

$$|C_i| = |\{d \mid d \in D, e < t\}|. \quad (3.1)$$

Yhtälössä 3.1 arvo e on hypoteesimallin M_i ja datapisteen d välinen etäisyys ja t on virhetoleranssitaso. Tarkasteltaessa RANSACia optimointimenetelmänä, sen kustannusfunktion F arvo hypoteesimallille M_i voidaan kirjoittaa minimointiongelmana muodossa

$$F_i = \sum_{j=1}^N \rho(e_j^2), \quad (3.2)$$

missä N on havaintodatajoukon D kardinaliteetti $|D|$ eli havaintopisteiden lukumäärä, e_j on havaintopisteen d_j virhe eli etäisyys sovitettavaan malliin M_i ja $\rho(e_j^2)$ on paloittain määritelty funktio [13]:

$$\rho(e_j^2) = \begin{cases} 0, & \text{kun } e_j^2 < t^2 \\ a, & \text{kun } e_j^2 \geq t^2 \end{cases}. \quad (3.3)$$

Paloittain määritelty sakkofunktio $\rho(e_j^2)$ antaa arvon 0 pisteille d , joiden virhe e sovitettavaan malliin mahtuu ennalta määritetyn virhetoleranssitason t sisään, ja positiivisen vakioarvon a pisteille d , jotka poikkeavat sovitettavasta mallista toleranssitason t verran tai enemmän. Yhtälössä 3.2 virhetoleranssitason ylittävät pisteet siis kasvattavat summan F_i arvoa, kun taas virhetoleranssitason sisään sopivat pisteet eivät. Tällainen funktion $\rho(e_j^2)$ määrittely tekee RANSAC-kustannusfunktiosta F minimointiongelman, jossa suurin konsensusjoukon koko $|C_{max}|$ tuottaa pienimmän kustannusfunktion F arvon.

RANSAC-kustannusfunktio F voidaan kirjoittaa myös maksimointiongelmana vaihtamalla funktion $\rho(e_j^2)$ antamat arvot 0 ja a keskenään toisin päin [9]. Tällöin virhetoleranssitason t sisään sopivat pisteet kasvattavat kustannusfunktion F arvoa, kun taas virhetoleranssitason saavuttavat ja ylittävät pisteet saavat arvon 0.

Alkuperäinen RANSAC ei erottele yhtä suuret konsensusjoukot kerääviä hypoteesimal-

leja M_i toisistaan. Ratkaisuna tähän ongelmaan Torr ym. esittelevät vuoden 2000 artikkelissaan variaation nimeltä MSAC, joka muotoilee RANSAC-kustannusfunktion uudella tavalla. MSAC-kustannusfunktiossa optimointiongelma kirjoitetaan yhtälön 3.2 minimointiongelmana, ja paloittain määritelty funktio $\rho(e_j^2)$ on muodossa [13]:

$$\rho(e_j^2) = \begin{cases} e_j^2, & \text{kun } e_j^2 < t^2 \\ t^2, & \text{kun } e_j^2 \geq t^2 \end{cases} \quad (3.4)$$

Alkuperäisen RANSAC-kustannusfunktion tavoin yhtälössä 3.4 virhetoleranssitasen t ylittävät havaintopisteet kasvattavat yhä kustannusfunktion arvoa vakiolla, jonka arvoksi Torr ym. valitsivat t^2 . RANSAC-kustannusfunktioista poiketen MSACissa hypoteesimallin M_i kanssa yhteensopivat havaintopisteet eivät kuitenkaan saa arvoa 0, vaan nekin kasvattavat kustannusfunktion F_i arvoa kunkin pisteen virhetermin e_j neliöllä. Käytännössä MSAC siis asettaa pienimmän neliösumman menetelmällä keskinäiseen parhausjärjestykseen myös sellaiset mallit, joiden konsensusjoukot ovat yhtä suuret. Torrin ym. mukaan MSAC saavuttaa RANSACia parempia tuloksia lisäämättä algoritmin laskennallista taakkaa, joten heidän mukaansa alkuperäisen RANSAC-kustannusfunktion käytölle MSAC-kustannusfunktion sijaan ei ole mitään syytä. [13]

MSAC-menetelmän kanssa samassa artikkelissa Torr ym. esittelivät MLESAC-menetelmän [13]. MLESAC käyttää RANSAC:in tavoin satunnaisotoksia S_i , joihin sovitetaan hypoteesimalleja M_i . MLESAC:in ideana on kuitenkin vertailla mallien uskottavuutta (engl. likelihood) eli havaintodatajoukon D todennäköisyyttä $p(D|M_i)$ kunkin hypoteesimallin M_i ehdolla. Tämä tehdään, koska Bayesin kaavan mukaan mallin uskottavuus $p(D|M_i)$ on suoraan verrannollinen posterioritodennäköisyyteen $p(M_i|D)$ eli hypoteesimallin M_i todennäköisyyteen havaintodatajoukon D ehdolla, ja hypoteesimallien M_i joukosta halutaan todellisuudessa valita paras juuri parhaan posterioritodennäköisyyden perusteella. [12] Torrin ym. mukaan MLESAC on RANSAC-menetelmän yleistys, sillä MLESAC valitsee satunnaisotokseen S_i sovitetuista malleista M_i parhaan suurimman todennäköisyyden perusteella, ja RANSAC-menetelmässä konsensusjoukon koon $|C_i|$ voidaan ajatella mittaavan tätä todennäköisyyttä. Torrin ym. artikkelissa todennäköisyyttä mitattiin sekoitemallin (engl. mixture model) avulla, olettaen poikkeavien havaintojen jakautuvan tasaisesti koko mittausalueelle ja lopullisen mallin mukaisten pisteiden kohinan olevan normaalisti jakautunutta. Artikkelissa MLESAC saavutti MSACia hieman parempia tuloksia, mutta lisäsi toisaalta vaadittavan laskentakapasiteetin määrää. [13]

Tässä työssä hypoteesimallin M_i tuen muodostamista käsitellään minimointiongelmana. Yhtälössä 3.2 funktiosta $\rho(e_j^2)$ käytetään sekä RANSAC-kustannusfunktion määrittelyä kaavan 3.3 tapaan että MSAC-kustannusfunktio-*muotoa* kuten kaavassa 3.4 ja tuloksia vertaillaan toisiinsa. MLESAC ohitetaan, sillä tarvittavat lisäpanostukset resurssihin katsotaan kannattamattomiksi suhteessa odotettuihin hyötyihin MSAC-menetelmän käyttöön

verrattuna. Poikkeavien ja hyvien havaintojen jakautumisesta mittausalueelle ei tällöin myöskään tarvitse tehdä artikkelin [13] mukaisia oletuksia.

3.2.3 Konsensusjoukon muodostaminen

Hypoteesimallin M_i konsensusjoukko C_i koostuu niistä pisteistä d , jotka sopivat malliin M_i eli poikkeavat siitä vähemmän kuin määritellyn virhetoleranssitason t verran. Alkuperäisessä RANSACissa konsensusjoukon suuruus $|C_i|$ kertoo, kuinka hyvin malli M_i sopii havaintodataan D . Havaintopisteille d voidaan asettaa virhetoleranssitason t sisään mahdumisen lisäksi myös muita lisäehtoja sille, että ne hyväksytään konsensusjoukkoon (katso esim. [8]). Ehdot voivat olla sovelluskohtaisia ja perustua kulloinkin käsittelyssä olevan sovituserongelman rajoitteisiin.

Alkuperäisessä RANSAC-menetelmässä koko havaintodata D käydään läpi kunkin hypoteesimallin M_i verifiointin kohdalla, eli kaikkia $N = |D|$ datapistettä d vertaillaan kuhunkin k sovitettavaan malliin M_i . Tämä voi vaatia paljon laskentakapasiteettia, jos datajoukko D ja iteraatiomäärä k ovat suuria. Algoritmia saadaan tehostettua tarkistamalla kunkin hypoteesimallin M_i kohdalla ensin pieni satunnainen osajoukko havaintodatasta D ja käymällä koko havaintodata D läpi vasta, jos malli M_i vaikuttaa tämän pienen osajoukon tuen perusteella lupaavalta; muussa tapauksessa malli hylätään. [11]

Tällä periaatteella toimivat esimerkiksi $T_{d,d}$ -testi, Bail-out -testi ja WaldSAC-menetelmässä käytettävä SPRT-testi, joista kaikilla saavutetaan parannuksia hypoteesimallien M_i verifiointitehokkuuteen. $T_{d,d}$ -testiä käytettäessä tarkistetaan minimaalisesta joukosta havaintopisteitä, suosituksen mukaan vain yhdestä pisteestä, sopivatko ne käsiteltyyn malliin M_i ja jos eivät sovi, malli M_i hylätään. Menetelmän nimi tulee Matasin ja Chumin vuoden 2004 artikkelista [14], jossa testi esiteltiin ja valittujen testipisteiden lukumäärää merkittiin kirjaimella d . Matas ja Chum tiivistivät artikkelissaan $T_{d,d}$ -testin toimintaperiaatteen toteamalla, että testi läpäistään, jos d kappaleesta valittuja datapisteitä kaikki d kappaletta tukevat testattavaa hypoteesimallia M_i . Menetelmä pitää sisällään riskin siitä, että hyviäkin malleja hylätään virheellisesti epäonnisesti valitun testipisteen d perusteella, joten $T_{d,d}$ -testin käyttö saattaa johtaa tarvittavien iteraatiokierrosten lisääntymiseen ennen kuin paras malli todella havaitaan.

Hieman kehittyneemmät bail-out-testi ja SPRT-testi (Sequential Probability Ratio Test) tarkastelevat kullakin i . iteraatiokierroksella siihen asti löydettyjen konsensusjoukkojen suuruuksiin perustuvaa todennäköisyyttä löytää niitä kaikkia suurempi konsensusjoukko C_i sillä hetkellä käsiteltävälle mallille M_i , ja malli M_i hylätään tämän todennäköisyyden osoittautuessa liian pieneksi. Bail-out-testin perustana on se, että mallin M_i mukaisten havaintojen osuus havaintodatajoukossa D noudattaa hypergeometrista jakaumaa, kun taas SPRT-testi perustuu uskottavuusosamäärän käyttöön. [11] Myös bail-out-testi voi $T_{d,d}$ -testin tavoin hylätä virheellisesti hyvän mallin, mutta käytännön kokeissa se on te-

hostanut RANSACin toimintaa jopa 2-7-kertaisesti [9]. SPRT-testi on optimaalinen hyvien havaintojen ϵ osuuden havaintodatasta D ollessa tiedossa [11]. Kuitenkin tämän tiedon puuttuessa se on osoittautunut hieman bail-out-testiä tehokkaammaksi, varsinkin vaikeampien ja enemmän poikkeavia datapisteitä sisältävien ongelmien kohdalla, tehostaen RANSAC:n toimintaa jopa 2-9-kertaisesti [9].

Näistä kolmesta testistä konseptuaalisesti yksinkertaisinta $T_{d,d}$ -testiä sovelletaan tässä työssä sylinterien sovittamisessa pistepilvidataan. Tavoitteena on määrittää, saavutetaanko sillä työn sovellusongelman kohdalla testittömään RANSAC-algoritmin käyttämiseen verrattuna jonkinlaisia tehokkuushyötyjä, eli laskeeko konsensusjoukkojen muodostamiseen käytetty aika ja toisaalta nouseeko tarvittava RANSAC-iteraatiokierrosmäärä.

3.2.4 Virhetoleranssitaso

Sopivan virhetoleranssitason t asettaminen on tärkeää RANSAC-algoritmin käytön onnistumiseksi. Liian suuri arvo t estää menetelmää erottelemasta poikkeavia ja hyviä havaintopisteitä toisistaan mielekkäästi, kun hypoteesimalleista M_i huomattavasti poikkeavia-kin datapisteitä hyväksytään mukaan konsensusjoukon C_i muodostamiseen ja näin huononkin mallin näennäinen tuki voi kasvaa suureksi. Jos taas virhetoleranssitaso on liian tiukka, todellisuudessa paras malli M_i voi jäädä löytymättä sen konsensusjoukon C_i jäädessä virheellisesti liian pieneksi. Liian pienen arvon t seurauksena RANSAC-algoritmin ajoaika eli iteraatiokierrosten määrä k myös kasvaa, kun luvun 3.2.5 yhtälössä 3.8 hyvien havaintojen arvioitu lukumäärä ϵ_e arvioidaan ajon aikana liian pieneksi liian pienten konsensusjoukkojen löytymisen seurauksena. Virhetermien e_j jakaumien määrittäminen analyttisesti ihanteellisen virhetoleranssitason t asettamiseksi ei usein ole mahdollista, ja arvo t on määritetäänkin käytännössä usein empiirisesti [4] [15].

Virhetoleranssitason asettaminen on kirjan *Multiple View Geometry in Computer Vision* [15] mukaan mahdollista formalisoida, jos hyvien mittapisteiden kohina eli virhe malliin M_{max} oletetaan normaalijakautuneeksi siten, että keskiarvo on 0 ja jakauman keskihajonta σ on mahdollista määrittää. Tällöin hyvien mittapisteiden $d \in H$ virheiden arvoja e_j^2 malliin M_{max} voidaan mallintaa khii-jakauman avulla ja näin pyrkiä varmistamaan halutulla luottamustasolla, että hyvä havainto tunnistetaan oikein esimerkiksi 95 % ajasta.

Jotkut menetelmät eivät vaadi kiinteän arvon t asettamista vaan arvioivat sitä algoritmin ajon aikana käsitellyn havaintodatan D ominaisuuksien perusteella. Riu ym. [16] kutsuvat vuoden 2022 artikkelissaan tällaisia menetelmiä automaattisiksi (engl. automatic) RANSAC-variaatioiksi ja vertailevat kyseisessä artikkelissa niistä kahta. AC-RANSAC (a contrario RANSAC) kokeilee monia eri toleranssitasoja t . Menetelmä valitsee parhaan hypoteesimallin M_i sen mukaan, mikä kokeilluista arvoista t minimoi NFA:ksi kutsutun statistiikan (Number of False Alarms) joka kuvaa heikkolaatuisten mallien virheellisen hyväksymisen todennäköisyyttä. LRTSAC-menetelmä (Likelihood Ratio Test Sample Consen-

sus) puolestaan käyttää uskottavuusosamäärätestiä (engl. likelihood ratio test) pyrkiesään maksimoimaan todennäköisyyden sille, että generoidun hypoteesimallin M_i pohjana on todella hyvien pisteiden joukko $H \subset D$ eikä muita, satunnaisesti jakautuneita pisteitä. Riu ym. toteaa artikkelissaan, että joidenkin LRTSACin ja AC-RANSACin piirteiden yhdistely toisi parannuksia tehokkuuteen AC-RANSACiin verrattuna ja tarkkuuteen LRTSAC-menetelmään verrattuna. [16] StaRSaC (Stable Random Sample Consensus) kokeilee AC-RANSACin tavoin joitakin eri toleranssitason t arvoja, ja siinä käyttöön valikoituu arvo joka tuottaa vähiten parametrivariaatiota ("variance of parameters"). StaRSaC-menetelmän käytössä tarkkuusparannuksia saavutetaan algoritmin ajoajan pitenemisen kustannuksella. [9]

Jotkut menetelmät eivät käytä virhetoleranssiarvoa t lainkaan. MAGSAC-menetelmässä (Marginalised Sample Consensus) toleranssitaso t arvioidaan tasajakautuneeksi välillä $(0, t_{\max})$ ja muodostetaan marginaalijakauma, jossa t poistuu eikä sen tarkkaa arvoa tarvitse määrittää. RECON (Residual Consensus) arvioi eri hypoteesimallien M_i konsensusjoukkojen C_i virhetermien e_j jakaumia. Löytäessään esimerkiksi 3 kappaletta malleja M_i , joiden konsensusjoukkojen virhetermit muistuttavat läheisesti toisiaan, RECON sovittaa lopullisen mallin näiden kolmen mallin synteessä. Samankaltaisten virhetermien ajatellaan kertovan siitä, että kaikkien kolmen hypoteesimallin pohjalla olleet satunnaisotokset S_i ovat tulleet yhteiseen, tavoiteltavaan malliin M_{\max} pohjautuvasta pistejoukosta, joka on hyvien havaintojen joukko H . [16] RECON vaatii RANSACin perusversiota enemmän laskentakapasiteettia vertaillakseen malleja keskenään [9].

Luvussa 3.2.2 esitelty MSAC-menetelmä vähentää Torrin ym. [13] mukaan virhetoleranssitason t valintaan liittyvää epävarmuutta, sillä MSAC asettaa hypoteesimallit M_i parhausjärjestykseen. Tällöin heikot ja parhaat mallit voidaan tunnistaa silloinkin, kun arvon t asettaminen optimaalista suuremmaksi tuottaa useampia suuria konsensusjoukkoja C_i . Arvon t asettamisen optimointi on jätetty tässä työssä käsittelyn ulkopuolelle, ja arvo asetetaan heuristisesti laserkeilauksen resoluutio huomioiden.

3.2.5 Iteraatioiden määrä ja lopetuskriteerit

Alkuperäisessä, Fischlerin ym. RANSAC-menetelmän esittelevässä artikkelissa RANSAC-algoritmi terminoidaan, kun on löydetty konsensusjoukko C_i joka on tarpeeksi suuri tullaan hyväksytyksi parhaaksi konsensusjoukoksi C_{\max} . Fischlerin ym. mukaan hyväksyttävän konsensusjoukon koon alaraja riippuu satunnaisotosten S_i koosta m ja hyvien havaintojen osuudesta ϵ havaintodatassa D . Artikkelissa todetaan, että 5 havaintopistettä otoskokoa m suurempi konsensusjoukko C_{\max} takaa 95% todennäköisyyden olla valitsematta virheellisesti heikkoa mallia, jos hyvien havaintojen osuus ϵ havaintodatajoukossa D oletetaan suuremmaksi kuin 0,5 eli jos yli puolet datasta on hyvälaatuista. [4]

2000-luvulle tultaessa standardina RANSAC-lopetuskriteerinä pidetään kuitenkin Ragu-

ramin ym. mukaan tarvittavien iteraatiokierrosten määrän k suorittamista sen takaamiseksi, että tietyllä luottamustasolla η edes yhdellä iteraatiokierroksella on poimittu satunnaisotos S_i , jossa ei ole ollut yhtään poikkeavaa havaintopistettä [11]. Myös tässä työssä lopetuskriteerinä toimii ensisijaisesti tämän tarvittavan vähimmäiskierrosmäärän suorittaminen silloin, kun se ei kasva algoritmin suoritusajan kohtuullisena pitävää maksimikierrosmäärää suuremmaksi. Jo Fischler ym. [4] esittelivät artikkelissaan teoreettisen kaavan tarvittavan iteraatiomäärän k laskemiseksi, mutta heidän artikkelissaan arvio kierrosmäärästä k toimi lopetuskriteerin sijasta RANSACin asympotoottisuuden määrittäjänä.

Johdetaan seuraavaksi alaraja tarvittavalle iteraatiokierrosmäärälle k Fischleriä ym. mukaillen. Halutaan varmistaa, että η luottamustasolla k kappaletta satunnaisotoksia S_i vähintään yksi otoksista on hyvälaatuinen eli koostuu ainoastaan hyvistä havaintopisteistä $d \in H$. Toisin sanoen halutaan varmistaa, että todennäköisyys kaikille k otokselle olla huonolaatuisia eli vähintään yhden poikkeavan havainnon $d \in P$ sisältäviä on korkeintaan $1 - \eta$. Olkoon p_H todennäköisyys sille, että m kappaletta havaintodatapisteitä $d \in D$ sisältävä yksittäinen satunnaisotos S_i on hyvälaatuinen. Kun ϵ on hyvien havaintojen $d \in H$ osuus koko havaintodatassa D , tämä yksittäisen hyvälaatuisen otoksen todennäköisyys p_H on

$$p_H = \epsilon^m. \quad (3.5)$$

Tällöin todennäköisyys p_P sille, että yksittäinen satunnaisotos S_i on huonolaatuinen, on komplementti

$$p_P = 1 - p_H,$$

josta sijoittamalla yhtälö 3.5 saadaan

$$p_P = 1 - \epsilon^m. \quad (3.6)$$

Todennäköisyys sille, että kaikki k kappaletta otoksia ovat huonolaatuisia, on

$$p(k \text{ huonoa otosta}) = p_P^k$$

joka yhtälön 3.6 avulla on

$$p(k \text{ huonoa otosta}) = (1 - \epsilon^m)^k.$$

Kun halutaan varmistaa tämän todennäköisyyden saavuttavan korkeintaan arvon $1 - \eta$,

voidaan kirjoittaa

$$(1 - \epsilon^m)^k \leq 1 - \eta,$$

mistä logaritmioperaatiolla puolittain saadaan

$$k \cdot \log(1 - \epsilon^m) \leq \log(1 - \eta). \quad (3.7)$$

Jakamalla puolittain termillä $\log(1 - \epsilon^m)$ yhtälö 3.7 saadaan muotoon

$$k \geq \frac{\log(1 - \eta)}{\log(1 - \epsilon^m)}. \quad (3.8)$$

Viimeisessä vaiheessa epäyhtälömerkin suunta vaihtuu, koska jakajatermin $\log(1 - \epsilon^m)$ merkki on negatiivinen arvon ϵ ollessa vähemmän kuin 1.

Kun yhtälössä 3.8 ϵ pienenee tai m kasvaa, arvo k kasvaa. Sekä enemmän poikkeavia havaintoja sisältävät havaintodatajoukot D että suuremmat otoskoot m siis kasvattavat tarvittavien otosten S_i eli RANSAC-iteraatiokierrosten määrää k . Usein kaavassa hyvien havaintojen osuutta ϵ ei tunneta tarkasti, jolloin siitä voidaan algoritmin alussa tehdä pessimistinen arvio ϵ_e , jota iteroinnin kuluessa päivitetään käyttämällä kullakin iteraatiokierroksella i suurinta siihen asti löydettyä konsensusjoukon C_i kokoa [11]. Tämä antaa arviolle tarvittavasta iteraatiokierrosmäärästä k ylärajan, joka voi iteroinnin kuluessa laskea muttei kasva. Syynä on, että jo löydettyjä konsensusjoukkoja C_i pienempiä joukkoja ei hyväksytä vaan arvon k uudelleenarviointiin käytetään aina vain aiempaa suurempaa konsensusjoukkoa C_i , joten estimoitua arvoa ϵ_e voidaan kaavassa 3.8 päivittää ylöspäin muttei alaspäin.

Esimerkki 3.3. Olkoon arvio esimerkkien 3.1 ja 3.2 havaintodatajoukon D ϵ -arvosta $\epsilon_e = 0,2$, ja olkoon haluttu luottamustaso ainakin yhden hyvän RANSAC-otoksen saamiseksi $\eta = 0,95$. Minimiotoskoko yksikäsitteisen ympyrämallin muodostamiseksi on $m = 3$. Tällöin kaavan 3.2.5 avulla odotetuksi tarvittavien RANSAC-iteraatiokierrosten määräksi saadaan

$$\begin{aligned} k &\geq \frac{\log(1 - 0,95)}{\log(1 - 0,2^3)} \\ &= 372,966\dots \\ \Rightarrow k &= 373. \end{aligned}$$

Alunperin tarvittavien RANSAC-otosten määräksi k arvioidaan siis 373 kappaletta. Jos

i . iteraatiokierroksella kuitenkin löydetään hypoteesimalli M_i jonka konsensusjoukkoon C_i kuuluu esimerkiksi 31 % datajoukon D pisteistä, on uusi arvioitu hyvien havaintojen osuus $\epsilon_e = 0,31$ jolloin uudelleenarvioitu tarvittavien RANSAC-iteraatioiden määrä on

$$\begin{aligned} k &\geq \frac{\log(1 - 0,95)}{\log(1 - 0,31^3)} \\ &= 99,052\dots \\ \Rightarrow k &= 100. \end{aligned}$$

Kun löydetään aiempaa suurempia konsensusjoukkoja C_i , arvioitu tarvittavien otosten eli iteraatiokierrosten määrä siis laskee.

Raguramin ym. mukaan [9] RANSAC käyttää usein käytännössä enemmän iteraatioita kuin kaavalla 3.8 voitaisiin datan D todellisen hyvien havaintojen osuuden ϵ pohjalta arvioida. Syynä tähän on se, että RANSACin tuottama paras malli M_{max} ei välttämättä saa konsensusjoukkoonsa C_{max} kaikkia hyviä havaintopisteitä $d \in H$, sillä malli M_{max} voi sisältää kohinaa johtuen pohjana olleen satunnaisotoksen S_{max} havaintopisteiden kohinasta. Tällöin joukon C_{max} koon pohjalta arvioitu ϵ_e kaavassa 3.8 jää pienemmäksi kuin havaintodatan D todellinen hyvien havaintojen osuus ϵ , minkä seurauksena RANSAC-iteroinnin aikana arvioidujen ϵ_e -arvojen perusteella kaavalla 3.8 päivitetty iteraatiomäärä k jää suuremmaksi kuin todellisen ϵ -arvon pohjalta laskettu k .

RANSAC-variaatiot, jotka suorittavat iteraatiokierroksia i yksi kerrallaan alusta loppuun eli muodostavat satunnaisotoksen S_i , sovittavat siihen mallin M_i ja laskevat konsensusjoukon C_i ennen siirtymistä seuraavalle iteraatiokierrokselle $i + 1$, ovat niin sanotusti syvyys-suunnassa eteneviä menetelmiä (engl. depth-first). Preemptive RANSAC on esimerkiksi niin sanotusti leveyssuunnassa etenevästä menetelmästä (engl. breadth-first), joissa ideana on suorittaa hypoteesimallien muodostus- ja käsittelyvaiheet yksittäisen hypoteesimallin sijasta joukolle hypoteesimalleja kerrallaan. Preemptive RANSACia käytettäessä ensin valitaan jokin ennalta määritetty kiinteä määrä k^* satunnaisotoksia S_i ja sovitetaan niihin hypoteesimallit M_i . Tämän jälkeen mallihypoteesit asetetaan paremmuusjärjestykseen laskemalla niille alustavat tuet hyvin pienestä osajoukosta havaintodataa D . Alustavan tuen perusteella vain parhaat mallit otetaan jatkokäsittelyyn, jossa niille muodostetaan konsensusjoukot C_i koko havaintodatasta D . [11]

Preemptive RANSACin tapaisia leveyssuunnassa eteneviä menetelmiä käytetään tyypillisesti silloin, kun tavoitellaan nopeita, jopa reaaliaikaisia tuloksia ja halutaan löytää "tarpeeksi hyvä" malli tietyn ennalta määritetyn laskenta-ajan puitteissa. Niiden heikkoutena voidaan kuitenkin pitää niiden jäykkyyttä: jos ennalta päätetty määrä k^* mallihypoteeseja M_i on liian pieni, todellisuudessa paras havaintodataan sopiva malli voi jäädä löytymättä,

ja jos taas iteraatioiden määrä k^* on tarpeettoman suuri, olisi syvyysuunnassa etenevä ja tarvittavaa kierrosmäärää k ajon aikana alaspäin päivittävä menetelmä voinut olla jopa nopeampi. [11]

Raguram ym. [11] pyrkivät vuoden 2008 artikkelissaan yhdistämään syvyys- ja leveysuunnassa etenevien menetelmien hyvät puolet ARRSAC-menetelmässä (Adaptive Real-Time Random Sample Consensus). ARRSAC-algoritmi tuottaa alussa pienen joukon hypoteesimalleja M_i ja laskee niille tuen pienestä havaintodatan D osajoukosta. Tämän tuen perusteella algoritmi arvioi havaintodatan D laatua ja päivittää arviota hyvien havaintojen osuudesta ϵ_e , minkä pohjalta tarvittavien hypoteesimallien määrää k päivitetään ja hypoteesimalleja tuotetaan tarvittaessa lisää, kuitenkin vain ennalta määritettyyn ylärajaan N asti. ARRSAC siis mahdollistaa jonkin verran dataan mukautumista ja asetettua ajoaikaa nopeamman suoriutumisen, muttei kuitenkaan riko asetettua maksimijoaikaa.

Myöhemmässä, vuoden 2013 artikkelissaan Raguram ym. ehdottavat inner-RANSACin ja virhetoleranssitason t iteratiivisen skaalaamisen yhdistelmälle perustuvaa hypoteesimallien M_i hienosäätömenetelmää, joka heidän mukaansa laskee RANSACin vaatiman iteraatiokierrosmäärän kaavan 3.8 ennustamalle tasolle [9] lisäämättä laskennallisen kapasiteetin tarvetta merkittävästi. Menetelmä kuvataan hypoteesimallien uudelleensovittamista koskevassa luvussa 3.2.6 ja siitä toteutetaan tässä työssä yksinkertainen versio. Työssä ei tavoitella reaaliaikaisuutta eikä sovelleta leveysuunnassa eteneviä RANSAC-menetelmiä.

3.2.6 Mallin sovittaminen ja uudelleensovittaminen

Koska hyvätkin havaintopisteet voivat sisältää kohinaa, voi hyvälaatuiseenkin otokseen S_i sovitettu malli M_i sisältää kohinasta johtuvaa virhettä. Minimaalisen otokseen m takia otosten S_i yksittäisten havaintopisteiden d vaikutus muodostuvaan hypoteesimalliin M_i on erityisen suuri [9]. Tästä syystä suuren konsensusjoukon C_i perusteella hyvältäkin vaikuttavaa mallia M_i voidaan haluta hienosäätää. Alunperin minimaalisen otoksen S_i perusteella muodostettu malli M_i voidaan uudelleensovittaa mallille löydettyyn konsensusjoukkoon C_i . Tässä askeleessa pienimmän neliösumman menetelmän käyttö on ollut standardi, sillä menetelmän heikkous poikkeavien havaintojen $d \in P$ harhaanjohtavalle vaikutukselle ei ole ongelma olettaessa että kaikki sovituksessa käytettävät havaintopisteet on varmistettu hyviksi havainnoiksi $d \in H$ [8].

Myös muita sovitusmenetelmiä on kehitetty. Eräs näistä on inner-RANSAC, joka konsensusjoukon C_i muodostamisen jälkeen toteuttaa RANSAC-iteraatioprosessin käyttäen havaintodatajoukon D sijasta konsensusjoukkoa C_i , mikäli C_i on suurin toistaiseksi löydetty konsensusjoukko. Ensimmäisen kertaluokan iteraatiokierroksen i perusteella muodostetusta konsensusjoukosta C_i siis valitaan toisen kertaluokan RANSAC-satunnaisotoksia S_j , joihin sovitetaan mallit M_j ja muodostetaan malleille konsensusjoukot C_j . Jos jokin

uudelleensovituksista tuottaa aiempaa paremman konsensusjoukon, otetaan uudelleensovitus käyttöön. Erona ensimmäisen kertaluokan RANSAC-iterointiin inner-RANSAC-vaiheessa otoskoon m_{inner} ei tarvitse olla minimaalinen. Lisäksi inner-RANSAC-vaiheessa otosten määrä k_{inner} on etukäteen määritetty, tyypillisesti 10-20 kappaletta. [9]

Raguram ym. [9] yhdistävät vuoden 2013 artikkelissaan inner-RANSACiin virhetoleranssitason t skaalaamista iteraatiokierroksittain. Siinä jokaiseen inner-RANSAC-otokseen S_j sovitettu hypoteesimalli M_j uudelleensovitetään vielä havaintodatapisteisiin $d \in D$, joiden virhetermi e malliin M_j nähden on pienempi kuin arvo Kt , missä t on RANSACissa käytetty virhetoleranssitaso ja K etukäteen määritetty painokerroin. Tätä mallin uudelleensovittamista skaalatun virhetason Kt sisään sopiviin pisteisiin jatketaan pienentäen joka kierroksella arvoa K , kunnes K saavuttaa arvon 1 eli kunnes termi Kt saavuttaa arvon t . Tämän yhdistelmämenetelmän käyttö saattaa Raguramin ym. mukaan RANSAC-iteraatiokierrosmäärän kaavan 3.8 ennustamalle tasolle.

Inner-RANSACia verrataan tässä työssä RANSAC-algoritmiin, jossa uudelleensovitukseen on käytetty pelkästään pienimmän neliösumman menetelmää hypoteesimallin M_i koko konsensusjoukossa. Lisäksi uudelleensovitukseen testataan virhetoleranssitason t asteittaista skaalaamista alaspäin muutaman iteraatiokierroksen ajan ja hypoteesimallin uudelleensovittamista skaalatun toleranssitason sisään sopiviin pisteisiin. Myös inner-RANSACin ja skaalatun virhetoleranssitason yhdistelmää kokeillaan. Skaalatun virhetoleranssitason käyttö ja sen yhdistäminen inner-RANSACiin mukailevat Fischlerin ym. alkuperäisessä, vuoden 1981 RANSAC-artikkelissaan kuvailemia heuristisia lähestymistapoja, joiden avulla poikkeavien havaintojen vaikutusta mallien sovittamiseen oli pyritty vähentämään ennen RANSAC-menetelmää [4]. Fischlerin ym. mukaan tällaiset menetelmät eivät kykene käsittelemään datassa esiintyviä huomattavia poikkeavia havaintoja eli suuressa määrin virheellisiä mittapisteitä, mutta alunperin RANSAC-algoritmillä muodostetun hypoteesimallin M_i uudelleensovitusvaiheessa voidaan olettaa, että alkuperäiseen mallisovitukseen käytetty RANSAC-otosmenetelmä on jo minimoinut virheellisten havaintojen vaikutuksen.

4. MENETELMÄT

Seuraavaksi kerrotaan tarkemmin, mitä menetelmiä luvussa 3 esitellyistä RANSAC-versioista on valittu käytettäväksi. Luvussa kuvaillaan, miten menetelmät on toteutettu MATLAB-ohjelmistolla ja mitä tietoja niiden generoimista malleista kerätään menetelmien suorituskyvyn arviomiseksi. Luvun sisältämien pseudokoodien lisäksi käytettyjen MATLAB-funktioiden koodit ovat luettavissa Zenodo-palvelussa [17]. Testattujen RANSAC-menetelmien lisäksi luvussa kuvaillaan myös testaamiseen käytetty data.

4.1 Sovelletut RANSAC-menetelmät

4.1.1 RANSAC

Ensimmäisenä RANSAC-menetelmästä on toteutettu standardiversio, josta käytetään vastedes lyhennettä RANSAC. RANSAC-toteutuksessa otokset S_i valikoidaan satunnaisesti, otoksiin sovitetaan hypoteesimallit M_i pienimmän neliösumman menetelmän avulla ja malleille muodostetaan konsensusjoukot C_i koko havaintodatan D pohjalta. Suurimman konsensusjoukon C_{max} kerännyt malli M_{max} sovitetaan uudelleen konsensusjoukkoonsa pienimmän neliösumman menetelmällä. Lopetuskriteerinä käytetään kaavan 3.8 mukaista tarvittavien iteraatiokierrosten suorituspäättämää k silloin, kun se ei ylitä ohjelmalle asetettua enimmäiskierrosmäärää, jonka täytyessä ohjelma keskeyttää iteroinnin ja palauttaa siihen asti löydettyistä hypoteesimalleista parhaan mallin M_{max} . Maksimi-iteraatiomääränä menetelmiä testattaessa on käytetty 50 000 iteraatiokierrosta, jollei toisin ole erikseen mainittu. Iteraatiomäärän k alkuperäiseen estimointiin käytettävä hyvien havaintojen arvioitu osuus ϵ_e on 0,3 ja luottamustason η arvoksi on valittu 0,95.

Virhetoleranssitasolle t on käytetty arvoa 0,01 ja otoskoko m on 10 havaintopistettä. Molempien muuttujien optimaalisen arvon määrittäminen on jätetty työn tarkastelun ulkopuolelle. Otoskokoa m valittaessa suoritettiin kuitenkin joitakin lyhyitä testiajoja arvoilla $m = 5$, $m = 7$ ja $m = 10$. Tulosten perusteella valittiin viimeisin ja suurin arvo, vaikka se nostaakin kaavan 3.2.5 perusteella tarvittavaa iteraatiokierrosten k määrää pienempään arvoon verrattuna.

4.1.2 MSAC

RANSAC-kustannusfunktioista on modifioitu testattavaksi MSAC-kustannusfunktio, joka yhtälön 3.4 mukaisesti minimoi pelkän virhetoleranssitason t ulkopuolelle jäävien pisteiden lukumäärän lisäksi myös virhetoleranssitason t sisään jäävien pisteiden virhetermin neliöiden e_j^2 summan. MSAC-kustannusfunktiolla modifioidusta RANSACista puhutaan vastedes MSACina.

4.1.3 $T_{d,d}$ -testi

Luvussa 3.2.3 esiteltiin joitain otosten S_i esitestausten menetelmiä ennen kokonaisen konsensusjoukon C_i muodostamista hypoteesimallille M_i , ja näistä on toteutettu konseptuaalisesti yksinkertainen $T_{d,d}$ -testi. Testin otoskooksi on valittu suositusten mukaisesti 1 piste, joka poimitaan satunnaisesti koko havaintodatasta D ja jonka sopimista alle virhetoleranssitason t päähän testattavasta mallista tarkastellaan kriteerinä konsensusjoukon C_i muodostamiselle tai hypoteesimallin M_i hylkäämiselle. RANSACin standardiversiota, johon on lisätty $T_{d,d}$ -testi, kutsutaan jatkossa tekstissä $T_{d,d}$ -menetelmäksi.

4.1.4 Inner-RANSAC, Kt-menetelmä ja niiden yhdistelmä

Mallin uudelleensovittamiseksi sille löydettyyn konsensusjoukkoon on toteutettu luvussa 3.2.6 kuvatut inner-RANSAC-menetelmä ja iteratiivisesti skaalatun virhetoleranssitason menetelmä eli Kt-menetelmä. Inner-RANSACin otoskooksi on valittu 50 pistettä ja sen iteraatiolukumääräksi 10 iteraatiota. Skaalatun virhetoleranssitason menetelmässä muodostetaan ensin uudelleensovittavalle mallille M_i konsensusjoukko C_j pisteistä d , joiden etäisyys uudelleensovittavaan malliin on vähemmän kuin $Kt = 8t$. Tämän jälkeen konsensusjoukkoon C_j sovitaan uusi malli M_j käyttäen alkuperäistä mallia M_i Gauss-Newton-optimoinnin alkuarvauksena, ja operaatio toistetaan 3 iteraatiokierroksen ajan puolittamalla arvo K joka kierroksella niin, että muiden kierrosten virhetoleranssitason arvoiksi tulee $4t$, $2t$ ja lopulta alkuperäinen virhetoleranssitason arvo t . Inner-RANSACin ja Kt-menetelmän lisäksi kolmantena uudelleensovitusmenetelmänä testataan näiden kahden yhdistelmää, jossa jokaiselle inner-RANSAC-mallille, jonka konsensusjoukko on siihen astisessa inner-RANSAC-iteroinnissa paras, sovelletaan skaalatun virhetoleranssitason uudelleensovitusta. Toisin kuin Raguramin ym. [9] toteutuksessa inner-RANSACin ja skaalatun virhetoleranssitason yhdistelmästä, tässä työssä Kt-uudelleensovitusta sovelletaan vain niille inner-RANSAC-malleille M_i joiden konsensusjoukko C_i on siihen astisista malleista suurin, eikä kaikkien inner-RANSAC-otosten tuotamille malleille. Inner-RANSAC-uudelleensovitus, Kt-uudelleensovitus tai niiden yhdistelmä toteutetaan yleisestikin niille hypoteesimalleille M_i , joiden konsensusjoukkojen C_i suuruus ylittää siihen asti löydettyjen konsensusjoukkojen suuruudet. Näistä RANSAC-

versioista käytetään tulevassa käsittelyssä termejä inner-RANSAC, Kt-menetelmä ja inner-RANSAC-Kt-yhdistelmä.

4.1.5 LMS

Viimeinen toteutettu väline on pienin neliöiden mediaani (least median of squares, LMS), jota käytetään Gauss-Newton-optimoinnin kustannusfunktiona hypoteesimallien M_i soveltamiseksi dataan ja uudelleensovittamiseksi konsensusjoukkoon C_i . RANSACin standardiversioon verrattuna LMS-menetelmää siis käytetään pienimmän neliösumman menetelmän tilalla. LMS-sovituskunniolla modifioitua RANSACia kutsutaan jatkossa työssä nimellä LMS-menetelmä.

4.2 Sylinterien parametrisointi

Sylinterin karakterisoivat tässä työssä sylinterin säde $r \in \mathbb{R}^+$, sylinterin akselin kolmiulotteinen suuntavektori $\vec{a} \in \mathbb{R}^3$, sylinterin akselin piste $p_a \in \mathbb{R}^3$ ja sylinterin pituus $l \in \mathbb{R}^+$. Suuntavektorit \vec{a} ovat työssä yksikkövektoreita ja akselin piste p_a sijoittuu sylinterin pohjaan. Työssä on tarkasteltu yksinkertaisuuden vuoksi pituudeltaan 1 yksikön eli metrin mittaisia sylintereitä, eli 1 metrin mittaisten generoitujen sylinteripistepilvien lisäksi myös kokonaiset puunrungot on mallinnettu 1 metrin mittaisia peräkkäisinä sylintereinä.

4.3 Menetelmien toteutus MATLAB-ohjelmistolla

RANSAC-menetelmät, apufunktiot ja kaikki datan käsittely on tehty MATLAB-ohjelmistolla MATLAB-ohjelmointikielellä toteutettuna koodina. Kaksi työssä käytettyä pääohjelmaa ovat `ransac_runko` ja `ransac_sylinteri`, joista `ransac_runko` mallintaa kokonaisen rungon peräkkäisinä sylintereinä kutsumalla sylinterien sovittamiseksi `ransac_sylinteriä`. Funktioiden `ransac_runko` korkean tason toteutus on näkyvässä pseudokoodina 4.1, ja koko käytetty koodi on luettavissa Zenodo-palvelussa [17].

`Ransac_runko` argumentit ovat pistepilvimatriisi D , josta puunrunko halutaan mallintaa sekä pituus `cyl_h`, jonka mittaisina sylintereinä runko halutaan mallintaa. Ohjelma jakaa pistepilven D z-akselin suunnassa arvon `cyl_h` korkuisiin osiin ja kutsuu jokaiselle osalle funktiota `ransac_sylinteri`, joka sovittaa sylinterin saamaansa pistepilven osaan. `Ransac_runko` visualisoi `ransac_sylinterin` määrittelemät hyvät ja poikkeavat havainnot ja sovitettujen sylinteripinnan koordinaatistossa sekä puun halkaisijan graafina. Runkojen visualisointia on käytetty koodauksen apukeinona koodin toiminnan laadun valvontaan ja erittäin huonojen sylinterisovitusnopeaan tunnistamiseen. Kuvassa 4.1 on esimerkiksi `ransac_runko` tuottamasta puunrunkokuvasta. Kuvassa siniset pisteet edustavat sylinterisovituksessa hyviksi havainnoiksi määriteltyjä datapisteitä ja punaiset poikkeaviksi havainnoiksi määriteltyjä datapisteitä. Jos jonkin sylinterin sovitus tuottaa selkeästi puun-

ransac_runko 4.1. *ransac_runko-ohjelman pseudokoodi.*

```

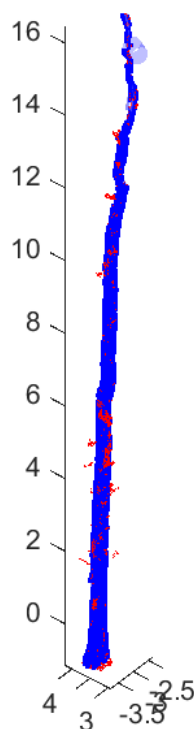
1 INPUT pistepilvi D, sylinterikorkeus cyl_h
2 OUTPUT Cyls, Ks, Costs, Epsilons, Anomals
3
4 SET RANSAC-parametrit param
5 INIT Cyls, Ks, Costs, Epsilons, Anomals
6 INIT puun halkaisijan graafi G
7 sylinterien lkm cyl_n := D:n korkeus / cyl_h
8 FOR i = 1:cyl_n
9     SET seuraava pistepilven mallinnettava osa D_section
10    [cyl, Cset, cost, k] := CALL ransac_sylinteri(D_section, param)
11    IF cyl eroaa liikaa alkuarvauksesta ja aiemmasta sylinterista cyl0
12        Anomals += 1
13    END IF
14    cyl0 := cyl
15    UPDATE Cyls(i), Ks(i), Costs(i)
16    Epsilons(i) := konsensusjoukon Cset koko / D_sectionin koko
17    SHOW Cset, D_section, cyl koordinaatistossa
18    UPDATE and SHOW graafi G
19 END FOR

```

rungosta poikkeavan sylinterin, sylinterin vaaleansininen pintamalli erottuu visualisaatiossa selvänä erillisenä sylinterinä. Kuvassa yksi tällainen poikkeava sylinteri näkyy aivan rungon yläosissa.

Ransac_runko palauttaa vektorina sovitetut sylinterit struct-tietotyypin olioina, jotka sisältävät sylinterin parametrit r , l , p_a ja \vec{a} , pistepilven pisteiden keskiarvoetäisyyden sylinteripinnasta eli muuttujan *mad* (eng. mean absolute distance) sekä tiedon siitä, suppeniko sylinterisovitus luotettavasti kohti ratkaisua ja olivatko Gauss-Newton-optimoinnissa käytettyjen matriisien kuntoluvut tarpeeksi pieniä, jotta ratkaisun voidaan sanoa olevan stabiili. Lisäksi ohjelma palauttaa vektoreina sylinterisovituksiin käytetyt RANSAC-iteraatiokierrosmäärät k , RANSAC-kustannusfunktioiden arvot F , hyvien havaintojen osuudet ϵ sylinterien pistepilvissä sekä lukumäärämuuttujan *an*, joka kertoo, kuinka moni sylinterisovitus joko poikkesi huomattavasti rungossa sitä edeltäneestä sylinteristä tai ei supennut luotettavasti kohti ratkaisua. Kriteerejä sylinterin poikkeavuudelle on kolme. Ensimmäisenä tarkasteltiin peräkkäisten sylinterien akselien \vec{a} välistä kulmaa, jonka arvon tulee olla alle 45° . Toisena kriteerinä on sylinterien säteiden r suuruuksien suhde, jonka tulee sijoitua avoimelle välille $(0,66; 1,5)$, eli uusi sylinterisäde saa poiketa edeltävän sylinterin säteestä alle kolmasosan verran. Viimeisenä poikkeavuuden kriteerinä tarkasteltiin uuden sylinterin alkupisteen p_a etäisyyttä sitä edeltävän sylinterin päätepisteeseen, ja tämän etäisyyden arvon tulee olla vähemmän kuin edeltävän sylinterin säteen r suuruus.

Ransac_runko antaa ransac_sylinteri-funktiolle sitä kutsuessaan parametreja, jotka ker-



Kuva 4.1. Esimerkki koodauksen apuna käytetystä visualisoinnista, jonka ransac_runko-ohjelma tuottaa.

tovat, millaista RANSAC-versiota rungon mallintamiseen halutaan soveltaa. Ransac_sylinteri-funktioon on toteutettu kaikki työssä käytettävät RANSAC-menetelmäversiot, ja funktiokutsusta riippuu, mitä versiota käytetään. Ransac_sylinterin korkean tason toteutus on esitetty pseudokoodina 4.2. Koodista voi lukea, että ransac_sylinterin toiminta riippuu siitä, mitä RANSAC-menetelmäversiota sen on funktiokutsun yhteydessä määritelty käytävän, ja se voi esimerkiksi kutsua eri apufunktioita sylinterimallien sovitukseen ja uudelleensovitukseen. Funktion toteutuksessa on tavoiteltu vaatimatonta modulaarisuutta siten, että haluttaessa siihen voitaisiin kohtuullisin panostuksin lisätä uusia modifikaatioita, kuten eri otoksen testausmenetelmiä tai hypoteesimallien uudelleensovitusfunktioita, ja että funktiokutsun yhteydessä olisi myös mahdollista säätää iteraatiokierrosmäärien ja otoskoon kaltaisia parametreja halutuiksi. Helppokäyttöisyyden vuoksi ainut pakollinen argumentti on kuitenkin pistepilvimatriisi, johon malli sovitetaan, ja muille RANSAC-parametreille on asetettu työssä toteutetun RANSACin standardiversion mukaiset oletusarvot.

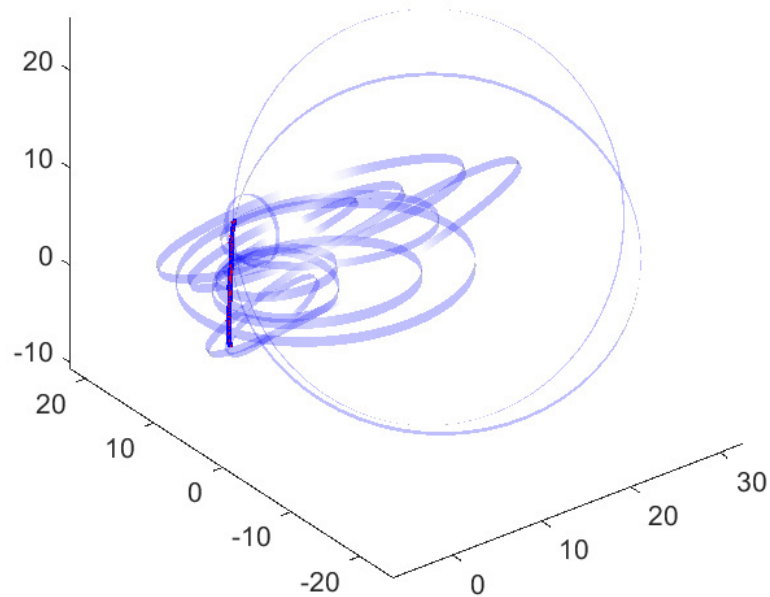
Ransac_sylinteri-funktiossa ei generoituja sylinteripistepilviä käsiteltäessä sovellettu sylinterien parametreja koskevia rajoitteita, joiden perusteella hypoteesimalleja M_i olisi RANSAC-iteroinnin aikana sylinterisovitusta tehdessä hylätty. Kokonaisia runkoja mallinnettessa funktioon kuitenkin lisättiin kohta, jossa tuotettua hypoteesimallia M_i verrattiin

ransac_sylinteri 4.2. ransac_sylinteri-ohjelman pseudokoodi.

```

1 INPUT pistepilvi D, (vapaaehtoinen: RANSAC-parametrit param)
2 OUTPUT cyl_best, Cset_best, cost_min, k
3
4 SET suurin sallittu iteraatioluku maxiter
5 SET RANSAC-parametrien oletusarvot
6 IF param
7     SET RANSAC-parametrit argumentin param perusteella
8 END IF
9 INIT cost_min, Cset_best, cyl_best
10 i := 0
11 WHILE i < k
12     SET D:n satunnaisotos sample
13     cyl := CALL parametrien sovitusfunktio fit(sample)
14     IF cyl ei suppene luotettavasti tai malli rikkoo rajoituksia
15         CONTINUE
16     ELSE
17         IF sampletest
18             test sample
19         END IF
20         IF sample = kelvollinen OR NOT sampletest
21             [Cset, cost] := CALL parametrien funktio cost(cyl)
22             IF cost < cost_min
23                 UPDATE cost_min, Cset_best, cyl_best, cyl0
24             END IF
25             IF refit
26                 cyl := CALL parametrien funktio refit(cyl)
27             END IF
28             IF cyl suppenee luotettavasti
29                 [Cset, cost] := CALL parametrien funktio cost(cyl)
30                 IF cost < cost_min
31                     UPDATE cost_min, Cset_best, cyl_best, cyl0
32                 END IF
33             END IF
34         END IF
35         UPDATE epsilon parhaan konsensusjoukon Cset_best koon perusteella
36         IF k NOT constant
37             UPDATE k epsilon perusteella
38         END IF
39     END IF
40 END WHILE
41 k := i

```



Kuva 4.2. Esimerkki runkomallien sovituksen tuloksista runkopistepilveen 1, jos koodissa ei sovelleta minkäänlaisia rajoitteita sovittaville sylintereille.

funktiolle sylinterisovituksen alkuarvaukseksi annettuun malliin M_0 eli rungon edelliseen sylinteriin. Hypoteesimalli M_i hylättiin ja sovitettiin uusi malli, jos mallin akselin \vec{a} kulma alkuarvaukseen eli edellisen sylinterin akseliin nähden oli 45° tai enemmän tai jos sen säde r oli pienempi kuin 50 % tai suurempi kuin 200 % edeltävän sylinterin säteeseen verrattuna. Ensimmäistä runkoon sovittavaa sylinteriä verrattiin edeltävän sylinterin puuttuessa sylinterisovituksen optimoinnin alkuarvaukseen. Rajoitteet lisättiin koeajon perusteella, jossa runkosovitusta kokeiltiin tehdä yhden laserskannauksen tuottamaan runkopistepilveen 1, sillä tulokset osoittautuivat muuten käyttökelvottomiksi. Kuvassa 4.2 näkyy esimerkki epäonnistuneista sylintereistä, joita runkopistepilveen 1 sovitettiin ilman rajoitteita.

Ransac_sylinterin kutsumat funktiot `least_squares_cylinder` pienimmän neliösumman sovitukseen ja `least_median_of_squares_cylinder` pienimmän neliömediaanin sovitukseen käyttävät sylinterimalliparametrien löytämiseen iteratiivista Gauss-Newton-optimointia, ja ne on saatu käyttöön ulkopuolelta. Optimoinnin alkuarvauksena toimivat yksittäisiä sylintereitä ja puunrunkojen 1. sylinteriä mallinnettaessa 1. RANSAC-iteraatiokierroksella sylinteriparametrit, joissa säteen r arvo oli 0,2, akselin suuntavektori \vec{a} oli z-akselin suuntainen yksikkövektori ja akselin aloituspiste p_a oli origossa. RANSAC-iteroinnin edetessä paremman hypoteesimallin M_i löytyessä sitä käytettiin jatkossa alkuarvauksena. Ran-

msac_consensus_cylinder 4.3. RANSAC-konsensusjoukot ja MSAC-kustannusfunktion arvot muodostava apufunktio.

```

1 INPUT pistepilvi D, sylinterimalli cyl, virhetoleranssitaso t
2 OUTPUT konsensusjoukko Cset, MSAC-kustannusfunktion arvo cost
3
4 pisteiden erot sylinteriakseliin: dist_axis :=
5     CALL distances_to_line(D, cyl akseli, cyl alkupiste)
6 pisteiden erot sylinteripintaan: dist_cyl := dist_axis - cyl sade
7 Cset := D:n pisteet joille dist_cyl < t
8 outlier_cost := t^2 * Csetin ulkopuolisten pisteiden lkm joukossa D
9 inlier_cost := sum(dist_cyl^2) joukon Cset pisteille
10 cost := outlier_cost + inlier_cost

```

Tdd_test_cylinder 4.4. $T_{d,d}$ -testin suorittavan apufunktion pseudokoodi.

```

1 INPUT otoskoko m, sylinterimalli cyl, pistepilvi D, virhetoleranssitaso t
2 OUTPUT sampleok
3
4 sample := m-kokoinen satunnaisotos pistepilvimatriisista D
5 pisteiden erot sylinteriakseliin: dist_axis :=
6     CALL distances_to_line(D, cyl akseli, cyl alkupiste)
7 pisteiden erot sylinteripintaan: dist_cyl := dist_axis - cyl sade
8 IF dist_cyl < t
9     sampleok := 1
10 ELSE
11     sampleok := 0
12 END IF

```

sac_sylinterin kutsumat muut apufunktiot ovat msac_consensus_cylinder, Tdd_test_cylinder ja scaled_t_refit_cylinder, jotka on kirjoitettu työn toteutusta varten. Ensimmäinen muodostaa konsensusjoukon ja kustannusfunktion arvot RANSAC- ja MSAC-menetelmissä, ja sen pseudokoodi on näkyvillä ohjelmana 4.3. Toinen suorittaa $T_{d,d}$ -testin ja on kuvattu pseudokoodina 4.4. Kolmas apufunktio suorittaa luvussa 4.1 kuvatun Kt-uudelleensovitusmenetelmän ja on kuvattu pseudokoodina 4.5. Kaikkien kolmen apufunktion täydet koodit ovat luettavissa Zenodo-palvelussa [17].

4.4 Menetelmien arviointiin käytetty data

Kuten luvussa 2 kerrotaan, RANSAC-menetelmiä käytetään työssä sylinterien sovittamiseen sekä oikeista puista laserkeilaamalla saatuun pistepilvidataan että keinotekoisesti generoituihin sylinteripistepilviin. Pisteiden koordinaattien mittayksikkönä on metri. Laserkeilauksen tuloksena saatu data on peräisin keskikokoisesta, noin 17 metrin korkuisesta tammesta. Puun pistepilvestä on etukäteen segmentoitu pois suurin osa oksista lähtöisin

scaled_t_refit_cylinder 4.5. Kt-uudelleensovituksen suorittavan apufunktion pseudokoodi.

```

1 INPUT pistepilvi D, alkuarvaus cyl0 , virhetoleranssitaso t
2 OUTPUT sylinterimalli cyl
3
4 t_scaled := 8 * t
5 cyl := cyl0
6 WHILE t_scaled greater or equal to t
7     Cset := CALL msac_consensus_cylinder(D, cyl , t_scaled)
8     IF Cset > 0
9         cyl := CALL least_squares_cylinder(Cset, cyl)
10    ELSE
11        RETURN
12    END IF
13    t_scaled := t_scaled / 2
14 END WHILE

```

olevista pisteistä, mutta se sisältää yhä runkoon kiinnittyneiden oksien tynkiä ja irtonaisten oksien kappaleita. Puunrunkodatan latvus on haasteellinen mallinnettava, sillä latvaa kohti mentäessä pisteiden määrä vähenee, rungon rakenne ohenee ja monimutkaistuu ja rungon päähaara tulee vaikeaksi määrittää sen haarautuessa latvukseksi. Rungolle muodostetun vertailumallin pohjalta työn tarkasteluun on otettu alimmat 14 metriä, ja muutama ylin metri alkuperäisestä runkopistepilvestä jätetään mallinnusmenetelmien tuloksia tarkasteltaessa huomiotta.

Rungosta on saatu käyttöön kolme pistepilveä, joista pistepilvi numero 1 kattaa kaikki laserkeilaamalla saadut datapisteet. Siitä on muodostettu pistepilvet numero 2 ja 3 alaspäin interpoloimalla (downsampling) eli harventamalla niin, että pistepilven 2 resoluutio on 1 pistettä per 2 cm² eli 0,5 pistettä per 1 cm² ja pistepilven 3 resoluutio 1 pistettä per 4 cm² eli 0,25 pistettä per 1 cm². Täyden pistepilven numero 1 tarkka resoluutio ei ole tiedossa, mutta pistepilvien 1, 2 ja 3 havaintopisteiden lukumäärien suhteiden perusteella sen resoluutioksi arvioidaan noin 3–4 pistettä per 1 cm². Tiheimmän pistepilven 1 resoluutio arvioidaan siis yli kymmenkertaiseksi harvimpaan pistepilveen 3 verrattuna. Pistepilvestä 1 on tuotettu menetelmien vertailua varten puunrungolle mahdollisimman tarkka malli RANSAC-menetelmän perusversiolla, huomattavan suurella sallitulla maksimi-iteraatiokierrosmäärällä $k = 200\,000$.

Puunrunko on kuvattu koko matkalta sen ympäriltä 6 eri skannauspisteestä. Sovitusmenetelmien haastamiseksi pistepilvistä numero 1 ja 3, eli suurimman ja pienimmän resoluution pistepilvistä, muodostettiin osittaisia versioita, joissa vain yksi tai kaksi skannauspisteistä on käytössä eli joko skannauspiste 1 tai skannauspisteet 1 ja 3. Näin on kahdella eri resoluutiolla ja kahdella eri mittapisteiden määrällä saatu yhteensä 4 kappaletta runkopistepilviä, joiden tiedot on koottu taulukkoon 4.1. Tarkoituksena on arvioida, voidaanko

vain yhtä skannausta ja pientä resoluutiota ja näin ollen mahdollisimman pientä datajoukkoa käyttämällä saada hyviä tuloksia, ja tuottaako joko toisen skannauksen lisääminen, resoluution nostaminen tai näistä molempien toimeenpano parempia mallinnustuloksia.

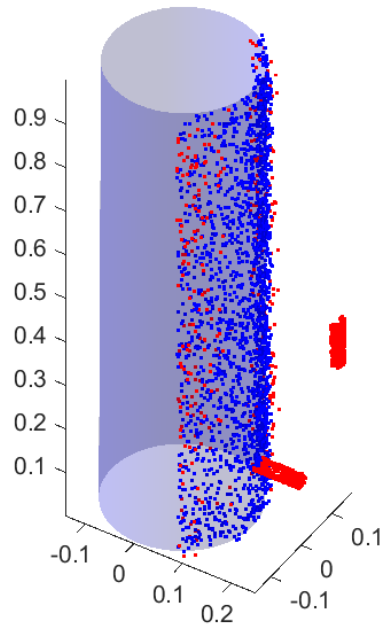
Taulukko 4.1. Puunrunkoa kuvaavien pistepilvien tiedot. Runkopistepilvet 3 ja 4 sisältävät pisteitä useammasta skannauspisteestä kuin runkopistepilvet 1 ja 2. Pistepilvien 2 ja 4 resoluutio on suurempi kuin pistepilvien 1 ja 3.

runkopistepilvi	resoluutio, piste/cm ²	laserskannauksia	datapisteiden lukumäärä N
R1	0,25	1	1 794
R2	3–4	1	124 381
R3	0,25	2	2 358
R4	3–4	2	210 627

Kahdeksi skannauspisteeksi on valittu skannauspisteet 1 ja 3, jotka eivät ole vierekkäiset mutta jotka sijoittuvat silti alle 180 asteen kierron päähän toisistaan puun ympärillä. Vierekkäisiä skannauspisteitä ei valittu, sillä oletuksen mukaan kauempana toisistaan sijaitsevat skannauspisteet tuottavat enemmän tietoa mallin muodostamiseksi kuin aivan vierekkäiset, kun havaintopisteitä on käytössä laajemmalta alueelta puun ympäriltä. Toisaalta toisinaan voi esiintyä tilanteita, joissa kuvauksen kohdetta voidaan lähestyä vain yhdeltä puolelta eikä kuvauksen kohteen ympäri päästä kulkemaan. Täten kahdeksi eri skannauspisteeksi ei valittu myöskään aivan vastakkaisia vaan hieman toisiaan lähempänä olevat skannaussuunnat.

Kokonaisten puunrunkojen mallintamisen lisäksi Matlabilla on generoitu 2 osittaisen sylinteripinnan pistepilvet. Vain kolmasosa sylinteripinnasta on katettu pisteillä, eli pisteitä on 120° matkalta sylinterin ympäriltä. Osittainen sylinteripistepilvi jäljittelee tilannetta, jossa puunrunkoa ei ole laserkeilattu joka suunnasta vaan dataa on kerätty vain yhdeltä puolelta puuta. Generoidut sylinteripistepilvet sisältävät laserkeilausdatan poikkeavien havaintojen jäljittelemiseksi simuloidun oksantyingän pisteitä sekä yhden sylinteristä kokonaan erilleen sijoittuvan, simuloidun oksantyingän kanssa samankokoisen klusterin havaintopisteitä esimerkkikuvan 4.3 mukaisesti. Kuvassa pistepilveen on sovitettu sylinteripintamalli, josta poikkeavat pisteet näkyvät punaisena. Mallin mukaiset hyvät pisteet näkyvät kuvassa sinisinä.

Molempien sylinterien akselin suuntavektorit \vec{a} ovat z-koordinaattiakselin suuntaisia yksikkövektoreita, akselin alkupisteet p_a sijaitsevat origossa ja sylinterien pituus l on 1. Kohinan tasoa ja sen kautta ϵ -arvoa on varioitu. Datapisteiden säteisiin tuotettiin MATLABin pseudosatunnaisia lukuja generoivien funktioiden `rand` ja `randn` avulla normaalisti jakautunutta kohinaa. Kohinan keskiarvon mittaluokka oli 0,0001 eli joitakin millimetrien



Kuva 4.3. Sylinterin osittainen pistepilvi, simuloitu oksantynkä ja irrallinen oksankappale. Pistepilveen on sovitettu sylinteripinta, ja punaiset pisteet ovat sylinteripintamallista poikkeavia havaintoja.

kymmenysosia, ja keskihajonnan mittaluokka sylinteripistepilvelle 1 oli 0,001 mittayksikköä eli joitakin millimetrejä ja sylinteripistepilvelle 2 0,01 mittayksikköä eli joitakin senttimetrejä. Näin hyvien havaintojen osuudeksi ϵ pistepilvissä on työssä käytetyllä virhetoleranssitasolla $t = 0,01$ saatu noin 0,63 sylinteripistepilvelle 1 ja 0,36 sylinteripistepilvelle 2. Sylinteripisteiden kohina on suurempaa generoitujen osittaisten sylinteripistepilvien reunoilla kuin sylinterisegmenttien keskellä laserkeilaustilanteen jäljittelemiseksi; lasersäteet heijastuvat puusta takaisin laserskanneriin suurimmalla intensiteetillä ja pienimmällä epävarmuudella laserskannerista katsottuna keskeltä puunrunkoa, kun taas puun reunoilta takaisin heijastuvien säteiden eli mittapisteiden epävarmuus on suurempaa. Generoitujen sylinterien parametrit ja pistepilvien tiedot on koottu taulukkoon 4.2.

Taulukko 4.2. Generoitujen sylinteripistepilvien tiedot. Tulokset esitetty 3 merkitsevän numeron tarkkuudella.

sylinteri	r , m	l , m	resoluutio, piste/cm ²	ϵ	mad, m	F
Sylinteri 1	0,15	1,00	0,25	0,634	0,0291	603
Sylinteri 2	0,15	1,00	0,25	0,363	0,0348	1050

4.5 Arvioitavat muuttujat

RANSAC-versioissa, joihin on yhdistetty $T_{d,d}$ -testi tai inner-RANSACin ja skaalatun virhetoleranssitason uudelleensovitus eli Kt-menetelmä, tarkastellaan ensisijaisesti sitä, muuttuuko tarvittava iteraatiokierrosmäärä k RANSACin perusversioon verrattuna. Muiden menetelmien kohdalla tarkastellaan ensisijaisesti tulosten tarkkuutta. Jokaista toteutettua menetelmää on arvioitu generoitujen sylinteripistepilvien numero 1 ja 2 avulla antamalla kunkin menetelmän suorittaa 30 sovitusta molempiin sylinteripistepilviin ja keräämällä talteen näiden 30 suorituksen mediaanit, aritmeettiset keskiarvot ja keskihajonnat keskeisistä suorituskykymuuttujista molempien sylinterien osalta. Tarkasteltavia muuttujia ovat menetelmän tuottaman sylinterimallin M_{max} säde r , sylinterin pituus l , sylinterin tilavuus V , sylinterimallin akselin suuntavektorin \vec{a} kulma $\angle_{\vec{a}}$ z-akselin kanssa, sylinterin akselin alkupisteen p_a etäisyys $|p_a|$ origosta, pistepilven D pisteiden keskivertoetäisyys muodostetun sylinterimallin M_{max} pintaan eli mad (engl. mean absolute distance), menetelmän löytämä hyvien havaintojen osuus ϵ pistepilvidatassa, RANSAC-iteraatiokierrosmäärä k ja kustannusfunktion arvo F .

Kaikkia toteutettuja menetelmiä on testattu myös luvussa 4.4 kuvattuihin puunrunkopistepilviin 1, 2, 3 ja 4. Tarkasteltavia muuttujia ovat pistepilven D pisteiden keskiarvoetäisyys sovitettuun sylinterimalliin eli mad , löydetty hyvien havaintojen osuus ϵ , RANSAC-iteraatiokierrosmäärä k , RANSAC- tai MSAC-kustannusfunktion arvo F , rungon tilavuus V , rungon mallintamisessa ilmenneiden poikkeavien sylintereiden lukumäärä an sekä mallintamiseen kuluva koodin suoritus aika. Runkomallit muodostettiin jokaisen menetelmän avulla jokaisesta runkopistepilvestä 30 kertaa, ja jokaiseen muodostuneeseen runkomalliin käytetyistä sylintereistä otettiin talteen mediaaniarvot edellä luetelluista muuttujista mad , ϵ , k ja F . Jokaiselle menetelmälle on ilmoitettu näistä mediaanimuuttujista mediaanit, keskiarvot ja keskihajonnat näiden 30 runkomallin kesken. Mediaanit, keskiarvot ja keskihajonnat 30 runkomallin kesken on ilmoitettu suoraan muuttujista an ja V . Tilastollisten tunnuslukujen laskennassa on jätetty huomiotta mahdolliset NaN-arvot, joita käytetyt funktiot saattavat palauttaa mallinnuksen epäonnistuessa täysin.

Koska tarkasteltavat muuttujat ilmoitetaan tilastollisina tunnuslukuina useiden suorituskertojen ajalta, muuttujien arvot eivät välttämättä välitä todellista kuvaa siitä, miten yksittäiset sylinterisovitukset ovat onnistuneet kunkin menetelmän osalta. Vaikka esimerkiksi jonkin tarkastellun menetelmän tuottamat sylinteripituus l ja säde r jollekin sylinterille olisivat molemmat keskiarvoina mitattuna oikeansuuruisia, on mahdollista, että ne olisivat peräisin eri sylinterimalleista, ja että todellisuudessa menetelmä ei olisi onnistunut sovittamaan yhtään kokonaan oikeankaltaista sylinteriä vaan sylintereitä, jotka olisivat oikean mittaisia mutta säteeltään väärän suuruisia ja toisaalta sylintereitä, jotka olisivat säteeltään oikean paksuisia mutta väärän mittaisia. RANSAC-menetelmien satunnaisen luonteen vuoksi keskiarvoja ja hajontalukuja tarkastelemalla pyritään kuitenkin tekemään

mahdolliseksi useamman suorituskerran keskiarvoisten tulosten tarkastelu robustiuden lisäämiseksi tavalla, joka on vielä helposti kvantifioitavissa ja esitettävissä. Generoitujen sylinterimallien tilavuuksien V tarkastelu muiden sylinteriparametrien lisäksi sekä kokonaisia runkoja mallinnettaessa rungon tilavuuden V tarkastelu yksittäisten sylinterien parametrien sijaan toivon mukaan vähentää edellä kuvatun kaltaisia ongelmia ja antaa paremman kokonaiskuvan mallien onnistumisesta.

Menetelmien tehokkuutta arvioidaan suoritettujen iteraatiokierrosten k määrän perusteella. Lisäksi tuloksia analysoitaessa huomioidaan se, millaisia operaatioita iteraatiokierrosten aikana suoritetaan eli kuinka laskennallisesti raskaita eri menetelmien suorittamat iteraatiokierrokset ovat toisiinsa verrattuna. MATLAB-ohjelmiston tarjoamien koodianalyysityökalujen mukaan työssä toteutettujen RANSAC-menetelmien laskennallisesti kallein askel on selvästi mallien sovittaminen pistepilveen pienimmän neliösumman tai pienimmän neliömediaanin menetelmällä sekä Gauss-Newton-optimoinnilla. Koodin muut osat käyttävät mallisovitukseen verrattuna laskenta-aikaa marginaalisen vähän. Tarkkojen absoluuttisten suoritusajojen käsittely päätettiin jättää tekemättä, sillä suoritusajoihin voivat vaikuttaa monet paikalliset ohjelmointiympäristöstä riippuvat tekijät ja lisäksi MATLAB-ohjelmiston toimintaan liittyvät tuntemattomat ominaisuudet, jotka testiajojen perusteella vaikuttavat mahdollisesti nopeuttavan funktioiden toimintaa useiden peräkkäisten funktiokutsujen suorittamisen yhteydessä. Toteutettujen ohjelmien tyypillisistä suoritusajoista annetaan kuitenkin suuntaa-antavat arviot, jotka kertovat, oliko mallinnuksiin kuluva aika tyypillisesti sekunteja, minuutteja vai tunteja.

5. TULOKSET

Luvussa 4 kuvailtujen RANSAC-versioiden tuottamat tulokset sekä generoidulle sylinteridatalle että käytössä olleelle puunrunkojen laserkeilausdatalle esitetään ja kuvaillaan tässä luvussa. Tarkat tulokset on esitetty taulukoimalla, ja tekstissä taulukoiden sisältöä avataan ja tuloksista tehdään keskeisiä poimintoja. Tuloksia jokaisen tarkasteltavan muuttujan suhteen käydään läpi tekstissä siinä järjestyksessä, kuin ne on taulukoissa esitetty, ja tämän jälkeen tiivistetään tulosten keskeinen sisältö.

5.1 Menetelmien suoriutuminen generoidulla sylinteridatalla

Sovellettujen RANSAC-menetelmien tulosten mediaani, keskiarvo ja keskihajonta 30 suorituskerralta luvussa 4.4 esitellyille sylinteripistepilville 1 ja 2 on koottu taulukkoihin 5.1 ja 5.2. Luvut on ilmoitettu kolmen merkitsevän numeron tarkuudella. Taulukko 5.1 sisältää menetelmien sovittamien sylinterimallien M_{max} parametrin, ja taulukko 5.2 sisältää RANSAC-menetelmien suoriutumista kertovia tunnuslukuja. Molempien taulukoiden vasemmanpuoleinen sarake sisältää tarkasteltavien muuttujien tunnuksia sekä tiedot siitä, minkä sylinterin pistepilvidatalla kullakin rivillä esiintyvät muuttujat ja arvot on saatu. Molempien taulukoiden oikeanpuolimmaisessa sarakkeessa on annettu muuttujien vertailuarvot, joita lähellä tulosten keskiarvon ja medianin tulisi olla. Sylinteriparametrien vertailuarvot taulukossa 5.1 ovat sylinterien pistepilvien generointiin käytettyjen sylinterimallien todelliset parametrin, ja RANSAC-tunnuslukujen vertailuarvot taulukossa 5.2 ovat sylinterien ja niiden pistepilvien todellisten tietojen pohjalta laskettuja arvoja.

Yksittäisen RANSAC-menetelmäversion suoriutumista sylinteripistepilvien 1 ja 2 kohdalla voidaan tarkastella valitsemalla taulukon yläriviltä menetelmäversion sarake, valitsemalla vasemmasta reunasta tarkasteltavan muuttujan rivi ja vertailemalla tunnuslukuja sylinterien 1 ja 2 riveillä toisiinsa ja taulukon oikeassa reunassa esitettyyn vertailuarvoon. Eri RANSAC-menetelmien suoriutumista suhteessa toisiinsa voidaan tarkastella valitsemalla taulukkorivi, jolla halutun muuttujan arvo esitetään, ja vertailemalla rivin soluja toisiinsa ja oikeanpuolimmaisesta sarakkeesta vertailuarvoon. Taulukoissa käytetyt sylinterien 1 ja 2 pistepilvien tarkemmat tiedot sisältyvät lukuun 4.4, mutta keskeisintä on sylinterien 1 ja 2 pistepilvien hyvien havaintojen osuuksien ϵ ero. Sylinterille 1 ϵ on 0,634 ja sylinterille 2 ϵ on matalampi, 0,363.

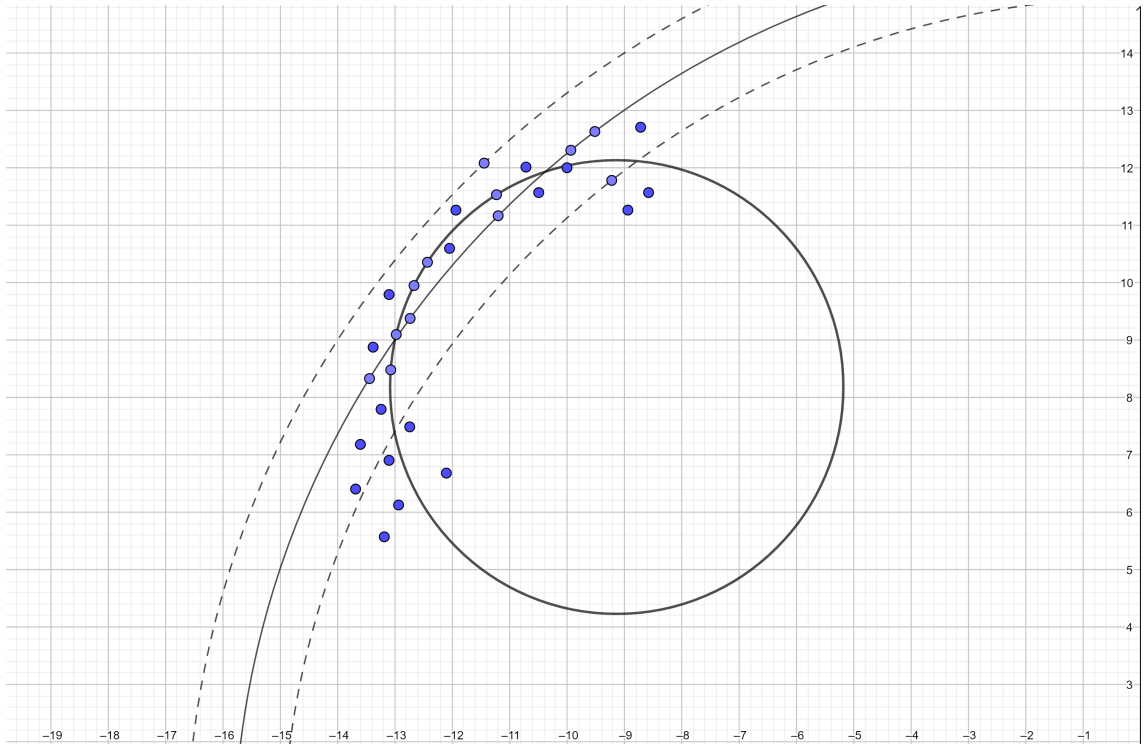
Kuten luvussa 4.5 kerrottu, työssä ei käsitellä tarkkoja absoluuttisia suoritusajkoja. Havaintojen perusteella menetelmien testauksen aikana yhden runkomallin sovittamisen voidaan kuitenkin arvioida kestäneen yleensä joitakin minutteja, tyypillisesti noin 2–5 minuuttia. Yksittäisen sylinterin sovittamisen generoituun pistepilveen kesti haastavamman sylinteripistepilven 2 kohdalla yleensä maksimissaan minuutin ja helpomman sylinteripistepilven 1 tulokset olivat valmiina lähes välittömästi.

5.1.1 RANSAC-versioiden tuottamat sylinteriparametrit

Taulukko 5.1 ilmoittaa järjestyksessä mediaanit, keskiarvot ja keskihajonnat muuttujille, joiden avulla voidaan arvioida menetelmien tuottamien sylinterisovitususten M_{max} onnistumista. Tarkastelun alaiset sylinteriparametrit ovat sylinterimallin säde r , sylinterin pituus l , sylinterin akselin \vec{a} alkupisteen p_a etäisyys origoon $|p_a|$ sekä sylinterin akselin \vec{a} ja Z -akselin välinen kulma $\angle_{\vec{a}}$ asteina. Taulukon viimeinen muuttuja mad (engl. mean absolute distance) kertoo sylinterin pistepilven pisteiden keskiarvoisen euklidisen etäisyyden menetelmien sovittamien sylinterimallien M_{max} pintaan.

Säteen r osalta sylinterin 1 sovituksessa parhaat mediaanisäteet tuottivat inner-RANSAC, inner-RANSAC-Kt-yhdistelmä sekä $T_{d,d}$ -menetelmä. Niiden mediaanisäteiden ero todelliseen oli alle prosentin verran. Muidenkin menetelmien tuottamat mediaanisäteet sylinterille 1 erosivat todellisesta säteestä maksimissaan noin 7 %. Kaikki mediaanisäteet sylinterille olivat todellista 0,15 suuruista sädettä suurempia, eivät koskaan pienempiä. Sylinterin 2 kohdalla taas RANSAC sekä sen LMS-sovituskäytännöllä, $T_{d,d}$ -testillä tai Kt-uudelleensovitusmenetelmällä varustetut versiot sovittivat mediaanisäteeltään maksimissaan 5 % todellista arvoa pienemmät sylinterit. Muiden menetelmien tuottamat sylinterit pistepilveen 2 olivat mediaanisäteeltään kymmeniä kertoja liian suuria. Säteiden keskiarvot ja keskihajonnat olivat säteen r todelliseen arvoon 0,15 verrattuna kymmeniä kertoja suurempia, molempien sylinteripistepilvien kohdalla, kaikille menetelmille. Tuotettujen sylinterien jakaumat ovat siis oikealle vinoja, ja niissä painottuvat liian suuret sylinterisovitukset. Pienimmät eli parhaat keskiarvot ja keskihajonnat tuotti $T_{d,d}$ -menetelmä ja suurimmat eli huonoimmat taas LMS-menetelmä.

Käytetty virhetoleranssitaso $t = 0,01$ on todelliseen sylinterisäteeseen $r = 0,15$ verrattuna melko suuri. Tämä on saattanut johtaa suurten konsensusjoukkojen C_i muodostumiseen suurille sylinterimalleille, joiden pinnasta suuri osa levittäytyy halkomaan sylinterien 1 ja 2 osittaisia pistepilviä. Tällaista mahdollista tilannetta havainnollistetaan äärimmäisen esimerkin avulla kuvassa 5.1, jossa käytetty virhetoleranssitaso t on liioitellun suuri. Kuvassa näkyvä ympyrä kuvaa todellisen sylinterin poikkileikkausta ja kuvan pisteiden joukko poikkileikkausta sen sylinteripistepilvestä. Kuvaan mahtuu näkyviin vain osittaisena paljon suuremman ympyrän kaari, joka on osa pistepilveen sovitetun liian suuren hypoteesimallisyylinterin M_i poikkileikkausta. Kuvaan on merkitty katkoviivoilla virhetolerans-



Kuva 5.1. Kärjistetty esimerkki siitä, miten liian suuri virhetoleranssitaso t edistää säteeltään r liian suurten sylinterien sovittamista sylinteripistepilviin. Kuvan pienempi ympyrä on todellisen sylinterin poikkileikkaus, ja suuremman ympyrän kaaren rinnalla kulkevat katkoviivat näyttävän virhetoleranssitason t .

sitasoalue, jonka sisään jäävät pisteet muodostavat hypoteesimallin M_i konsensusjoukon C_i . Kuvassa näkyy, että liian suuren hypoteesimallin loiva kaari ulottuu kulkemaan koko pistepilven poikki, toisin kuin liian pienen hypoteesimallin kaari ulottuisi, jolloin sopivaan kohtaan asettuva suuri hypoteesimalli saattaa saada kerättyä konsensusjoukkoonsa C_i suuren osan pistepilven pisteistä. Ilmiötä saattaa voimistaa sylinteripistepilvien pisteiden suurempi kohina eli pisteiden suurempi hajautuminen pistepilven reunoilla luvussa 4.4 kuvattuun tapaan.

Sylinteripituudet l osuivat sädettä paremmin oikeaan arvoon 1. Sylinterin 1 kohdalla mediaanipituudet olivat hyviä kaikille menetelmille, huonoimmillaan inner-RANSACille ja inner-Kt-yhdistelmälle 6 % ja 4 % liian pieniä mutta muilla menetelmillä alle 1 % liian pieniä. Toisin kuin säteen r kohdalla, sylinterille 1 keskiarvotkin olivat kohtalaisia, mutta useamman menetelmän osalta silti yli 10 % liian pieniä. Myös keskihajonnat olivat säteen r keskihajontoja kohtuullisempia ollen keskimäärin 15 % ja maksimissaan 25 % todellisen pituuden arvosta. Sylinterin 2 kohdalla MSAC oli ainoa menetelmä, jonka tuottama mediaanipituus erosi alle 10 % todellisesta, ja eniten todellisesta eroavan mediaanisäteen sylinterille 2 tuotti inner-RANSAC, jonka mediaanisylinteri oli yli 20 % liian lyhyt. Muille menetelmille ero oli noin 15 %. Parhaat keskiarvot ja keskihajonnat sylinteripituuksissa tuotti MSAC ja huonoimmat säteen r tapaan LMS-menetelmä.

Menetelmien tuottamien sylinteritilavuuksien V arvoissa näkyy se, millaisia arvoja menetelmät tuottivat sylinterisäteelle r . Säteiden r arvojen perusteella odotetusti tarkimpia tilavuuksia V olivat mediaaniltaan löytäneet sylinterille 1 inner-RANSAC-Kt-yhdistelmä, jonka mediaanitilavuuden ero todelliseen arvoon oli alle 1 %, sekä $T_{d,d}$ ja inner-RANSAC, joiden tuottamien mediaanitilavuuksien ero todelliseen oli alle 3 %. Kt-menetelmän tuottaman mediaanitilavuuden poikkeama sylinterille 1 oli säteen r tapaan menetelmistä suurin, tilavuuden ollessa noin 15 % liian suuri. Muiden menetelmien mediaanitilavuudet sylinterille 1 olivat 5-10 % liian suuria. Sylinterin 2 kohdalla RANSACin tuottama mediaanitilavuus ylsi noin 10 prosentin päähän ja LMS-, $T_{d,d}$ - sekä Kt-menetelmien alle 15 prosentin päähän todellisesta arvosta, muiden menetelmien mediaaniarvojenkin ollessa yli 100 kertaa liian suuria. Keskiarvot ja keskihajonnat olivat säteen tapaan erittäin suuria, pienimmilläänkin kymmeniä kertoja ja suurimmillaan kymmeniä tuhansia kertoja todellisen tilavuuden V verran. Pienin keskiarvo ja keskihajonta oli sylinterin 1 kohdalla Kt-menetelmällä ja sylinterin 2 kohdalla $T_{d,d}$ -menetelmällä. Suurimmat keskiarvot ja keskihajonnat molemmille sylintereille tuotti jälleen LMS-menetelmä.

Aloituspisteen etäisyys origoon $|p_a|$ oli sylinterille 1 mediaaniltaan pieni kaikkien muiden paitsi inner-RANSACin kohdalla. Sen tuottamien sylinterien aloituspisteen mediaanietäisyys origoon oli noin 0,05, mikä on jo yli 30 % sylinterin säteestä r . Inner-RANSAC-Kt-yhdistelmän aloituspisteen etäisyys origoon pyöristyy myös arvoon 0,02, kun lopuille menetelmille arvo pyöristyy lukemaan 0,01. Sylinterille 2 RANSAC ja $T_{d,d}$ sijoittivat sylinterit mediaanina mitattuna keskimäärin noin 0,10 päähän origosta ja LMS sekä Kt-menetelmä noin 0,15 päähän, mikä on jo sylinterin säteen suuruinen heitto. Keskiarvot ja keskihajonnat ovat useita metrejä kaikille menetelmille ja molemmille sylintereille, mutta suurimmat LMS-menetelmän kohdalla. Edellä nähtiin, että sovitetut sylinterimallit olivat usein säteeltään hyvin suuria, ja koska suurten sylinterimallien pinta asettuu sylinteripistepilvien tuntumaan, sijoittuu suuren sylinterimallin keskipiste kauas todellisen sylinteripistepilven keskipisteestä.

Sylinterisovitusten akselin \vec{a} kulma z-akselin kanssa on mediaanein mitattuna sylinteripistepilven 1 kohdalla keskimäärin alle asteen verran kaikille menetelmille. Sylinteripistepilvelle 2 sovitusten akselin kulma pyöristyy 1 asteeseen RANSACin, LMS-menetelmän, $T_{d,d}$ -menetelmän ja Kt-menetelmän kohdalla, mikä myös on tuloksena hyvä. Inner-RANSAC ja inner-RANSAC-Kt-yhdistelmä antavat kohtalaiset mediaanitulokset, 4° ja 5° . Poikkeuksen menetelmien joukossa tekee LMS-menetelmä suurella 20 asteen mediaanikulmalla. Keskiarvot ovat 20° – 50° ja keskihajonnat mediaaneja suurempia molemmille sylintereille. Suurimman keskihajonnan molemmille sylintereille tuotta inner-RANSAC-Kt-yhdistelmä ja suurimmat keskiarvot sylintereille 1 ja 2 inner-RANSAC ja LMS-menetelmä.

Pisteiden keskiarvoetäisyyden mad kaikki tunnusluvut ovat sylinteripistepilvelle 1 arvoltaan alle 10 mm, eli kaikkien menetelmien mediaani, keskiarvo ja keskihajonta mitataan millimetreissä, kun sylinterimallin säde on 0,15 metrin suuruinen. Sylinterin 2 kohdalla

jotkin arvot nousevat hieman 10 millimetrin yläpuolelle. Suurimmat keskiarvot ja keskihajonnat molemmille sylintereille on tuottanut LMS-menetelmä.

Yleisesti liian leveät sylinterit olivat ongelma erityisesti sylinteripistepilven 2 kohdalla, jonka ϵ -arvo oli sylinteriä 1 matalampi, vain 0,363. Sovitukset sylinteripistepilven 1, jonka ϵ -arvo oli 0,634, onnistuivat pääosin hyvin mediaaniarvoja tarkasteltaessa, mutta hajonta tämänkin pistepilven sovitustuloksissa oli huomattavan suurta ja suuret sylinterit vahvasti edustettuina. LMS-menetelmä tuotti menetelmistä eniten hajontaa tuloksissa jokaisen muuttujan osalta tuottamatta merkittävästi lisäarvoa muihin menetelmiin verrattuna missään muussakaan suhteessa. RANSACin perusversio menestyi sovituksissa tasaisen hyvin: Sen tulokset olivat helpomman sylinterin 1 kohdalla kohtalaisia ja mediaanipoikkeama säteelle r ja tilavuudelle V 10 % kokoluokkaa, mikä teki siitä muihin menetelmiin verrattuna huonomman. Haastavamman sylinterin 2 kohdalla sen tulokset olivat kuitenkin edelleen kohtalaisia ja mediaanipoikkeama 10 % kokoluokkaa, mikä sylinterin 2 kohdalla tarkoitti, että RANSAC menestyi muihin menetelmiin verrattuna paremmin. Erityisesti MSACin ja $T_{d,d}$ -menetelmien suoriutumista leimasi se, että ne suoriutuivat sylinterin 1 kohdalla hieman RANSACia paremmin mutta sylinterin 2 kohdalla sitä heikommin. Inner-RANSAC ja inner-RANSAC-Kt-yhdistelmä tuottivat erinomaiset säteet ja tilavuudet sylinteripistepilvestä 1 mutta tulokset romahtivat sylinteripistepilven 2 kohdalla, ja sädetä ja tilavuutta lukuunottamatta muiden muuttujien arvot olivat näille menetelmille myöskin muihin menetelmiin verrattuina keskitasolla tai sitä alempana. Hyvät mediaanitulokset sylinterin 1 sovituksessa säteen ja tilavuuden osalta saattoivat siis olla onnekasta sattumaa. Kt-menetelmä menestyi yleisesti melko hyvin eli samantasoisesti RANSACin kanssa, mutta joidenkin muuttujien kohdalla kuitenkin selkeästi huonommin kuin RANSAC. Se ei siis vaikuta tuottaneen parannuksia RANSACiin verrattuna.

5.1.2 RANSAC-tunnusluvut sylinteridatalla

Taulukko 5.2 sisältää joitakin RANSAC-menetelmän tunnuslukuja. Ensimmäinen niistä on mallisovituksen löytämien hyvien havaintojen osuus pistepilvidatassa, ϵ . Sen vertailuarvon muodostamiseksi taulukon oikeanpuolimmaiseen sarakkeeseen on laskettu, kuinka suuri osa sylinterien pistepilvidatasta on työssä käytettyä virhetoleranssitason t arvoa 0,01 lähempänä todellisen sylinterimallin pintaa. Toisena RANSAC-tunnuslukuna tarkastellaan mallin sovittamiseksi suoritettujen iteraatiokierrosten määrää k , jonka vertailuarvot taulukossa ovat luvussa 3.2.5 käsitellyn kaavan 3.8 perusteella odotettuja iteraatiomääriä. Viimeisenä taulukossa esitetään RANSAC-kustannusfunktioiden arvot luvun 3.2.2 kaavan 3.2 mukaisesti, paitsi MSAC-menetelmän kohdalla, jolle on ilmoitettu kaavan 3.4 mukaiset MSAC-kustannusfunktion arvot. MSAC-kustannusfunktion arvoja voi erilaisen laskentatavan vuoksi vertailla taulukossa vain toisiinsa sylinteripistepilvien 1 ja 2 välillä muttei muiden menetelmien kustannusfunktioiden arvoihin, mistä muistutuksena taulu-

Taulukko 5.2. Testattujen RANSAC-menetelmäversioiden suoritumista kuvaavia arvoja sylinterien 1 ja 2 pistepilvidatalla. Muuttujista on esitetty 30 suorituskerran mediaani, keskiarvo ja keskihajonta tässä järjestyksessä. Viimeisessä sarakkeessa on esitetty vertailuarvot, jotka on laskettu pistepilvien pohjana olleet todelliset sylinterimallien parametrit tuntien.

muuttuja	RANSAC	MSAC	RANSAC + LMS	$T_{d,d}$	inner-R.	Kt	inner-R. + Kt	vertailu
ϵ	S1	0,630	0,629	0,629	0,632	0,629	0,631	0,634
		0,562	0,595	0,546	0,589	0,576	0,554	
		0,095	0,0779	0,104	0,0819	0,0863	0,0996	
	S2	0,368	0,338	0,367	0,367	0,355	0,367	0,363
		0,362	0,346	0,361	0,362	0,359	0,361	
		0,00999	0,0160	0,0109	0,00985	0,0107	0,0108	
k	S1	310	310	318	295	312	338	303
		4 420	3 310	6 860	3 340	4 370	6 210	
		5 470	5 930	9 800	5 890	5 270	8 480	
	S2	66 300	100 000**	66 800	69 100	95 100	66 800	81 900
		76 100	89 100	78 100	77 000	82 200	77 900	
		17 400	13 600	18 700	15 800	18 500	18 000	
F	S1	611	0,0798*	612	607	612	609	603
		723	0,0856*	748	678	699	735	
		157	0,0124*	171	135	142	164	
	S2	1 040	0,124*	1 040	1 040	1 060	1 040	1 050
		1 050	0,124*	1 050	1 050	1 060	1 050	
		16,5	0,000617*	18,0	16,2	17,6	17,7	

kossa MSAC-kustannusfunktion arvojen perään on merkitty asteriskit *. Suoritusohjelma sallima maksimi-iteraatiomäärä nostettiin sylinteripistepilven 2 kohdalla muutoin käytetystä 50 000 iteraatiosta 100 000 iteraatiokierrokseen sen pienen ϵ -arvon takia. Tämän maksimi-iteraatiomäärän saavuttavat luvut on merkitty kaksoisasteriskilla ** huomautuksena siitä, että luku ei ole muodostunut luonnollisesti vaan se olisi voinut nousta korkeammaksi, jos ohjelman olisi sallittu käyttää enemmän aikaa iterointiin.

Hyvien havaintojen osuuksien ϵ mediaanit mukailevat sylinterin 1 kohdalla todella hyvin odotettua vertailuarvoa 0,634 alle 1 prosentin poikkeamilla. Keskiarvot sylinterille 1 ovat myös kohtalaisia, poiketen kuitenkin noin 10 %, maksimissaan noin 15 % LMS-menetelmän kohdalla. ϵ -arvojen jakaumat sylinterille 1 ovat siis vasemmalle vinoja, mikä on odotettavissa. Välillä menetelmien löytämät mallisovitukset M_{max} ovat huonompia kuin datan D pohjalla oleva todellinen objekti ja löydetyt ϵ -osuudet ovat todellista matalampia, mutta todellista objektiä parempaa mallia ja todellista suurempaa ϵ -arvoa ei ole mahdollista löytää. Keskihajonnat sylinterin 1 ϵ -arvoille ovat 15 prosentin suuruusluokkaa. Sylinterille 2 mediaanit ovat melko hyviä mutteivät yhtä hyviä kuin sylinterin 1 kohdalla, poiketen kuitenkin alle 10 % vertailuarvosta 0,363. Sylinterin 2 kohdalla keskiarvot ovat lähempänä mediaaneja sylinterin 1 tapaukseen verrattuna, poiketen mediaanien tavoin vertailuarvosta alle 10 %. Sylinteriin 1 verrattuna sylinterin 2 keskihajonnat ϵ -arvoille eri

menetelmien kohdalla ovat myös selvästi pienempiä, vain parin prosentin luokkaa vertailuarvoon verrattuna. Sylinterin 2 kohdalla löydettyt ϵ -jakaumat eivät ole sylinterin 1 tapaan selkeästi vasemmalle vinoja, vaan joidenkin menetelmien kohdalla keskiarvo on suurempi kuin mediaani ja joidenkin kohdalla toisinpäin. Matalamman ϵ -arvon sylinteripistepilven 2 kohdalla löydetty ϵ -arvot ovat siis keskimäärin kauempana pistepilven todellisesta ϵ -arvosta, mutta ne ovat tasaisemmin jakautuneita. Parhaan keskiarvon ja keskihajonnan sylinterille 1 tuotti MSAC ja sylinterille 2 RANSAC ja $T_{d,d}$ -menetelmä yhdessä. Huonoimman keskiarvon ja keskihajonnan sylinterille 1 tuotti LMS ja sylinterille 2 MSAC.

Iteraatiokierrosten k mediaanit sylinterille 1 olivat pienimmät inner-RANSAC-Kt-yhdistelmälle ja $T_{d,d}$ -menetelmälle. Ne jäivät 5 % ja 3 % alle kaavan 3.8 perusteella lasketun vertailuarvon 303. Kt-menetelmä vaati mediaaniltaan yli 10 % vertailuarvoa enemmän iteraatiokierroksia, ja muiden menetelmien mediaanit olivat alle 5 % vertailuarvoa suurempia. Keskiarvot ja keskihajonnat olivat suuria sylinterille 1, tuhansia kaikkien menetelmien kohdalla eli kymmeniä kertoja vertailuarvon 303 verran. Sylinterin 2 kohdalla mediaani-iteraatiomäärä jäi 15–20 % vertailuarvoa 81 900 matalammaksi RANSACilla, LMS-menetelmällä, $T_{d,d}$ -menetelmällä ja Kt-menetelmällä, kun se taas nousi 10–15 % vertailuarvoa suuremmaksi inner-RANSACille ja inner-RANSAC-Kt-yhdistelmälle. Inner-RANSACin ja inner-RANSAC-Kt-yhdistelmän käyttämät suuremmat iteraatiomäärät sylinterille 2 ovat linjassa sen tuloksen kanssa, että niiden aikaansaamat mallisovitukset sylinterin 2 pistepilveen olivat luvussa 5.1.1 kuvattuun tapaan heikkoja. Sama koskee MSACia, joka suoritti ohjelman salliman maksimi-iteraatiomäärän 100 000 eli yli 20 % vertailuarvoa suuremman määrän iteraatiokierroksia. Sylinterille 2 menetelmien käyttämien iteraatiomäärien keskiarvot ovat huomattavan lähellä vertailuarvoa sylinteriin 1 verrattuna, poiketen siitä maksimissaankin alle 10 %, ja ovat välillä vastaavaa mediaania suurempia ja välillä pienempiä. Sylinterin 1 pistepilven kohdalla käytettyjen iteraatiokierrosmäärien k jakaumat ovat siis oikealle vinoja, kun taas sylinterin 2 kohdalla eivät. Huonoimman keskiarvon ja suurimman keskihajonnan tuotti sylinterille 1 LMS-menetelmä ja sylinterille 2 inner-RANSAC ja inner-RANSAC-Kt-yhdistelmä yhdessä.

Eri menetelmien tuottamat kustannusfunktion F mediaaniarvot sylinterille 1 ovat hyvin lähellä sekä toisiaan että vertailuarvoa 603. Erot ovat alle prosentin suuruisia. Keskiarvot ovat korkeampia, LMS-menetelmän keskiarvon erotessa vertailuarvosta lähes 25 % ja pienimmänkin eron ollessa $T_{d,d}$ -menetelmän kohdalla yli 10 %. Kustannusfunktioiden arvojen keskihajonta sylinterille 1 on menetelmillä 25 prosentin suuruusluokkaa vertailuarvoon ja mediaaneihin verrattuna. MSAC-kustannusfunktion arvoa ei voida erilaisen laskutavan vuoksi verrata muihin menetelmiin, mutta sen mediaanin, keskiarvon ja keskihajonnan suhteet toisiinsa nähden eroavat hieman muista menetelmistä: keskiarvon ero mediaaniin on pienempi, noin 7 %, ja hajonta on mediaaniin verrattuna noin 15 %. Sylinterin 2 kohdalla sekä mediaanit ja keskiarvot ovat sylinterista 1 poiketen hyvin lähellä toisiaan ja vertailuarvoa, ja keskihajonnat ovat sylinterin 2 kohdalla pienempiä, vain 1-

2 prosentin suuruusluokkaa vertailuarvoon nähden. Parhaat keskiarvot ja keskihajonnat tuotti $T_{d,d}$ -menetelmä ja huonoimmat LMS-menetelmä.

MSAC vaikuttaa erottuvan muista menetelmistä pienemmällä kustannusfunktioarvojen F jakauman hajonnalla sekä sylinterille 1 että sylinterille 2, vaikka sen tuottamat mallisovituksiset eivät muilta tunnusluvuiltaan, erityisesti löydettyjen hyvien havaintojen osuuk-sien ϵ ole hajonnaltaan muita menetelmiä pienempiä. Tämä voi selittyä sillä, että MSAC-kustannusfunktion arvot eri hypoteesimallikandidaattien M_i välillä voivat erota toisistaan äärettömän vähän. Jos kahden mallin poikkeaviksi havainnoiksi $d \in P$ luokittelemien pisteiden määrä on sama, niiden kustannusfunktioiden arvot eroavat toisistaan vain joidenkin hyviksi havainnoiksi $d_j \in H$ luokiteltujen pisteiden mallivirhetermien e_j^2 summan verran, ja virhetermit e_j voivat olla mielivaltaisen pieniä. Myös kahden erisuuret konsensusjoukot C_i muodostaneen mallin MSAC-kustannusfunktioiden arvot F_i voivat teoriassa olla mielivaltaisen lähellä toisiaan, sillä hyväksi havainnoksi hyväksytyn pisteen d ero e malliin voi olla äärettömän lähellä hylkäykseen johtavaa virhetoleranssiarvoa t ja alittaa sen vain äärimmäisen täpärästi. Tosin jos MSAC-kustannusfunktioiden arvojen hajonta olisi ollut RANSAC-kustannusfunktion arvojen hajontaa suurempaa, myös tätä oltaisiin voitu pyrkiä selittämään MSAC-kustannusfunktion ominaisuuksilla, sillä teoriassa kaksi samankokoista konsensusjoukkoa C_i kerännyttä hypoteesimallia M_i voivat MSAC-kustannusfunktio-arvojensa osalta erota selvästi toisistaan, kun taas kaksi samankokoista konsensusjoukkoa kerännyttä mallia saavat aina samat RANSAC-kustannusfunktion arvot. Tämä efekti kuitenkin näyttää taulukon 5.2 tulosten perusteella jäävän marginaaliseksi.

Kokonaisuutena taulukon 5.2 tuloksista voidaan huomata, että odotetun kaltaiset RANSAC-tunnusluvut eivät välttämättä sellaisenaan kerro mallisovituksen onnistumisesta. Luvussa 5.1.1 todettiin sylinterisovitusten onnistuneen paremmin sylinterin 1 kuin sylinterin 2 pistepilven kohdalla, ja pistepilvestä 2 tuotetut sylinteriparametrit eivät usein olleet lähelläkään oikeita arvoja. Taulukon 5.2 RANSAC-tunnuslukutulokset ovat kuitenkin lähellä odotettuja arvoja sekä sylinterin 1 että sylinterin 2 kohdalla, ja sylinterin 2 kohdalla ne ovat jopa lähempänä vertailuarvoja sekä tasaisemmin jakautuneita. Menetelmät ovat siis tuottaneet sekä sylinteripistepilven 1 että sylinteripistepilven 2 joukon huonoja malleja, mutta laadukkaamman sylinteripistepilven 1 pohjalta tehtyjen sovitusten joukko sisältää huonojen mallien lisäksi useampia paljon onnistuneempia sovituksia, mikä saa aikaan suurempaa hajontaa sylinterin 1 sovitustuloksissa.

$T_{d,d}$ -menetelmä ei tässä tarkastelussa vaikuta lisänneen tarvittavaa iteraatiokierrosten määrää k , toisin kuin teoria antaa odottaa. Taulukon 5.2 tulosvertailussa $T_{d,d}$ -menetelmä menestyi keskimäärin hieman RANSACia paremmin, mutta luvussa 5.1.1 todettiin, etteivät sen sovittamat sylinterimallit olleet aina yhtä tarkkoja kuin RANSACin sovittamat. Sama koskee Kt-menetelmää. Inner-RANSAC-Kt-yhdistelmä käytti RANSACiin verrattuna vähemmän iteraatiokierroksia k sylinterin 1 sovittamiseen mutta enemmän sylinterin 2 sovittamiseen, joten sen iteraatiokierroksia vähentävästä vaikutuksesta ei ainakaan tämän

työn toteutusmuodossa ja näillä sylinteripistepilvillä testattaessa näy merkkejä. Raguramin ym. [9] toteuttamassa muodossa se kirjoittajien mukaan vähensi RANSACin tarvitsemien iteraatiokierrosten määrää kaavan 3.8 antamaa arvoa korkeammalta tasolta kaavan tasolle. Sylinterisovituksia tarkasteltaessa ei kuitenkaan näy selviä merkkejä myöskään siitä, että RANSAC-menetelmät johdonmukaisesti vaatisivat kaavaa korkeampia iteraatiomääriä. Hajonta kierrosmäärissä k on tosin suurta ja niiden jakaumat oikealle vinoja, joten Raguramin ym. toteutuksen, jonka mukaan RANSAC "usein" vaatii kaavaa 3.8 enemmän kierroksia, voidaan kyllä tulkita myös pitävän paikkansa.

LMS-menetelmä jatkaa taulukossa 5.2 taulukon 5.1 tarkastelun tapaan suurimpia hajontoja sisältävien tulosjakaumien tuottamista. Pienimmän neliösumman menetelmän korvaaminen pienimmän neliömediaanin menetelmällä RANSAC-hypoteesimallien sovituksessa vaikuttaa siis entisestään korostavan RANSAC-tulosten satunnaisuutta.

5.2 RANSAC-versiot puunrunгон pistepilvidatalla

Tässä kappaleessa käydään läpi luvussa 4.5 määritellyt tulosmuuttujat eli muodostettujen runkomallien tilavuudet V , pisteiden keskivertoetäisyys sylinterimalliin mad , löydetty hyvien havaintojen osuudet datassa ϵ , käytetyt iteraatiokierrosmäärät k , kustannusfunktioiden arvot F ja havaitut poikkeavien sylinterien lukumäärät an jokaiselle sovelletulle RANSAC-menetelmälle, kun testidatana toimii keskikokoisen tammirungon laserkeilausdata. Pistepilviä on muodostettu 4 kappaletta resoluutiota ja käytössä olevien laserskannausten määrää varioimalla luvussa 4.4 kuvattuun tapaan niin, että runkopistepilvien 1 ja 2 pisteet ovat peräisin vain yhdestä laserskannuksesta ja runkopistepilvien 3 ja 4 kahdesta ja runkopistepilvien 2 ja 4 resoluutio on runkopistepilvien 1 ja 3 resoluutiota suurempi. Tuloksista on ilmoitettu yhden runkomallin kaikkien sylinterien mediaanien mediaani, keskiarvo ja keskihajonta 30 suorituskerran ajalta, ja ne on esitetty taulukoissa 5.3 ja 5.4. Taulukossa 5.3 esitetään tulokset runkopistepilville 1 ja 2, jotka sisältävät vain yhden laserskannuksen datapisteet ja taulukossa 5.4 runkopistepilvien 3 ja 4 tulokset, jotka on saatu kahden laserskannuksen datapisteitä käyttämällä. Runkojen 1 ja 3 resoluutio on 0,25 pistettä per neliösenttimetri ja runkojen 2 ja 4 resoluutio korkeampi, noin 3–4 pistettä per neliösenttimetri. Taulukoiden oikeanpuolimmaisien sarakkeen vertailuarvo puunrunгон tilavuudelle V on luvussa 4.4 kuvattuun tapaan tuotettu kuudesta eri laserskannuksesta peräisin olevista runkopisteistä. Sen sijaan vertailuluvut ϵ -arvoille ja kustannusfunktioarvoille F on laskettu jokaiselle runkopistepilville 1–4 erikseen RANSACilla nostamalla sallittu maksimi-iteraatiomäärä 200 000 iteraatiokierrokseen ja käyttämällä 30 suorituskerran mediaaniarvoa. mad - ja an -muuttujilla ei ole vertailuarvoa, mutta näiden muuttujien arvojen tulisi hyvän mallin kohdalla olla mahdollisimman pieniä eli lähellä arvoa 0. MSAC-kustannusfunktion arvot F eivät erilaisen laskutavan vuoksi ole taulukoissa vertailukelpoisia muiden menetelmien antamien kustannusfunktion arvojen kanssa, mutta

ne ovat vertailukelpoisia eri runkopistepilvein välillä ja ilmoitetaan taulukoissa asteriskilla * merkittynä.

5.2.1 Menetelmien suoriutuminen yhden laserskannauksen tuottamasta pistepilvestä

Taulukossa 5.3 mediaanein mitattuna lähimmäksi mallinnetun rungon todellista tilavuutta V pääsi harvemman runkopistepilven 1 kohdalla Kt-menetelmä, jonka tuottama mediaanitulavuus erosi oikeasta arvosta 0,806 noin prosentin verran. MSACin ja LMS-menetelmän tuottamat tilavuudet erosivat todellisesta alle 2 % ja muiden menetelmien maksimissaan 10 %. Keskiarvot poikkesivat mediaaneista ja vertailuarvosta alaspäin muutamia prosentteja, mutta erityisesti inner-RANSAC-Kt-yhdistelmän, $T_{d,d}$ -testin ja RANSACin kohdalla 15–20 %. Näiden menetelmien ja lisäksi MSACin kohdalla keskihajonnat olivat runkopistepilven 1 tapauksessa myös suurimpia. Keskihajonnat olivat yleisesti 10–30 % tilavuuden V vertailuarvosta. Runkopistepilven 2 osalta MSAC, Kt, RANSAC, $T_{d,d}$ -testi ja LMS-menetelmä erosivat mediaanitulavuudelta alle 5 % todellisesta ja inner-RANSAC-Kt-yhdistelmän sekä inner-RANSACin mediaanitulavuudet noin 5 % ja 10 %. Keskiarvot runkopistepilven 2 mallitulavuuksille poikkeavat runkopistepilven 1 tapaan hieman alaspäin, alle 10, % paitsi inner-RANSACin tapauksessa noin 15 %. Inner-RANSAC, inner-RANSAC-Kt-yhdistelmä ja LMS-menetelmä, joiden tuottamat keskihajonnat runkopistepilvelle 1 olivat pienimmät, tuottivat runkopistepilvelle 2 taas suurimmat keskihajonnat. Keskimäärin runkopistepilvelle 2 keskihajonnat ovat kuitenkin pistepilveä 1 pienempiä, noin 5–10 % tilavuuden vertailuarvosta. Runkopistepilven 2 kymmenkertainen resoluutio runkopistepilveen 1 verrattuna ei näytä siis tuottaneen mediaaneilla mitattuna tarkempia tuloksia, mutta tuloksien hajonta on pienempää ja keskiarvot lähempänä mediaaneja.

Havaintopisteiden etäisyyttä sovitettuihin malleihin mittaavat mad -arvot ovat taulukossa 5.3 kaikille menetelmille sekä mediaaniltaan että keskiarvoltaan noin 4 millimetrin luokkaa. Runkopistepilvelle 2 ne ovat aavistuksen korkeampia kuin runkopistepilvelle 1, mikä voi johtua pistepilven 2 suuremmasta havaintopisteiden kokonaismäärästä. Suurempi pisteiden määrä toisaalta mahdollistaa tarkemman mallisovituksen, jos hyvien havaintojen suhteellinen osuus ϵ säilyy samana, mutta toisaalta se myös lisää kaukana sovitettavasta mallista sijaitsevien pisteiden määrää. mad -arvojen keskihajonnat taulukossa 5.3 ovat suuruusluokaltaan millimetrien kymmenesosia ja sadasosia.

Löydetyt hyvien havaintojen osuudet ϵ ovat hyvin lähellä runkopistepilville laskettuja vertailuarvoja. Runkopistepilven 1 kohdalla mediaaniltaan jopa eniten poikkeavan MSACin löytämä ϵ -arvo eroaa vertailuarvosta alle 5 %, ja runkopistepilven 2 kohdalla eniten eroavat MSAC ja RANSAC eroavat mediaaniltaan vertailuarvosta alle 2 %. Kuten sylinteripistepilvien kohdalla luvussa 5.1.2, löydetyt konsensusjoukot C_i vaikuttavat pääsevän lähelle todelliseksi oletettua hyvien havaintojen osuutta havaintodatassa. Kaikkien mene-

Taulukko 5.3. Testattujen RANSAC-menetelmäversioiden tuottamat tulokset runkopistepilvillä, jotka on muodostettu vain yhden laserskannauksen mittausdatasta. Rungon numero 1 resoluutio on pienempi kuin rungon 2.

muuttuja	RANSAC	MSAC	LMS	$T_{d,d}$	inner-R.	Kt	inner-R. + Kt	vertailu	
V	R1	0,736	0,819	0,796	0,747	0,725	0,814	0,730	0,806
		0,696	0,826	0,774	0,671	0,723	0,800	0,665	
		0,181	0,197	0,150	0,248	0,0883	0,0823	0,190	
	R2	0,771	0,778	0,769	0,770	0,722	0,771	0,747	
		0,757	0,791	0,749	0,762	0,689	0,754	0,731	
		0,0718	0,0517	0,107	0,0784	0,108	0,106	0,0793	
$mad,$ mm	R1	3,60	3,48	3,73	3,70	3,55	3,63	3,53	-
		3,60	3,51	3,70	3,67	3,59	3,59	3,55	
		0,217	0,0973	0,237	0,186	0,234	0,234	0,252	
	R2	4,03	3,92	4,03	4,03	3,88	3,88	3,84	
		4,03	3,91	4,03	4,02	3,87	3,86	3,87	
		0,0779	0,101	0,0728	0,0955	0,184	0,0824	0,151	
ϵ	R1	0,785	0,763	0,781	0,785	0,792	0,799	0,789	0,799
		0,745	0,737	0,763	0,705	0,788	0,797	0,745	
		0,146	0,139	0,0434	0,238	0,0228	0,0163	0,147	
	R2	0,765	0,765	0,773	0,773	0,768	0,775	0,774	0,776
		0,761	0,765	0,757	0,765	0,753	0,760	0,765	
		0,0274	0,0116	0,0376	0,0339	0,0408	0,0370	0,0304	
k	R1	63	79	71	70	55,5	54,3	63	27
		1 744	1 740	176	5 070	58,5	54,9	1 740	
		9 114	9 110	507	15 200	23,4	19,2	9 120	
	R2	64,8	66	57,3	57,8	59,5	59	59,5	37
		77,8	66,7	88,5	92,2	108	85,8	78,4	
		44,3	18,7	80,2	163	178	84,3	64,6	
F	R1	20	0,00379*	20,3	20,5	20	20	19,5	20
		19,8	0,00374*	20,5	20,2	19,8	19,6	19,3	
		1,67	0,000254*	2,08	1,66	0,996	1,05	1,66	
	R2	1 130	0,203*	1 090	1 120	1 130	1 100	1 130	1 090
		1 100	0,204*	1 080	1 090	1 090	1 080	1 090	
		50,4	0,00839*	62,3	66,3	71,7	78,1	50,6	
an	R1	9	8	9	9	9	9	9	-
		9,73	9,30	9,20	10,6	10,4	9,33	9,63	
		3,13	4,08	2,78	3,78	3,67	3,28	3,09	
	R2	5	5	5	5	6	5	6	
		5,17	4,80	6,13	6,03	6,33	5,43	5,93	
		1,34	0,925	3,64	3,55	1,90	1,89	1,74	

telmien keskiarvot ovat hieman matalampia kuin vastaavan menetelmän mediaani, mutta niiden suuruusluokka on sama. Joukosta poikkeaa hieman $T_{d,d}$ -menetelmä runkopistepilven 1 kohdalla, sillä sen ϵ -keskiarvo eroaa vertailuarvosta ja mediaanista yli 10 %. LMS-menetelmän, inner-RANSACin ja Kt-menetelmän tuottamat keskihajonnat runkopistepilvelle 1 ovat pieniä, noin 3 % ϵ -arvon vertailuarvosta, kun taas muiden menetelmien keskihajonnat pistepilvelle 1 ovat yli 20 % vertailuarvosta, selkeästi suurimpana $T_{d,d}$ -menetelmän keskihajonta. Keskihajonnat runkopistepilvelle 2 ovat sekä $T_{d,d}$ -menetelmän että muiden osalta pienempiä, kaikkien menetelmien kohdalla vain 1–5 % vertailuarvosta.

Menetelmien käyttämät iteraatiomäärät k ylittävät keskimäärin selkeästi ϵ -arvojen pohjalta kaavalla 3.8 lasketut vertailuarvot. Runkopistepilven 1 kohdalla MSAC oli k -arvojen mediaaneissa mitattuna hitain ja käytti lähes kolminkertaisen määrän iteraatioita vertailuarvoon 27 nähden. LMS ja $T_{d,d}$ -menetelmä käyttivät vastaavasti mediaaniltaan noin 160 % vertailuarvoa enemmän iteraatiota, RANSAC ja inner-RANSAC-Kt-yhdistelmä noin 133 % enemmän ja inner-RANSAC sekä Kt-menetelmä noin kaksinkertaisen määrän. Lisäksi runkopistepilven 1 kohdalla keskiarvot ja keskihajonnat iteraatiomäärille k mitattiin tuhansissa kaikille paitsi inner-RANSACille ja Kt-menetelmälle, joiden keskiarvot olivat samaa tasoa mediaanien kanssa ja keskihajonnat noin puolet sen arvosta, sekä LMS-menetelmälle, jonka keskiarvo ja keskihajonta mitattiin sadoissa. Suurempiresoluutiosien runkopistepilven 2 kohdalla, jolle k -vertailuarvo on kuitenkin pienemmän ϵ -arvon vuoksi 37 eli noin kolmaosan pistepilveä 1 suurempi, menetelmät suoriutuivat pistepilven 1 lukuihin verrattuna hieman pienemmästä ϵ -arvosta huolimatta keskimäärin nopeammin ja niiden keskinäiset suuret erot suoriutumisessa tasoittuivat melko paljon. Inner-RANSACin ja Kt-menetelmän käyttämien iteraatiomäärien mediaanit olivat pistepilveä 1 noin 10 % suurempia, mutta RANSACin ja inner-RANSAC-Kt-yhdistelmän olivat samaa tasoa. LMS-menetelmän, $T_{d,d}$ -menetelmän ja MSACin runkopistepilvelle 2 käyttämät mediaani-iteraatiomäärät olivat jopa 20 % runkopistepilven 1 määriä pienemmät. Menetelmien käyttämät mediaani-iteraatiomäärät olivat kuitenkin myös pistepilven 2 kohdalla vertailulukua suurempia, keskimäärin noin 60 %. Keskiarvot ja keskihajonnat olivat merkittävästi pienemmät runkopistepilven 2 kohdalla kuin runkopistepilven 1, ollen samaa kokoluokkaa mediaanien ja vertailuarvon kanssa. $T_{d,d}$ -menetelmän keskiarvo ja keskihajonta olivat kaikista menetelmistä selkeästi suurimmat runkopistepilvelle 1 ja toiseksi suurimmat runkopistepilvelle 2.

Kustannusfunktioiden arvoissa F mitattuna menetelmät ovat hyvin lähellä toisiaan ja vertailuarvoa sekä mediaaneilla että keskiarvoilla mitattuina, kuten luvussa 5.1.2 nähtiin sylinteripistepilvien kohdalla olevan. Keskiarvot eivät eroa selkeästi mediaaneista ylös- tai alaspäin. Keskihajonnoissa menetelmien välillä on hieman enemmän vaihtelua, mutta niidenkin arvo rajoittuu välille 5-10 % mediaaniarvoista molemmille runkopistepilville eikä havaittavissa ole yhtenäisyyksiä keskihajontojen suuruuksissa runkojen 1 ja 2 välillä eri menetelmien osalta. MSAC-kustannusfunktioiden hajonta ei ole enää selkeästi muiden

menetelmien kustannusfunktioiden hajontaa pienempää, kuten se yksittäisiä sylintereitä mallinnettaessa luvussa 5.1.2 oli.

Myös poikkeavien sylinterisovitusten määrät an rungoissa ovat taulukossa 5.3 hyvin samankaltaisia kaikille menetelmille, ja mediaanit ja keskiarvot ovat lähellä toisiaan. Keskihajontojen suuruusluokka on joitakin kymmeniä prosentteja mediaaniarvoista. Harvemmassa runkopistepilvessä 1 on tuotettu noin 9 edellisestä sylinteristä poikkeavaa sylinterisovitusta ja tiheämmästä runkopistepilvestä lähes puolet vähemmän, noin 5 kappaletta. Koska yhden poikkeavan sylinterimallin sovitus saattaa aiheuttaa rungossa seuraavan sylinterin merkitsemisen poikkeavaksi, vaikka se olisi runkoon hyvin sopiva, voidaan näitä poikkeavien sylinterien määriä ajatella ylärajoina ja alarajan voidaan katsoa olevan puolet tästä määrästä. Runkopistepilven 1 kohdalla vähintään 4–5 sylinteriä on siis eronnut jollain tavalla selkeästi rungossa sitä edeltävästä sylinteristä, mikä on suuri osuus koko runkoon sovitetusta 14 sylinteristä. Runkopistepilvessä 2 poikkeavia sylintereitä on syntynyt enää vähintään 2–3, mikä sekin on kuitenkin suuri prosentuaalinen osuus. Runkomalleja kannattaisikin koittaa parantaa antamalla sovitettujen RANSAC-sylinterien pituuden l määräytyä orgaanisesti runkopistepilven muotojen perusteella.

Kaiken kaikkiaan menetelmät onnistuivat määrittämään rungon mediaanitulavuuden V todella hyvin ottaen huomioon, että käytössä on vain yksi skannauspiste ja kuvattuna hyvin kapea osa puunrunkoa. Kaikkien menetelmien mediaanitulavuudet jäivät alle 10 % poikkeaman päähän todellisesta arvosta. Rungon 2 kymmenkertaisesti suurempi resoluutio runkopistepilven 1 verrattuna ei tuonut parannuksia tilavuuden mediaanitarkkuuteen, mutta se toi keskiarvot lähemmäksi mediaaneja ja pienensi tulosten keskihajontaa eli teki niistä tasalaatuisempia. LMS-menetelmä ei enää tässä tarkastelussa tuottanut muihin menetelmiin verrattuna suurta hajontaa tuloksiin, kuten se teki aiempien lukujen sylinteripistepilvien tarkastelussa, vaan LMS-menetelmä sulautui kaikilta osin muiden menetelmien joukkoon suoriutuen niihin nähden keskitasoisesti. Tilavuuden V määrittämisessä se sijoittui menetelmistä yläpäähän, muttei kuitenkaan ollut paras. MSAC, joka iteraatiomäärissä k mitattuna oli menetelmistä hitain, tuotti tässä tarkkuudeltaan kärkipäätä olevia tilavuuksia V . MSACin tavoin myös Kt-menetelmä, joka sylinteripistepilviin sovittaessaan luvuissa 5.1.1 ja 5.1.2 jäi kokonaissuorituskyvyltään RANSACia heikommaksi, suoriutui tässä RANSACia paremmin sekä tilavuuden V tarkkuudella että iteraatiomäärissä k mitattuna, ollen molemmissa menetelmien kärkikastia. RANSACin suoriutuminen kaikilla mittareilla oli muiden menetelmien joukossa keskitasoa. Inner-RANSAC ja inner-RANSAC-Kt-yhdistelmä olivat iteraatiokierroksin k mitattuina nopeimpia, mutta niiden tilavuustulokset V olivat menetelmistä häntäpäättä, poiketen kuitenkin alle 10 % todellisesta arvosta. Niiden käyttämät iteraatiokierrosmäärät eivät kuitenkaan olleet suuresti muita menetelmiä alemmalla tasolla ottaen huomioon sen, että inner-menetelmät suorittavat mallin uudelleensovituksessa ylimääräisiä inner-RANSAC-iteraatioita ja myös Kt-menetelmä tekee malleja uudelleensovittaessa laskentakapasiteetiltaan kalliita ope-

raatiota. $T_{d,d}$ -menetelmä ei vielä yksittäisiä sylintereitä sovitettaessa käyttänyt RANSACia enempää iteraatiokierroksia k , mutta nyt kokonaisten runkojen mallinnustuloksissa tilanne näyttää olevan toinen ja sen käyttämät iteraatiomäärät suuria.

5.2.2 Menetelmien suorituminen kahden laserskannauksen tuottamasta pistepilvestä

Taulukossa 5.4 eri menetelmien muodostamien runkomallien mediaanitulavuudet V ovat runkopistepilvelle 3 keskimäärin noin 20 % liian pieniä ja kauempana vertailuarvosta 0,806 kuin ne olivat taulukossa 5.3 runkopistepilville 1 ja 2. Tämä on odottamatonta, sillä runkopistepilvet 3 ja 4 sisältävät kahdesta laserskannauksesta muodostettuna enemmän havaintodataa kuin runkojen 1 ja 2 yhdestä skannauksesta muodostetut pistepilvet. Oletuksen mukaan suurempi määrä havaintodataa johtaisi parempiin sovitustuloksiin. On mahdollista, että käytössä olleet 6 laserskannausta eivät ole olleet tuottamansa mittausdatan suhteen tasalaatuisia. Tästä antaa viitteitä myös se, että luvussa 4.4 runkopistepilvien tiedot kokoavan taulukon 4.1 mukaan toisen skannauksen ottaminen mukaan ei kaksinkertaista datapisteiden d lukumäärää N rungossa 3 runkoon 1 verrattuna eikä rungossa 4 runkoon 2 verrattuna, vaan pisteiden määrä kasvaa ensimmäisessä tapauksessa noin 30 % ja toisessa noin 70 %. Toisen mukaan otetun laserskannauksen tuottamien havaintopisteiden määrä on siis ensimmäistä laserskannausta matalampi.

Todellista arvoa lähimpänä olevan mediaanitulavuuden ilmoitti runkopistepilvelle 3 MSAC 15 % poikkeamalla ja suuremman pistetiheyden omaavalle runkopistepilvelle 4 inner-RANSAC-Kt-yhdistelmä noin 13 % poikkeamalla. Kauimmaksi tilavuuden todellisesta arvosta mediaaniltaan jäi rungolle 3 $T_{d,d}$ -menetelmä noin 22 % poikkeamalla ja rungolle 4 inner-RANSAC noin 23 % poikkeamalla. Keskihajonnat olivat runkojen 1 ja 2 tilanteen tapaan mediaaneja hieman pienemmät harvempiresoluutioiselle runkopistepilvelle 3, mutta pistepilven 4 kohdalla selkeää eroa ei ole havaittavissa mediaanien ja keskiarvojen välillä, joten tilavuusjakaumat tiheämmälle runkopistepilvelle 4 eivät ole samaan tapaan vasemmalle vinoja kuin harvemmälle runkopistepilvelle 3. Runkopistepilvien 1 ja 2 tarkastelun tapaan myös tilavuuden keskihajonnat ovat pienempiä tiheämmälle runkopistepilvelle 4 kuin harvemmälle runkopistepilvelle 3. $T_{d,d}$ -menetelmän keskihajonta oli suurinta molemmille runkopistepilville 3 ja 4. Inner-Kt-RANSAC-yhdistelmä tuotti rungolle 4 menetelmistä parhaan mediaanitulavuuden, mutta sekin poikkesi vertailuarvosta 13 %, ja huonoimman mediaanitulavuuden arvo rungolle 4 inner-RANSACilta poikkesi jopa 23 %.

mad -arvot ovat taulukossa 5.4 runkopistepilville 3 ja 4 samaa luokkaa kuin ne olivat runkopistepilville 1 ja 2 taulukossa 5.3, noin 4 millimetriä. Aiempaan tapaan mediaanit ja keskiarvot ovat lähempänä toisiaan ja arvot suuriresoluutioiselle rungolle 4 hieman suurempia kuin rungolle 3.

Taulukko 5.4. Testattujen RANSAC-menetelmäversioiden tuottamat tulokset runkopiste-pilvillä, jotka on muodostettu kahden laserskannauksen mittausdatasta. Rungon numero 3 resoluutio on pienempi kuin rungon 4.

muuttuja	RANSAC	MSAC	LMS	$T_{d,d}$	inner-R.	Kt	inner-R. + Kt	vertailu	
V	R3	0,642	0,686	0,646	0,631	0,672	0,659	0,806	
		0,601	0,605	0,591	0,523	0,624	0,584		
		0,139	0,189	0,147	0,248	0,156	0,203		
	R4	0,661	0,670	0,662	0,652	0,617	0,658		
		0,673	0,705	0,696	0,603	0,643	0,687		
		0,0652	0,0540	0,0630	0,150	0,0578	0,0513		
$mad,$ mm	R3	3,92	3,77	3,98	3,94	3,81	3,82	-	
		3,90	3,77	3,99	3,97	3,80	3,82		
		0,227	0,142	0,224	0,296	0,215	0,255		
	R4	4,31	4,18	4,30	4,29	4,17	4,15		
		4,32	4,17	4,30	4,29	4,17	4,13		
		0,0567	0,0567	0,0478	0,0533	0,164	0,0548		
ϵ	R3	0,695	0,674	0,687	0,688	0,698	0,696	0,700	
		0,676	0,626	0,665	0,583	0,664	0,632		
		0,0341	0,145	0,0830	0,234	0,116	0,176		
	R4	0,647	0,634	0,649	0,644	0,648	0,649	0,649	
		0,642	0,632	0,647	0,612	0,646	0,648		
		0,0140	0,00877	0,00818	0,118	0,00921	0,00667		
k	R3	139	177	154	166	137	138	105	
		239	4 022	1 480	7 660	3 130	4 125		
		187	12 400	6 860	17 400	11 400	12 900		
	R4	298	358	284	332	290	288		225
		301	347	276	2 070	279	276		
		74,0	65,7	53,8	9 060	52,6	42,9		
F	R3	42	0,00627*	42	42	41	42,3	42	
		42,0	0,00624*	42,1	40,9	41,3	42,5		
		1,18	0,000225*	1,42	7,06	0,953	1,68		
	R4	2 130	0,412*	2 135	2 150	2 150	2 120		2 160
		2 140	0,414*	2 140	2 110	2 150	2 140		
		40,2	0,0143	54,3	254	66,6	54,0		
an	R3	9	7	9,50	10	9	10	-	
		9,53	8,13	10,2	11,0	9,80	10,5		
		2,64	2,65	3,17	3,20	2,71	2,81		
	R4	6	4	6	6	7	6		
		5,47	4,37	5,87	6,70	7,67	5,63		
		1,22	2,81	3,37	2,67	2,59	2,58		

Menetelmien löytämät ϵ -arvot runkopistepilville 3 ja 4 noudattavat kaikilta osin samoja linjoja kuin edellä käsiteltyjen runkopistepilvien 1 ja 2 kohdalla. Kaikkien menetelmien mediaanit erosivat vertailuarvosta alle 5 % runkopistepilven 3 kohdalla ja alle prosentin runkopistepilven 4 kohdalla. Keskiarvot olivat hieman mediaaneja pienempiä ja keskihajonnat suurempiresoluutioisen pistepilven 4 kohdalla pienempiä pistepilven 3 keskihajontoihin verrattuna. Pienimmät ϵ -arvojen mediaanit molemmille rungoille tuotti MSAC ja suurimmat keskihajonnat $T_{d,d}$ -menetelmä.

Käytetyt iteraatiokierrosmäärät k runkopistepilville 3 ja 4 heijastavat näiden pistepilvien pienempiä ϵ -arvoja runkopistepilviin 1 ja 2 verrattuna ja ovat selkeästi suurempia. Kuten MSAC-menetelmän löytämien menetelmistä pienimpien ϵ -arvojen perusteella voidaan odottaa, MSACin käyttämät iteraatiomäärät k olivat mediaaneiltaan molempien runkopistepilvien kohdalla suurimpia ylittäessään vertailuarvon runkopistepilven 3 kohdalla noin 70 prosentilla ja runkopistepilven 4 kohdalla noin 60 prosentilla. Pistepilven 3 kohdalla LMS-menetelmä ja $T_{d,d}$ -menetelmä käyttivät mediaaneiltaan noin 60 % ja 45 % vertailuarvoa 105 enemmän iteraatioita ja RANSAC, inner-RANSAC, Kt-menetelmä ja inner-RANSAC-Kt-yhdistelmä noin 30 % vertailuarvoa enemmän. Menetelmien k -arvojen keskinäinen järjestys runkopistepilven 3 kohdalla noudattaa samaa järjestystä kuin edellä myös pieniresoluutioisen runkopistepilven 1 kohdalla, mutta toisen laserskannauksen lisäämisen jälkeen menetelmien k -arvojen mediaanit eivät ole prosentuaalisesti aivan yhtä kaukana vertailuarvon yläpuolella pistepilvelle 3. Muiden menetelmien keskiarvot ja keskihajonnat mitataan tuhansissa runkopistepilven 3 kohdalla, kuten aiemmin runkopistepilven 1 kohdalla, paitsi RANSACin ja inner-RANSAC-Kt-yhdistelmän kohdalla mittaluokka on joitakin satoja. Pistepilven 4 kohdalla resoluution nostaminen runkopistepilveen 3 verrattuna on tuonut keskiarvot ja keskihajonnat mediaanien tasolle, mihin poikkeuksen tekee vain $T_{d,d}$ -menetelmä muita menetelmiä merkittävästi suuremmalla keskiarvolla ja keskihajonnalla. Myös $T_{d,d}$ -menetelmän k -arvojen mediaani pistepilvelle 4 on menetelmistä suurimmasta päästä, lähes 50 % vertailuarvoa suurempi ja toiseksi suurin MSACin jälkeen. Pistepilven 4 iteraatiomäärä on pienin inner-RANSAC-Kt-yhdistelmälle, jonka mediaani on vain 14 % vertailuarvoa 225 suurempi. RANSACin, LMS-menetelmän, inner-RANSACin ja Kt-menetelmän mediaanit runkopistepilvelle ovat noin 30 % vertailuarvoa suuremmat.

Kustannusfunktioiden arvot F noudattivat rungoille 3 ja 4 samaa kaavaa kuin rungoille 1 ja 2 aiemmin keskiarvojen ja mediaanien suhteen. Arvot noudattelevat vertailuarvoa. Kunkin menetelmän F -arvojen keskiarvo ja mediaani ovat taulukossa 5.4 hyvin lähellä toisiaan, ja mediaanit ja keskiarvot ovat myös eri menetelmien välillä vahvasti samaa tasoa. Keskihajonnat ovat sekä rungolle 3 että rungolle 4 muutaman prosentin luokkaa, mistä poikkeuksena $T_{d,d}$ -menetelmän keskihajonta rungolle 3 on yli 15 % ja rungolle 4 myöskin yli 10 %.

Poikkeavien sylinterisovitusien määrä rungoissa eli an -arvo oli rungoille 3 ja 4 samaa luokkaa mutta aavistuksen korkeampi kuin rungoille 1 ja 2 aiemmin. Siinä missä mene-

telmät ilmoittivat rungolle 1 an -arvoksi mediaaniltaan 8–9 ja rungolle 2 5–6, runkopistepilvälle 3 mallinnettiin mediaanina 9–10 poikkeavaa sylinteriä per runko ja tiheämmälle runkopistepilvälle 4 6–7 poikkeavaa sylinteriä per runko. Poikkeuksena MSAC ilmoitti vähemmän poikkeavia sylintereitä, 7 kappaletta rungolle 3 ja 4 kappaletta rungolle 4. Tämä jatkaa trendiä taulukosta 5.3, jossa MSAC ainoana ilmoitti yhden poikkeavan sylinterin vähemmän rungolle 1 kuin muut menetelmät.

Kokonaisuutena toisen laserskannauksen lisääminen yhden laserskannauksen rinnalle ei vaikuta tuottaneen tarkempia runkomalleja. Päinvastoin kahden skannauksen tuloksena muodostettujen runkomallien tilavuudet V erosivat yhden skannauksen tuottamia tilavuuksia enemmän puunrungon vertailumallin tilavuudesta. Tähän voivat vaikuttaa kahden laserskannauksen tuottamien runkopistepilvien 3 ja 4 pienemmät havaitut ϵ -arvot eli hyvien havaintojen osuudet havaintodatassa runkopistepilviin 1 ja 2 verrattuna. Mittausdatan lisääminen voi siis huonontaa mallinnustuloksia, jos lisättävän datan laadusta ei pystytä varmistumaan. Koska käytössä olleiden kuuden laserskannauksen tarkempia tietoja ei tässä työssä ole saatavilla, ei ole mahdollista määrittää, mikä eri skannausten laatuun on vaikuttanut. Skannauspisteen 1 rinnalle otetun skannauspisteen 3 laserkeilaus on esimerkiksi voinut tapahtua suunnasta, jossa laserskannerin ja puunrungon pinnan välinen näkyvyys on huono, tai skannausten 1 ja 3 tuottamien datapisteiden rekisteröinti eli sovittaminen samaan koordinaatistoon on voinut epäonnistua. Tulosten perusteella vaikuttaa siltä, että jos puita kuvattaessa on mahdollista suorittaa useita kuvauksia eri suunnista mutta jos tulokset ovat jostain suunnasta huonolaatuisempia kuin toisista tai tuloksia on vaikeaa sovittaa yhteen, kannattaa huonompilaatuiset pisteet jättää kokonaan pois mallinnuksesta. Edellä nähtiin taulukon 5.3 tulosten perusteella, että ainakin tässä työssä käytetyt menetelmät ja laserkeilausresoluutiot tuottavat mediaanitilavuudeltaan hyviä runkomalleja, vaikka käytössä olisi vain yksi hyvälaatuinen laserkeilaussuunta. Iteraatiokierroksilla k mitattuna menetelmät suoriutuivat suhteellisesti nopeammin kahden laserskannauksen pistepilvistä 3 ja 4, kun pienemmän ϵ -arvon vaikutukset iteraatiomääriin otetaan huomioon. Absoluuttiset iteraatiomäärät olivat kuitenkin suurempia kuin yhden laserskannauksen pistepilvien kohdalla.

RANSAC jatkoi rungoille 3 ja 4 tasaista suoriutumista. Se ei ollut millään arvioitavilla muuttujan arvoilla menetelmistä huonoin, muttei myöskään paras. LMS oli RANSACin kanssa samaa tasoa. Kt-menetelmän suoriutuminen ylsi runkojen 3 ja 4 kohdalla, kuten aiemmin runkojen 1 ja 2 kohdalla, RANSACin kanssa joko samalle tai korkeammalle tasolle. MSAC oli runkojen 1 ja 2 tapaan runkojen 3 ja 4 kohdalla iteraatiomääriltään k hitain mutta tilavuustuloksiltaan V tarkin, ja se ilmoitti myös muista menetelmistä poiketen vähiten poikkeavia sylintereitä. $T_{d,d}$ -menetelmä oli kokonaisuutena toiseksi hitain kierrosmäärissä k , sen sovittamien runkojen tilavuudet V olivat tarkkuudessaan menetelmien häntäpäässä ja sen tuloksissa oli paljon hajontaa. Tämä on poikkeus sylinterisovitusten tuloksiin luvussa 5.1.1, jossa $T_{d,d}$ -menetelmän tuottamien sylinterisovitusten hajonta oli menetelmien

joukossa pienimmästä päästä.

6. PÄÄTELMÄT JA YHTEENVETO

Seuraavaksi vedetään yhteen työn tähänastinen sisältö, palataan luvussa 1 asetettuihin tutkimuskysymyksiin ja reflektoidaan työn rajoituksia ja jatkokehitysideoita. Ensiksi käsitellään työn mallinnustuloksia yleisellä tasolla ja tämän jälkeen jokaisen toteutetun menetelmän suoriutumista yksitellen.

Luvussa 5.1.1 nähtiin, että menetelmien testaamista varten generoituihin sylinteripistepilviin 1 ja 2 sovitettiin usein säteeltään r huomattavasti liian suuria ja usein myös todellisesta sylinteristä akselinsa suunnalta poikkeavia sylinterimalleja. Tämä saattaa kertoa liian suuresta virhetoleranssitasosta t , jonka arvoa olisi syytä optimoida. Aitoa puunrungoista saatua mittausdataa käsiteltäessä onnistuttiin tuottamaan hyviä puunrunkomalleja verrattaen vähäiselläkin laserkeilausdatalla, kun sovitettavien sylinterien akselien \vec{a} kulma rajoitettiin alle 45 asteen suuruiseksi edeltävän puunrungon osaan verrattuna ja säteen r suuruus alimmillaan puoleen ja korkeimmillaan kaksinkertaiseksi edellisen puunrunkosylinterin säteestä. Rajoitteiden asettaminen kuitenkin kaventaa mahdollisuuksia soveltaa menetelmiä geometrisesti hyvin monimuotoiseen puustoon.

Työssä toteutettiin 7 konseptuaalisesti yksinkertaista ja matemaattisesti helposti lähestyttävää versiota RANSAC-menetelmästä, ja ne kaikki sovittivat vain yhdestä laserskannauksesta koostuvaan pistepilvidataan puunrunkomallit, joiden tilavuus V poikkesi alle 10 % todelliseksi oletetusta tilavuuden vertailuarvosta. Työn alussa asetetuista tutkimuskysymyksistä toinen kysyi, miten puunrungon mallinnustulokset muuttuvat, kun puunrungon pistepilven resoluutiota ja käytössä olleiden laserskannausten määrää varioidaan. Pienin käytetty pistepilvidatan resoluutio oli 0,5 datapistettä per cm^2 . Sen kasvattaminen moninkertaiseksi tuotti mediaaniltaan samanlaisia mallinnustuloksia, mutta vähensi tulosten hajontaa. Resoluution kasvattamisen voidaan siis tulkita tuottavan luotettavammin tarkkoja tuloksia, mikä on työn toteutusta suunnitellessa voimassa olleen oletuksen mukaista. Oletuksen vastaisesti mukaan otettu toinen laserskannaus tulosten parantamisen sijaan taas heikensi niitä, joten tulosten perusteella yksi hyvälaatuinen skannaus on parempi kuin useampi mahdollisesti huonolaatuinen tai huonosti yhteensovitettu skannaus.

Muodostettujen runkomallien tilavuuksista V ylivoimainen enemmistö oli vertailuarvoa pienempiä, vaikka sovitettaessa yksittäisiä sylintereitä generoituun dataan virheelliset säteet r olivat yleensä todellista sädettä suurempia. Yksittäisiä sylintereitä sovitettaessa

kuitenkin nähtiin myös, että sylinterien pituus l jäi yleensä todellista pistepilven korkeutta ja sen pohjalla olevaa sylinterimallia lyhyemmäksi. Sama saattaa olla toistunut runkopistepilvien kohdalla ja voi selittää tilavuuden poikkeamaa todellisesta arvosta alaspäin. Runkomallien sylinterien jääminen usein odotettua lyhyemmäksi voi myös nostaa runkomalleissa havaittujen poikkeavien sylinterien määrää an , sillä yksi sylinterin poikkeavuuden kriteereistä oli peräkkäisten sylinterien suuri etäisyys toisiinsa. Käytetty ohjelma pyrki asettamaan sylinterien alkupisteet aina 1 mittayksikön välein toisistaan, vaikka välissä olisi lyhyempiä sylintereitä.

Toinen työn tutkimuskysymyksistä oli se, millaisia toteutettujen RANSAC-versioiden mallinnustulokset ja tehokkuus ovat toisiinsa verrattuina. Työssä toteutettu RANSACin standardiversio suoriutui sekä nopeudeltaan että tarkkuudeltaan muita toteutettuja menetelmiä paremmin käsiteltäessä haastavaa sylinteripistepilvää 2, jossa ϵ -arvo oli alle 0,4 eli poikkeavien havaintojen osuus yli 60 % datasta. Parempilaatuista pistepilvidataa eli sylinteriä 1 sekä runkopistepilviä 1–4 käsiteltäessä taas RANSACin suoriutuminen oli keskitasoa verrattuna toteutettuihin modifikaatioihin, joista monien avulla saavutettiin parannuksia tuloksiin.

MSAC-menetelmän tulokset olivat helpomman sylinteripistepilven 1 kohdalla menetelmistä parhaimmista, mutta sylinteripistepilven 2 kohdalla tulokset eivät olleet lähelläkään vertailuarvoja ja käytetyt iteraatiokierrosmäärät olivat valtavia. Runkoja mallinnettaessa MSACin tulokset olivat kaikista toteutetuista menetelmistä kärkitasoa rungon tilavuuden V määrittämisessä. Vastoin Torrin ym. [13] väittämää MSAC vaikuttaa saavuttaneen nämä tarkkuusparannukset suuremman iteraatiomäärän k kustannuksella. Iteraatiomäärää k päivitettiin työn toteutuksessa iteroinnin aikana löydettyjen konsensusjoukkojen C_i kokojen perusteella. MSACin ilmoittamien lopullisten hyvien havaintojen osuuksien ϵ perusteella runkopistepilvissä 1–4 MSAC löysi runkopistepilvistä pienempiä konsensusjoukkoja muihin menetelmiin verrattuna, mikä on johtanut sen käyttämiin suurempiin iteraatiokierrosmääriin. Pienempien konsensusjoukkojen tuottaminen on todennäköisesti seurausta MSAC-kustannusfunktion ominaisuuksista: RANSAC-kustannusfunktion kohdalla aiempaa suurempi löydetty konsensusjoukko C_i tuottaa automaattisesti aiempaa pienemmän kustannusfunktion arvon F_i , mutta MSAC-kustannusfunktion kohdalla näin ei ole. Suuremmasta konsensusjoukosta C_i eli pienemmästä poikkeavien havaintojen $P \in D$ määrästä huolimatta on mahdollista, että löydettyjen hyvien havaintojen $H \in D$ mallivirheidien neliöiden e_j^2 yhteenlaskettu suuruus kasvaa niin paljon, että niiden tuottama suurempi summa $\sum e_j^2$ kompensoi poikkeavien havaintojen tuottaman pienemmän summan $\sum t^2$ osuuden MSAC-kustannusfunktiossa yhtälössä 3.4 ja uutta mallia M_i ei oteta käyttöön. Koska uutta suurempaa konsensusjoukkoa C_i ei aina oteta käyttöön sellaisessa tilanteessa jossa RANSACia käytettäessä otettaisiin, iteraatiokierrosmäärää k ei aina päivitetä alaspäin sellaisessa tilanteessa jossa RANSACia käytettäessä päivitetäisiin.

Myös Kt-uudelleensovitusmenetelmä suoriutui runkoja mallinnettaessa tilavuuden V tark-

kuudessa ja iteraatiokierrosten k määrässä joko RANSACin kanssa samantasoisesti tai sitä paremmin, vaikka generoidulla sylinteripistepilvidatalla testattaessa sen tulokset olivat olleet joko RANSACin kanssa samaa tasoa tai huonompia. Se vaikuttaa siis parantaneen tositalanteessa tulosten tarkkuutta hieman. Työn koodissa laskenta-ajassa mitattuna kallein osuus oli kuitenkin sylinterimallin sovittaminen pistepilvidataan Gauss-Newton-optimoinnilla, joten tulosparannukset eivät välttämättä olleet tarpeeksi suuria oikeuttaakseen syntyneet laskenta-aikakustannukset siitä, että toteutetussa Kt-menetelmässä malli sovitetiin uudelleen neljä kertaa RANSACin standardiversion yhteen uudelleensovituskertaan verrattuna.

$T_{d,d}$ -testillä varustettu RANSAC-versio ei yksittäisiä sylintereitä generoituihin sylinteripistepilviin 1 ja 2 sovitettaessa käyttänyt selkeästi suurempia määriä iteraatiokierroksia k RANSACin standardiversion verrattuna, mutta runkoja mallinnettaessa tämä ominaisuus tuli näkyviin. Käytetyt iteraatiokierrosmäärät k olivat yleisesti kokonaisia runkoja mallinnettaessa kaavan 3.8 ennustamia arvoja jonkin verran suurempia kaikkien menetelmien kohdalla. $T_{d,d}$ -testi lisäsi runkoja mallinnettaessa sekä keskimäärin käytettyjä iteraatiokierroksia k että k -arvojen hajontaa reilusti RANSACiin verrattuna. Testiä kannattaakin käyttää vain tilanteissa, joissa konsensusjoukkojen muodostaminen vie niin paljon resursseja, että tämän askeleen tarpeettomien suorituskertojen vähentäminen kompensoi ylimääräisistä iteraatiokierroksista k ja niistä aiheutuvista ylimääräisistä mallisovituksista M_i ylimääräisiin satunnaisotoksiin S_i syntyviä kustannuksia. Koska tämän työn toteutuksessa mallien sovittaminen oli aikaavievin osuus ja konsensusjoukkojen muodostaminen siihen verrattuna nopeaa, on todennäköistä, että $T_{d,d}$ -testin käyttö ei tässä työssä alentanut laskenta-aikaa vaan pikemminkin nosti sitä. Työssä ei pidetty lukua siitä, kuinka monta mallia $T_{d,d}$ -testin perusteella hylättiin, mikä olisi ollut oleellista suorituskyvyn arvioinnin kannalta niissä tilanteissa, joissa konsensusjoukkojen muodostaminen on kallista.

Inner-RANSAC ja inner-RANSAC-Kt-yhdistelmä toteuttivat lupaavien hypoteesimallien M_i uudelleensovituksen Kt-menetelmän tavoin RANSACiin verrattuna pidennetyksi ja Kt-menetelmään verrattuna vielä perusteellisemmin. Ne sovittivat parempilaatuiseen sylinteripistepilveen 1 menetelmistä parhaat mallit ja inner-RANSAC-Kt-yhdistelmä oli sovituksessa myös menetelmistä nopein, mutta haastavamman sylinteripistepilven 2 kohdalla niiden tulokset romahtivat. Runkomallien sovittamisessa runkopistepilviin 1–4 ne olivat muihin menetelmiin verrattuna iteraatiomäärissä k mitattuna nopeammasta päästä, mutta ero oli pieni. Tuotettujen runkomallien tilavuusarvot V olivat inner-RANSACille pääosin RANSACia hieman huonompia ja inner-RANSAC-Kt-yhdistelmälle RANSACin kanssa kokonaiskvaltaan samaa tasoa.

Jotta Kt-menetelmän, inner-RANSACin ja inner-RANSAC-Kt-yhdistelmän tehokkuutta RANSACiin verrattuna olisi voitu todella arvioida, olisi niiden tekemien työn toteutuksessa laskennallisesti kalliiden mallisovitusten määrä kannattanut mitata. Koska Kt-menetelmässä jokaiseen k kappaleeseen satunnaisotoksia S_i sovitetiin hypoteesimalli M_i ja lisäksi pa-

ras siihen asti löydetty malli sovitettiin aina uudelleen 4 kertaa, on Kt-menetelmän tekemien mallisovitusien määrä tässä työssä $k + 4M$, kun M on sellaisten hypoteesimallien M_i määrä, jotka löytyessään ovat olleet siihen asti parhaita. Inner-RANSACin tekemien mallisovitusien määrä työssä käytetyllä iteraatiomäärällä $k_{inner} = 10$ on vastaavasti $k + 10M$, ja inner-RANSAC-Kt-yhdistelmän mallisovitusien määrä on $k + 10M + 4L$, missä L on inner-RANSAC-uudelleensovitusiteraatioiden sisällä parhaiden toistaiseksi löydettyjen mallien määrä. Määrittämällä muuttujien M ja L tyypillisiä arvoja voitaisiin paremmin arvioida sitä, kuinka paljon matalampi lopullisten iteraatiokierrosten määrä k tulisi näille menetelmille olla, jotta niiden tekemien ylimääräisten mallisovitusien kustannukset RANSACiin verrattuna tulisivat kompensoiduiksi.

Pienimmän neliösumman korvaaminen sylinterisovituksissa pienimmällä neliöiden mediaanilla tuotti LMS-menetelmän. Sen tuloksissa oli yksittäisiä sylintereitä generoituihin sylinteripistepilviin sovitettaessa paljon hajontaa, joten se näytti voimistavan RANSAC-sovitustuloksissa esiintyvää satunnaisvaihtelua. Runkopistepilvistä runkomalleja muodostettaessa tämä efekti kuitenkin hävisi, ja LMS suoriutui runkojen mallinnuksesta keskitasoisesti muihin menetelmiin verrattuna ja sen tulokset olivat RANSACin standardiversion kanssa samaa luokkaa.

Työn tuloksia tarkasteltaessa tulee ottaa huomioon, että RANSAC-menetelmät perustuvat satunnaisuuteen ja todennäköisyyksiin. Siksi on mahdollista, että RANSACin näennäisesti muihin verrattuna parempi suorituskyky työn ainoalla alle 50 % hyviä havaintoja sisältävällä havaintoaineistolla eli sylinteripistepilvellä 1 on sattumaa ja testaaminen kannattaisi suorittaa muutamilla eri aineistoilla ja useammilla toistokerroilla. Työssä ei ole käytetty menetelmien testaamiseen useita erikokoisia sylintereitä, vaikka käytetyssä puunrungossa esiintyykin sen matkalta hieman vaihtelua säteen suuruudessa r , ja menetelmien testaaminen eri muotoisilla sylintereillä saattaisi tuoda lisää tietoa. Kohinaa ja poikkeavien havaintojen määrää ei ole työn pistepilvissä myöskään tutkittu erikseen, kuten esimerkiksi Schnabelin ym. [8] artikkelissa on tehty. Näiden erittely ja arvojen testaaminen erikseen voisi tuoda hyödyllistä tietoa siitä, minkälaisille havaintoaineistoille menetelmiä kannattaa soveltaa. Käytetystä runkopistepilvidatasta on tarkempien tietojen puutteessa tehty usein oletuksia, jotka voivat vaikuttaa pistepilvien käsittelyyn ja tulosten tulkitsemiseen, ja näitä oletuksia on pyritty tuomaan esiin siellä missä se on ollut relevanttia.

Virhetoleranssitason t tunnollisemman määrittämisen sekä kohinan ja poikkeavien havaintojen erittelyn lisäksi satunnaisotosten valikoinnin ohjailuun voisi kiinnittää huomiota. Esimerkiksi otosten hajauttaminen saattaisi edistää otospisteiden sijoittumista sylinterin koko kaaren laajuudelle, mikä voisi helpottaa säteen r arvon määrittämistä. Myös laserkeilausdatan signaalien voimakkuuksien tarkastelu eri havaintopisteiden kohdalla ja havaintopisteiden luokittelu sen perusteella voisi mahdollistaa jonkinlaista otostestausta ja vain lupaavien satunnaisotosten käsittelyä. Tämä voisi nopeuttaa ohjelman suoritusaikaa tässä työssä käsitellyn kaltaisissa tilanteissa, joissa mallien sovittaminen otoksiin on

aikaavievää. Myös luvussa 2.1 mainittu puunrunkoon sovitettavien sylinterien pituuden joustava määrittäminen olisi mahdollisuus sekä pyrkiä parantamaan tämän työn mallinustulosten tarkkuutta että tehdä käytetyistä menetelmistä laajemmin sovellettavia.

LÄHTEET

- [1] Jean-François Côté, Richard A. Fournier, and Richard Egli. “An architectural model of trees to estimate forest structural attributes using terrestrial LiDAR”. In: *Environmental Modelling & Software: with Environment Data News* 26.6 (2011), pp. 761–777.
- [2] Tuomas Yrttimaa et al. “Detecting and characterizing downed dead wood using terrestrial laser scanning”. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 151 (2019), pp. 76–90.
- [3] Zhonghua Su et al. “Extracting Wood Point Cloud of Individual Trees Based on Geometric Features”. In: *IEEE Geoscience and Remote Sensing Letters* 16.8 (2019), pp. 1294–1298.
- [4] Martin A. Fischler and Robert C. Bolles. “Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography”. In: *Communications of the ACM* 24.6 (1981), pp. 381–395.
- [5] Xing Fang et al. “On the total least median of squares adjustment for the pattern recognition in point clouds”. In: *Measurement: Journal of the International Measurement Confederation* 160 (2020), pp. 107794–.
- [6] Zhonghao Xie et al. “Partial least median of squares regression”. In: *Journal of chemometrics* 36.8 (2022).
- [7] Peter J. Rousseeuw. “Least Median of Squares Regression”. In: *Journal of the American Statistical Association* 79.388 (1984).
- [8] R. Schnabel, R. Wahl, and R. Klein. “Efficient RANSAC for Point-Cloud Shape Detection”. In: *Computer graphics forum* 26.2 (2007), pp. 214–226.
- [9] Rahul Raguram et al. “USAC: A Universal Framework for Random Sample Consensus”. In: *IEEE transactions on pattern analysis and machine intelligence* 35.8 (2013), pp. 2022–2038.
- [10] Kaijun Peng, Jieqing Tan, and Guochang Zhang. “A Method of Curve Reconstruction Based on Point Cloud Clustering and PCA”. In: *Symmetry (Bandung)* 14.4 (2022), pp. 726–.
- [11] Rahul Raguram, Jan-Michael Frahm, and Marc Pollefeys. “A Comparative Analysis of RANSAC Techniques Leading to Adaptive Real-Time Random Sample Consensus”. In: *Computer Vision – ECCV* 5303.1 (2008), pp. 1–2.
- [12] B. J. Tordoff and D. W. Murray. “Guided-MLESAC: Faster Image Transform Estimation by Using Matching Priors”. In: *IEEE transactions on pattern analysis and machine intelligence* 27.10 (2005), pp. 1523–1535.

- [13] P. H. S. Torr and A. Zisserman. “MLESAC: A New Robust Estimator with Application to Estimating Image Geometry”. In: *Computer Vision and Image Understanding* 78.1 (2000), pp. 138–156.
- [14] J. Matas and O. NChum. “Randomized RANSAC with $T_{d,d}$ test”. In: *Image and Vision Computing* 22.10 (2004), pp. 837–842.
- [15] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004. 559 pp.
- [16] Clément Riu, Vincent Nozick, and Pascal Monasse. “Automatic RANSAC by Likelihood Maximization”. In: *Image processing on line* 12 (2022), pp. 27–49.
- [17] *MATLAB code for fitting cylinders to tree trunk data as a part of a Finnish Master’s thesis work*. Nov. 22, 2023. URL: <https://doi.org/10.5281/zenodo.10184507> (visited on 11/22/2023).