

Samanaikainen paikannus ja kartoitus  
autonomisilla drooneilla korkea- ja  
matalastruktuurisissa ympäristöissä

TURUN YLIOPISTO  
Tietotekniikan laitos  
TkK-tutkielma  
Tietotekniikka  
Huhtikuu 2026  
Eemil Einola

TURUN YLIOPISTO  
Tietotekniikan laitos

EEMIL EINOLA: Samanaikainen paikannus ja kartoitus autonomisilla drooneilla  
korkea- ja matalastruktuurisissa ympäristöissä

TkK-tutkielma, 29 s.  
Tietotekniikka  
Huhtikuu 2026

---

Tässä tutkielmassa tarkastellaan SLAM:ia hyödyntävän autonomisen droonin toimintakykyä eri ympäristöissä. Tutkielmassa keskitytään Stereo-SLAM:iin, mutta käsitellään myös muita menetelmiä. Stereo-SLAM:iin keskitytään sen laajimman käyttömahdollisuuksien vuoksi. Stereo-SLAM käyttää paikannuksessa perinteisen GPS:n sijasta kameroita, jotka mahdollistavat etäisyyksien laskennan ja luotettavan mallinuksen ympäristöstä. Tämä mahdollistaa autonomisen droonin käytön myös signaalia häiritsevässä ympäristöissä, kuten kaupungeissa ja sisätiloissa. Stereo-SLAM:in luotettavuutta tarkastellaan siirtymävirheen perusteella eri testiympäristöissä. Tutkielmassa tarkastellaan muun muassa KITTI:n, EuRoC:in ja Blackbird:in tarjoamia testiympäristöjä.

Tuloksista ilmenee droonin parempi suorituskyky valoisissa korkeastruktuurisissa ympäristöissä. Tutkielmassa esitetyt algoritmit saavuttavat alle yhden prosentin virhearvion ideaali ympäristöissä. Matalastruktuurinen ympäristö ilmenee kuitenkin haastavammaksi ympäristöksi eri piirteiden, kuten avainpisteiden puutteellisuuden myötä. Erojen pienentämiseksi on kehitelty avainpisteitä havaitsevien algoritmien lisäksi linjapiirteitä tunnistavia algoritmeja. Parannetuilla algoritmeilla saavutetaan jopa alle 1.5% virhearvioita haastavimmissa skenaarioissa. Tutkimus suoritettiin kirjallisuuskatsauksena.

Asiasanat: SLAM, drooni, stereo, KITTI, EuRoC, Blackbird, korkeastruktuuri, matalastruktuuri

# Sisällys

<b>1 Johdanto</b>	<b>1</b>
<b>2 Samanaikainen paikannus ja kartoitus</b>	<b>4</b>
2.1 SLAM:in toimintaperiaate ja menetelmät . . . . .	4
2.2 Stereonäköön perustuva SLAM . . . . .	8
2.2.1 Ominaisuuksiin pohjautuvat algoritmit . . . . .	9
2.2.2 Suorat ja puolisuorat algoritmit . . . . .	10
<b>3 Stereo-SLAM:ia hyödyntävän droonin tulokset eri toimintaympäristöissä</b>	<b>12</b>
3.1 Korkeastruktuurinen ympäristö . . . . .	13
3.2 Matalastruktuurinen ympäristö . . . . .	17
<b>4 Autonomisen droonin luotettavuus, haasteet ja ratkaisut</b>	<b>19</b>
4.1 Tulosten analyysi . . . . .	19
4.2 Parannusmahdollisuudet . . . . .	24
<b>5 Pohdinta</b>	<b>26</b>
<b>6 Yhteenveto</b>	<b>28</b>
<b>Lähdeluettelo</b>	<b>30</b>

# Kuvat

2.1	SLAM-menetelmän toimintavaiheet . . . . .	5
4.1	Visualisoidut siirtymävirheet viidessä eri KITTI-sekvenssissä ©2025 [24]. . . . .	22
4.2	Droonin arvioima lentoreitti (mustalla) verrattuna todelliseen reit- tiin (punaisella) V101, V202, MH03 ja MH05 EuRoC-ympäristöissä ©2011 IEEE. . . . .	23

# Taulukot

2.1	Tutkielmassa käytetyt SLAM-algoritmit ja niiden perusominaisuudet sekä piirteet . . . . .	8
3.1	Perusominaisuudet valituista SLAM-tietoaineistoista . . . . .	12
3.2	Stereo-SLAM-algoritmien RMSE-tuloksia 10:ssä valitussa KITTI-sekvenssissä [14] . . . . .	13
3.3	Stereo-SLAM-algoritmien RMSE-tuloksia 10:stä valitusta KITTI-sekvenssistä eri parametrimuunnelmilla [10]. . . . .	14
3.4	Stereo-SLAM-algoritmien RMSE-keskiarvot sekä onnistumisprosentti Backbird-testiympäristössä [23] . . . . .	15
3.5	Stereo-SLAM-algoritmien siirtymävirheet valituissa EuRoC-tietoaineiston sekvensseissä . . . . .	16
3.6	Eri Stereo-SLAM-algoritmien tulokset matalastrukturisessa KITTI 01 -sekvenssissä . . . . .	17
3.7	SLAM-algoritmien siirtymävirheet Gomez Ojedan ja muiden esittämässä testiympäristössä . . . . .	17
3.8	Stereo-algoritmien siirtymävirheet EuRoC-tietoaineistossa . . . . .	18
3.9	Stereo-SLAM:in siirtymävirheet Rosario-tietoaineistossa [22]. . . . .	18
4.1	Stereo-algoritmien siirtymävirheiden lasketut keskiarvot hyödyntäen seitsemän eri testiympäristön tuloksia . . . . .	19

# 1 Johdanto

Miehittämättömiä lennokkeja (engl. *unmanned aerial vehicles*, UAV) hyödynnetään yhä laajemmin erilaisissa tehtävissä, kuten sotilastiedustelussa, ilmakartoituksessa sekä maatalouden eri sovelluksissa [1], [2]. Nykypäivänä kuitenkin suurin osa UAV-järjestelmistä käyttää paikannukseen maailmanlaajuisista satelliittinavigointijärjestelmää (engl. *Global Navigation Satellite System*, GNSS) [3]. Urbaanisessa sekä tiheään kasvillisuuden alueilla GNSS:n tarkkuus heikkenee huomattavasti signaalien esteiden ja heijastumisen vuoksi [3]. Lisäksi ulkoinen signaalien häirintä on aiheuttanut ongelmia GNSS-pohjaisille laitteille.

Ongelman ajankohtaisuuden vuoksi tässä tutkielmassa tarkastellaan yhtä vaihtoehtoista paikannusmenetelmää: samanaikainen paikannus ja kartoitus (engl. *Simultaneous Localization and Mapping*, SLAM). Erityisesti keskitytään stereonäköön perustuvan SLAM:in (engl. *Stereo-Visual -SLAM*) toimivuuteen UAV-järjestelmissä korkea- ja matalastruktuurisissa ympäristöissä. Stereo-SLAM hyödyntää visuaalista havaintoa ympäristöstä, mikä mahdollistaa riippumattomuuden ulkoisiin majakoihin [4]. Tämän ansiosta voidaan tuottaa tarkempia paikannustietoja ympäristöistä, joissa paikantaminen on haasteellisempaa, kuten kaupungeissa.

Tämä tutkielma on suoritettu kirjallisuuskatsauksena, jonka päätavoitteena on selvittää Stereo-SLAM:in toimintaa ja suorituskykyä korkea- sekä matalastruktuurisissa ympäristöissä. Näiden tietojen pohjalta tutkimuskysymyksiksi valikoitui seuraavat kysymykset:

- Tutkimuskysymys 1: Mitä keskeisiä menetelmiä SLAM-teknologiaan kuuluu?
- Tutkimuskysymys 2: Kuinka luotettavasti stereonäköön perustuva SLAM toimii autonomisten dronien paikannuksessa korkea- sekä matalastruktuurisissa ympäristöissä?
- Tutkimuskysymys 3: Kummassa toimintaympäristössä Stereo-SLAM-menetelmä on ideaalimpi?

Tutkielmassa hyödynnetyt artikkelit ovat saatu kahdesta eri tietokannasta: Sage Journalista ja IEEE:stä. Osa artikkeleista on saatu jo valittujen artikkelien lähde-luetteloista. Tutkielmassa on hyödynnetty KEENIOUS:ta, mikä on tekoälypohjainen tietokanta, sekä Turun yliopiston VOLTER-tietokantaa. Pääosin artikkelit ovat julkaistu 2020 jälkeen, jotta tutkielmasta saisi mahdollisimman ajankohtaisen kuvan. Tutkielmassa on kuitenkin hyödynnetty vanhempia alalla standardeiksi muodostuneita tietoaineistoja ja artikkeleita, jotka ovat vielä nykypäivänäkin alalla relevantteja. Tutkimuksessa käytetään myös vanhempaa aineistoa silloin, kun niiden testitulokset todentuvat paremmiksi kuin uudemmissa julkaisuissa esitetyt tulokset.

Tiedonhaku aloitettiin yhdistämällä aiheeseen liittyvää termistöä yhteen englannin kielellä, esimerkiksi sanoilla Drone, SLAM, Dataset, Urban. Pienten kokeilujen jälkeen molempiin tietoaineistoihin hyödynnettiin omia hakulausekkeita. IEEE-tietoaineistoon valikoitui kaksi lausetta: (stereo\* AND (urban\* OR natur\* OR vege\*) AND (drone OR UAV) AND SLAM) sekä (STEREO AND SLAM AND dataset AND (Dynamic OR challeng\* OR texture)). Kyseisillä hakulausekkeilla löytyi IEEE:stä 288 hakutulosta. Karsinta suoritettiin aluksi julkaisuvuoden perusteella. Tutkielmaan otettiin ainoastaan vuosina 2023-2025 julkaistut artikkelit huomioon, mistä jäi jäljelle yhteensä 136 artikkelia. Tämän jälkeen otsikoiden ja tiivistelmän perusteella suoritettiin seuraava karsinta. Viimeiseksi artikkeleita karsittiin JUFO-luokitusten mukaisesti. Sage Journalsia käytettiin ainoastaan lisätietoaineistojen et-

sintään, minkä myötä hakulausekkeeksi valikoitui: (SLAM AND Dataset AND Stereo). Hakutulos tuotti 2018-2025 aikarajoituksella 103 tulosta, jota karsittiin ensin otsikkotasolla, ja sitten tiivistelmä tasolla. Näiden jälkeen valituista kuudesta artikkelista karsittiin tulostaulukoiden perusteella. Lopuksi tutkielmassa hyödynnettiin 16 eri artikkelia, jotka koostuivat kahdesta Sage Journals artikkelista, viidestä IEEE artikkelista, kolmesta Keenious artikkelista, sekä kuudesta artikkelista, jotka ovat saatu jo valituista artikkeleista.

Tutkielman toisessa luvussa käydään läpi SLAM:in toimintaperusteet sivuten muita sensoritekniikoita, minkä jälkeen syvennyttään stereonäön toimintaperusteisiin. Kolmannessa luvussa esitetään Stereo-SLAM:in paikannustarkkuuden tulokset sekä testiympäristöt. Tämän jälkeen neljännessä luvussa analysoidaan tutkielmassa esitetyt tulokset ja esitetään mahdollisia ratkaisumenetelmiä. Tutkielma keskittyy tarjoamaan lukijalle ajankohtaisen kuvan stereonäköön perustuvan SLAM:in toimintakyvystä eri toimintaympäristöissä.

# 2 Samanaikainen paikannus ja kartoitus

SLAM yhdistää reaaliaikaisen agentin paikannuksen sekä kartoituksen yhdeksi prosessiksi, mikä mahdollistaa UAV-järjestelmien toiminnan myös haastavammissa ja dynaamisemmissä ympäristöissä [5], [6]. SLAM on siis tapa paikantaa agentin liikettä ja samalla kartoittaa agentin ympäristöä ennestään tuntemattomassa ympäristössä. SLAM-algoritmit koostuvat yleisesti kahdesta komponentista: paikannuksesta ja kartoituksesta [6]. Paikannus perustuu agentin sijainnin arviointiin kartoitetussa ympäristössä, mikä tapahtuu hyödyntämällä todennäköisyyspohjaisia menetelmiä. Kartoituksella taas tarkoitetaan ympäristön mallintamista sensoreista saatavan datan pohjalta.

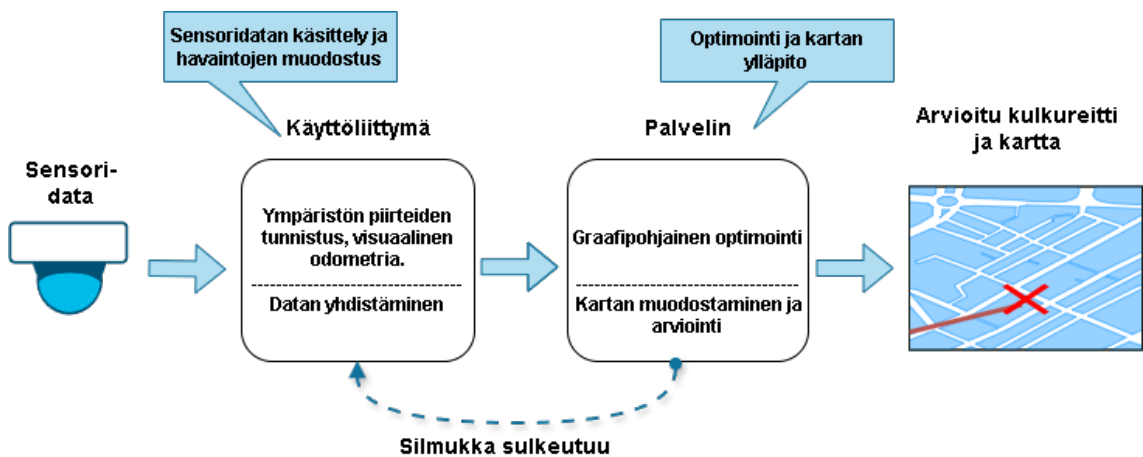
SLAM-menetelmät luokitellaan tavallisesti niissä hyödynnettyjen sensorien perusteella. GNSS-haasteiden korvaamiseksi esiin on noussut etenkin näköön perustuvat SLAM-menetelmät sekä laserpohjaiset -menetelmät, kuten valotutka [1]. Kyseisistä menetelmistä varsinkin visuaalinen SLAM on yleistynyt sen erinomaisen ympäristön havaitsemiskyvyn ja kameroiden alhaisten kustannusten myötä [1].

## 2.1 SLAM:in toimintaperiaate ja menetelmät

SLAM-algoritmi voidaan jakaa käyttöliittymäpuoleen (engl. *Frontend*) sekä palvelinpuoleen (engl. *Backend*) [7]. Käyttöliittymäprosessissa algoritmi hyödyntää senso-

reista, kuten kameroista tai valotutkasta, saatua tietoa. Sensoreilla kerätään havain-  
toja dronin peräkkäisten sijaintien väliltä eli odometriaa sekä dronin ja ympäris-  
tön maamerkkien välisiä etäisyyksiä [8]. Näiden havaintojen avulla palvelinpuolella  
SLAM-algoritmi arvioi laitteen oman sijainnin muodostaen samalla kartan ympäris-  
töstään optimointitekniikan avulla, kuten graafipohjaisella optimoinnilla [7], [8].

SLAM-algoritmissa voidaan myös hyödyntää silmukan sulkuvaihetta (engl. *Loop closure*)  
optimoimisen edistämiseksi [7]. Tarkoituksena on tunnistaa jo käyty sijain-  
nit vertailemalla uutta sensoridataa vanhaan [3], [6], [7]. Datan täsmätessä algorit-  
mi päivittää kartta- ja sijaintiarviot sen mukaisesti [6]. Prosessi parantaa sijain-  
nin luotettavuutta sekä tarkkuutta. Kuvassa 2.1 visuaalisesti havainnoidaan SLAM-  
algoritmin toimintavaiheet SLAM:in selventämiseksi.



Kuva 2.1: SLAM-menetelmän toimintavaiheet

Kuvasta 2.1 voidaan havaita, että sensoridata on SLAM-menetelmän olennai-  
nen osa, minkä vuoksi menetelmät luokitellaan usein käytettyjen sensorien perus-  
teella [9]. Tässä tutkielmassa keskitytään ensisijaisesti stereonäköön perustuvaan  
SLAM:iin. Kuitenkin SLAM-järjestelmän kokonaiskuvan hahmottamiseksi ja stereo-  
tekniikan valinnan perustelujen ymmärtämiseksi on hyödyllistä tarkastella yleisim-

piä sensoritekniikoita. Näitä ovat muun muassa näköön perustuvat sensorit, joihin lukeutuvat erilaiset kamerat sekä laserpohjainen sensoritekniikka valotutka [9].

**Valotutka-SLAM** eli valon havainnointi ja etäisyyden mittaaminen (engl. *Light Detection And Ranging*, LiDAR) on yksi käytetyimmistä sensoreista SLAM-tekniikassa. SLAM:issa hyödynnetään yleensä joko 2D- tai 3D-valotutkaa [9]. Molemmat mallit hyödyntävät useampaa laseria ympäristön tiedon keräämiseen. 2D-valotutka ei kuitenkaan pysty mittaamaan esteiden korkeutta, sillä laserit rajoittuvat vaakasuuntaan. Sen sijaan 3D-valotutka hyödyntää useampaa laseria eri tasoilla, mikä mahdollistaa kolmiulotteisen rakennedatan esteistä. Valotutkan suorituskyky voi kuitenkin huomattavasti heikentyä tietyissä valaistusolosuhteissa, kuten erittäin kirkkaassa valossa esteen ollessa voimakkaasti heijastava [3], [9]. Valotutkan luokitellaan myös olevan huomattavasti hintavampi tekniikka verrattuna visuaalisiin menetelmiin [7].

**Monokulaarinen-SLAM** (engl. *monocular SLAM*) kuuluu visuaalisiin SLAM-menetelmiin, mikä hyödyntää vain yhtä kameraa ympäristön tiedon keräämiseen [9]. Monokulaariset kamerat ovat laajasti hyödynnettyjä niiden kompaktisuuden ja hintatehokkuuden myötä [9], [10], [11]. Kuitenkin vain yhden kameran hyödyntämisellä ei voida arvioida syvyyttä ympäristöstä, syntyy epävarmuutta ympäristön mittasuhteista sekä arvioidusta kulkureitistä [9], [10].

**Syvyyskamera-SLAM** (engl. *Red, Green, Blue and Depth -Camera*, RGB-D Camera) on suosituin sensoritekniikka SLAM-menetelmissä sisätilaympäristöissä [12]. Syvyyskamera mahdollistaa värikuvan sekä syvyyskuvan samanaikaisen tuottamisen, minkä avulla ympäristön rakenteita voidaan mallintaa [12]. Syvyyskameran muodostaa infrapunalähetin, -vastaanotin sekä yksi värikamera, mitkä mahdollistavat syvyysarvioinnin ympäristöstä [7]. Syvyysarviointi tuotetaan muodostamalla infrapunasta lähetetyillä ja vastaanotetuilla strukturoiduilla valokuvioilla olevan RGB-kuvan, jossa jokaiselle kuvapisteelle voidaan määrittää tarkka etäisyysarvo [7].

Syvyyskameralla voidaan siten luoda tiheä jäljitelmä ympäristöstä ja sen tekstuurista ilman erillistä kartan alustamista. Tämän myötä syvyyskamerat soveltuvat erinomaisesti SLAM-algoritmeihin sisätiloissa [12].

**Stereo-SLAM** on puolestaan menetelmä, joka hyödyntää vähintään kahta kameraa etäisyystietojen arvioimisessa vertailemalla samaa kohdetta kahdesta eri kulmasta otettujen kuvien avulla [7], [11]. Stereo-SLAM pystyy määrittämään syvyyden, eli etäisyysarviot parallaksiperiaatteen avulla, ilman tarvetta lisäantureille, kuten infrapunasensoreille [11]. Stereo-SLAM on siis mainituista SLAM-menetelmistä ainut, joka hyödyntää useampaa RGB-kameraa ympäristön havainnointiin.

Eri ympäristöissä, erityisesti ulkona esiintyy useita haasteita muille SLAM-menetelmille, joita Stereo-SLAM kykenee ratkaisemaan. Syvyyskameraan perustuvalla SLAM:ille ongelmia tuottaa syvyyskameroiden näkökenttä ja mittausetäisyyksien rajoitteellisuudet, mitkä johtuvat vain yhden kameran käytöstä [7], [11]. Syvyyskameran toiminta perustuu infrapunasäteilyyn, minkä vuoksi myös voimakas valo, kuten auringonvalo, voi häiritä syvyyskameran toimintaa. Monokulaarinen SLAM-menetelmä ei kykene määrittämään syvyystietoja lainkaan [10], [11]. Valotutkamene- telmissä ongelmia tuottavat muun muassa asennon suunnan arviointi sekä hinta [7]. Stereo-SLAM:illa saamme siis tasapainoisen ja kustannustehokkaan ratkaisun sekä korkea- että matalastrukturisten ympäristöjen havainnointiin ja kartoitukseen UAV-järjestelmissä. Tämän vuoksi Stereo-SLAM on valittu tutkielman päämenetelmäksi.

Luvusta 2.2 eteenpäin stereo- tai visuaalisella-SLAM-menetelmällä viitataan koko SLAM-järjestelmään sisältäen sensorin, käyttöliittymän, palvelinpuolen sekä kartan muodostamisen. Tulevissa luvuissa tullaan mainitsemaan Stereo-SLAM-menetelmän alla toimivia menetelmiä, kuten suorat ja epäsuorat SLAM-menetelmät (engl. *direct and feature based methods*). Näitä menetelmiä kuvataan samalla ni-

mitystyylillä, mutta kuitenkin yhtenäisellä termillä viitataan tällöin vain Stereo-SLAM:in rakenteen osaan, eikä koko SLAM-järjestelmään.

## 2.2 Stereonäköön perustuva SLAM

Stereo-SLAM-algoritmin prosessiketju agentin paikantamiseen ja ympäristön kartoitukseen eroaa hieman yleisestä SLAM-mallista. Ketjun muodostaa piirteiden tunnistus, yhdistäminen, sijainnin arviointi, pistepilven muodostaminen (*engl. Point cloud estimation*) sekä palvelinpuolen optimointitekniikka [3]. Algoritmit, jotka määrittävät kameroista saadun tiedon käsittelyn voidaan jakaa kahteen osaan: suoriin menetelmiin ja ominaisuuksiin pohjautuviin menetelmiin [5], [7], [8]. Taulukko 2.1 kuvaa Stereo-SLAM:issa hyödynnettyjen algoritmien edustajia, joita tullaan käyttämään tutkielmassa tulosten esittelyyn.

Taulukko 2.1: Tutkielmassa käytetyt SLAM-algoritmit ja niiden perusominaisuudet sekä piirteet

Algoritmi	Tyyppi	Silmukansulku	Kameratuki	IMU-tuki	Havaintoyksikkö
ORB-SLAM2 [10], [13]	Epäsuora	Kyllä	Stereo/RGB-D	Ei	AP
PL-SLAM [1], [14]	Epäsuora	Kyllä	Stereo	Ei	AP + LP
L-SLAM [14]	Epäsuora	-	Stereo	Ei	LP
LSD-SLAM [5], [10]	Suora	Ei	Mono/Stereo	Ei	PI
VINS-FUSION [15]	Epäsuora	Kyllä	Mono/Stereo	Kyllä	AP + PL
S-PTAM [16]	Epäsuora	Kyllä	Stereo	Ei	AP
SVO [17]	Puolisuora	Ei	Mono/Stereo/RGB-D	Ei	AP + PI

AP = avainpiste, LP = linjapiirre, PI = pikseli-intensiteetti ja viiva = puuttuva tieto

Oikean menetelmän valitseminen perustuu vahvasti käyttötarkoitukseen ja ympäristöön, missä menetelmää hyödynnetään [8]. Algoritmit määrittävät piirteiden tunnistamisen, tiedon käsittelyn, kartan optimoimistekniikan sekä sen onko silmukan sulku havainnointia. Eri algoritmien hyödyntämisellä voidaan parantaa droonin suorituskykyä eri toimintaympäristöissä, minkä vuoksi on hyödyllistä tarkastella muutamaa yleisintä algoritmia.

### 2.2.1 Ominaisuuksiin pohjautuvat algoritmit

Epäsuorien eli ominaisuuksiin pohjautuvien algoritmien toiminta perustuu avainpisteisiin, joissa otetusta kuvasta haetaan tiettyjä piirteitä sekä avainkohtia, millä arvioidaan ympäristön kartoitus sekä sijainti [5], [8]. Perinteisille epäsuorille algoritmeille haasteita tuo matalatekstuurisat ympäristöt niiden avainpisteiden puutteellisuuden myötä [14]. Tämän myötä on kehitelty algoritmejä, jotka fuusioivat avainpisteiden ja linjapiirteiden tunnistuksen yhteen, kuten taulukossa 2.1 esitetty PL-SLAM ja VINS-Fusion algoritmit. Linjapiirteiden käyttämisellä mahdollistetaan enemmän hyödyllisten piirteiden tunnistaminen geometrisistä rakenteista ja vähennetään valaistuksen vaikutusta [15]. Linjapiirteiden fuusioimisella mahdollistetaan robustimpi vaihtoehto matalatekstuuriin ympäristöihin [14].

#### ORB-mallit

Suuntautunut FAST ja rotaation mukautettu BRIEF (engl. *Oriented FAST and Rotated BRIEF*, ORB) -mallit, joihin lukeutuu ORB-SLAM, ORB-SLAM2, sekä ORB-SLAM3 ovat suosituimpia visuaalisia SLAM-algoritmejä sensoridatan käsittelemiseen sekä yhdistämiseen [7]. ORB yhdistää kaksi avainpisteiden piirteiden muodostamisen ja yhdistämisen tekniikkaa: suuntautuneen FAST-menetelmän sekä rotaatioon mukautetun BRIEF-menetelmän [7]. Tekniikoiden peruseriaate on löytää yhteisiä piirteitä vertailemalla piirteiden vastaavuuksien kuvaajia. Kuvaajien suuntauksen ja syvyyden perusteella voidaan arvioida kameran liikerata ja sijainti. Tämän jälkeen ORB-menetelmät mahdollistavat myös silmukan sulkemisen havainnoinnin sekä optimoinnin paikallisella niputuksen hienosäädöllä (engl. *Local Bundle-Adjustment*, LBA) [8]. Optimoinnin edistämiseksi LBA:n lisäksi algoritmi suorittaa silmukan sulkuvaiheessa asento-graafi optimoinnin kertyneen ajautuman korjaamiseksi [10]. Tämän lisäksi suoritetaan vielä täysi BA optimaalisimman rakenteen laskemiseksi.

ORB-SLAM2-menetelmällä mahdollistettiin stereokameran hyödyntäminen [7], mikä ei edeltävässä mallissa ollut mahdollista. Uusin ja modernein menetelmä on kuitenkin ORB-SLAM3. Menetelmä integroi visuaalis-inertiaalisen menetelmän (engl. *visual-inertial*, IMU) ORB-SLAM2-menetelmään [1], [8]. Menetelmä mahdollistaa siten paremman suorituskyvyn SLAM-järjestelmään kompensoimalla näköä lisätiedolla, kuten kiihtyvyydellä ja suunnalla [1]. Visuaali-inertiaaliset algoritmit vaativat kuitenkin lisätietolähteitä toimiakseen, minkä vuoksi ORB-SLAM2 luokitellaan yhä johtaviin SLAM-algoritmeihin [8].

### PL-SLAM

Piste-linja-SLAM (engl. *Point-Line-SLAM*, PL-SLAM) on epäsuora SLAM-algoritmi, jonka Gomez Ojeda R ja muut [14] kehittivät huomioimaan tavallisten avainpisteiden lisäksi myös linjapiirteitä. Algoritmi käyttää optimoinnissaan LBA:ta molempien piirteiden huomioimiseen. PL-SLAM hyödyntää myös silmukan sulkemista ja tutkijoiden mukaan prosessia on edistetty huomioimalla myös linjapiirteet.

### 2.2.2 Suorat ja puolisuorat algoritmit

Suorien SLAM-algoritmien paikannus ja kartoitus perustuu sensoreista saatujen kuvien pikselien intensiteettiin [5], [18]. Ainoastaan avainpisteiden eli kulma- ja reunapisteiden sijaan suorissa algoritmeissa hyödynnetään kokokuvan pikseleitä paikannukseen ja kartoitukseen [7], [8]. Myös suorissa menetelmissä hyödynnetään optimointia. Suorat menetelmät ovat kuitenkin erittäin herkkiä valoisuuden vaihtelulle ja ne on rajattu pieniin kameran liikkeisiin [14]. Suorien algoritmien tarkkuus heikenee siten suurilla näkymän muutoksilla. Suuremman robustisuuden ja sijainnin arvion tarkkuuden myötä suurinosa näköön perustuvista menetelmistä nykypäivänä hyödyntää epäsuoria algoritmeja [18].

## LSD-SLAM

Suuri-mittakaavainen suora-SLAM (engl. *Large-Scale Direct SLAM*, *LSD-SLAM*) on suorapohjainen SLAM-algoritmi, joka hyödyntää pikseleitä paikannukseen avainpisteiden sijasta [10], [19]. LSD-SLAM suunniteltiin alunperin monokulaariselle kameralle, mutta on myöhemmin laajennettu myös stereokameralle [19]. Stereo LSD-SLAM hyödyntää asento-graafi optimointia sijainnin ja asennon määrittämiseen, mutta ei sisällä kykyä havainnoida silmukan sulkua [19].

## SVO

Puolisuora visuaalinen odometria (engl. *Semidirect Visua Odometry*, *SVO*) on nimensä mukaan puolisuora SLAM-algoritmi, mikä C. Forsterin ja muiden [17] mukaan implementoi suorien ja epäsuorien algoritmien ominaisuudet. SVO havaitsee avainpisteitä vain valituissa avainkehyksissä rinnakkaisessa säikeessä, jolloin prosessi on kytketty irti reaaliaikaprozessista. Tämän jälkeen suoralla seuranta algoritmilla poistetaan tarve robustiselle datan keräämiselle. Tutkijoiden mukaan vaiheet mahdollistavat vain harvan kartoituksen ympäristöstä, mikä tekee prosessista kevyen ja nopean. SVO-algoritmi on visuaalinen odometria, joka suorittaa BA-optimoinnin tarkkuuden edistämiseksi, mutta ei kykene havaitsemaan silmukan sulkua [17].

# 3 Stereo-SLAM:ia hyödyntävän dronin tulokset eri toimintaympäristöissä

Luvussa jaetaan Stereo-SLAM:ia hyödyntävän dronin suoritusten tulokset korkeastrukturiseen sekä matalastrukturiseen ympäristöön. Luvussa hyödynnetään neljää eri tietoaaineistoa eri tiedeartikkeleista KITTI- [20], EuRoC- [21], Rosario- [22] sekä Blackbird-tietoaaineistoja [23]. Osa tietoaaineistoista on kuvattu maassa operoivalla aluksella dronin sijaan. Kyseiset tietoaaineistot tarjoavat kuitenkin arvokasta tietoa Stereo-SLAM:in luotettavuudesta eri ympäristöissä, jota voidaan soveltaa matalalla lentävään drooniin. Taulukko 3.1 kuvaa luvussa hyödynnetyt tietoaaineistot, niiden operoinnin alustat, tekstuurit, käytetyn stereokameran tarkkuuden, dynaamisuuden sekä silmukan sulun mahdollisuuden.

Taulukko 3.1: Perusominaisuudet valituista SLAM-tietoaaineistoista

Tietoaaineisto	alusta	Rakenne (korkea/matala)	Silmukan sulkeminen	Kamera (stereo)	Dynamiikka
KITTI [20]	Auto	Molemmat	Mahdollinen	10 Hz	Maltillinen
EuRoC [21]	UAV	Molemmat	Mahdollinen	20Hz	Vaihteleva
Blackbird [23]	UAV	Molemmat	Ei	60Hz	Agressiivinen
Rosario[22]	Robotti	Matala	Ei	15Hz	Maltillinen

### 3.1 Korkeastruktuurinen ympäristö

Korkeastruktuurisella ympäristöllä tarkoitetaan tässä tutkielmassa rakennettua tekstuuririkasta ympäristöä, kuten urbaania ympäristöä, joka on avainpisteiden kannalta rikas. Luvussa hyödynnetään ulkoilma- sekä sisätilaympäristöjä kokonaiskuvan saamiseksi.

Droonin paikkannustarkkuus on riippuvainen hyödynnetystä SLAM-algoritmista, parametreista sekä ympäristöasetuksista [8]. Näiden myötä luvussa hyödynnetään useita toisistaan poikkeavia tietoaaineistoja kokonaiskuvan muodostamiseksi. A. Geigerin ja muiden julkaisema KITTI-tietoaaineisto [20] on yksi näistä aineistoista, josta on samalla muodostunut nykypäivän standardi visuaalisten SLAM-menetelmien vertailemiseen [14]. KITTI-tietoaaineisto tarjoaa tuloksia useasta eri ympäristöstä [3], [14], [20]. Näistä ympäristöistä taulukkoon 3.2 on valittu KITTI-tietoaaineiston kymmenen urbaanista sekvenssiä. Taulukon 3.2 tulokset eivät kuitenkaan ole suoraan KITTI-tietoaaineistosta, vaan Gomez-Ojedan ja muiden [14] esittämistä tuloksista KITTI-sekvensseissä. KITTI-sekvenssit on kuvattu 10 hertsin ja 1240 x 376 resoluution stereokameralla [10].

Taulukko 3.2: Stereo-SLAM-algoritmien RMSE-tuloksia 10:ssä valitussa KITTI-sekvenssissä [14]

algoritmi		00	02	03	04	05	06	07	08	09	10
L-SLAM	$T_{rel}$	3.32	6.62	6.83	-	3.06	5.99	3.58	6.15	6.92	6.86
	$R_{rel}$	6.67	9.42	7.14	-	2.25	8.29	6.52	9.23	4.66	5.55
PL-SLAM	$T_{rel}$	2.36	2.35	3.74	2.21	1.74	3.51	1.83	2.18	1.68	1.21
	$R_{rel}$	0.89	0.91	1.54	0.30	0.88	2.72	1.03	1.15	0.92	0.99
ORB-SLAM2	$T_{rel}$	1.82	1.81	2.79	1.07	1.03	1.25	0.94	1.78	1.11	1.09
	$R_{rel}$	0.58	0.46	0.44	0.18	0.37	0.56	0.46	0.57	0.42	0.46

Taulukossa 3.2 hyödynnetään kahta eri mittaria tulosten esittämiseen. Keskimääräinen suhteellinen siirtymävirhe (engl. *translation error*,  $T_{rel}$ ) ja rotaatiovirhe (engl.

*Rotation error,  $R_{rel}$* ) [10], [14]. Siirtymävirheet esitetään prosentteissa, kun taas rotaatiovirheet esitetään muodossa aste jaettuna 100:lla metrillä. Sekvenssi 02 voidaan esittää seuraavasti: PL-SLAM-algoritmin arvioitu siirtymä poikkeaa 2.35 prosenttia ja arvioitu rotaatio poikkeaa todellisesta 0.91 astetta sadalla metrillä [14]. Viivalta taas merkitään testiä, joka on epäonnistunut. Tulosten mukaan drooni suoriutuu hyvin testiympäristöissä ORB-SLAM2- sekä PL-SLAM-algoritmeilla. Tämä on kuitenkin ennaltaoletus Gomez Ojedan ja muiden [14] mukaan, sillä algoritmit ovat ominaisuuksiin pohjautuvia menetelmiä. Taulukossa 3.3 esitetään toinen KITTI-tietoaineistoon perustuva tarkkuuden vertailu, mikä poikkeaa hieman taulukosta 3.2.

Taulukko 3.3: Stereo-SLAM-algoritmien RMSE-tuloksia 10:stä valitusta KITTI-sekvenssistä eri parametrimuunnelmilla [10].

algoritmi		00	02	03	04	05	06	07	08	09	10
ORB-SLAM2	$T_{rel}$	0.70	0.76	0.71	0.48	0.40	0.51	0.50	1.05	0.87	0.60
	$R_{rel}$	0.25	0.23	0.18	0.13	0.16	0.15	0.28	0.32	0.27	0.27
LSD-SLAM	$T_{rel}$	0.63	0.79	1.01	0.38	0.64	0.71	0.56	1.11	1.14	0.72
	$R_{rel}$	0.26	0.23	0.28	0.31	0.18	0.18	0.29	0.31	0.25	0.33

Vertailuun on valittu taulukon 3.3 kaikki urbaanisessa ympäristössä kuvatut sekvenssit. Taulukot 3.2 ja 3.3 perustuvat samoihin KITTI-sekvensseihin, mutta tulokset on saatu eri lähteistä. Taulukko 3.2 perustuu Gomez Ojedan ja muiden [14] esittämiin testituloksiin, joissa käytetyt parametrit poikkeavat alkuperäisestä toteutuksesta [10], johon puolestaan taulukko 3.3 perustuu. Tämä aiheuttaa tuloksissa eroavaisuuden taulukoiden välillä, vaikka testiympäristö ja algoritmi olisikin sama [14]. Parametrien eroavaisuuksien myötä algoritmien tulokset eivät ole suoraan vertailukelpoisia. Tulokset tarjoavat kuitenkin arvokkaan kuvan algoritmin toiminnasta KITTI-sekvensseissä sekä korostavat parametrivalintojen vaikutusta tuloksiin.

Taulukossa 3.4 hyödynnetään Blackbird-tietoaineistoa [23]. Blackbird tarjoaa nopealiikkuvasta droonista otettuja tietoja, jotka on koottu viidestä eri ympäristöstä. Tässä tutkielmassa hyödynnetään metroasemalla kuvattua tietoaineistoa kolmella eri vaikeustasolla. Blackbird-aineisto on kuvattu eteenpäin ja alaspäin suuntautuneella 120 hertsisellä stereoparikameralla sekä droonilla, jonka nopeus on 0.5-7 m/s [23]. Taulukossa 3.4 esitetään Stereo-SLAM-algoritmien siirtymävirheet sekä rotaatiovirheet metreissä.

Taulukko 3.4: Stereo-SLAM-algoritmien RMSE-keskiarvot sekä onnistumisprosentti Blackbird-testiympäristössä [23]

lento	Taso	Nopeus (m/s)	YAW	VINS		ORB-SLAM2		Onnistumis % VINS   ORB2
				$T_{rel}$	$R_{rel}$	$T_{rel}$	$R_{rel}$	
sid	H	1-7	vakio	$5.27 \pm 0.93$		$1.6 \pm 1.2$		85.7   28.6
mouse	M	1-7	vakio	$4.89 \pm 0.89$		$0.55 \pm 0.4$		100   28.6
mouse	M	1-7	eteenpäin	$13.5 \pm 3.38$		$1.7 \pm 2.1$		57   14.3
picasso	V	1-6	vakio	$5.08 \pm 0.82$		$1.67 \pm 0.67$		83.3   50
picasso	V	1,3-5	eteenpäin	$5.65 \pm 0.55$		$0.4 \pm 0.1$		50   25

Blackbird-tietoaineistossa hyödynnetään kahta droonin suuntakulmaa: eteenpäin suuntautunutta kulmaa sekä vakiota suuntakulmaa (engl. YAW) [23]. Vakio suuntakulmassa drooni pitää suunnilleen vakiosuunnan liikkueessaan, kun taas eteenpäin suuntautuneessa drooni osoittaa suuntaan, johon se siirtyy eli suuntakulma muuttuu liikkeen mukana. Vaikeustasolla tarkoitetaan suorituksen aikana seurattujen piirteiden keskimääräistä lukumäärää [23]. Tasot vaihtelevat yksinkertaisesta reitistä vakiosuuntakulmalla monimutkaisiin ratoihin, joissa nopeus, korkeus ja suuntakulma muuttuvat. Taulukossa 3.4 esitetään jokaisen vaikeustason lentojen virhetarkkuuksien keskiarvo onnistuneista suorituksista sekä onnistuneiden suoritusten keskiarvo. Taulukko voidaan siten esittää seuraavanlaisesti: keskitason "mouse"lentoradalla va-

kiosuuntakulmalla VINS -algoritmin keskiarvollinen siirtymävirhe oli 4.89 % ja 0.89 astetta jokaista kuljettua metriä kohti. Amado Antoninin ja muiden [23] mukaan on kuitenkin tiedostettava, että tietoaaineistossa molempiin algoritmeihin on hyödynnetty oletusparametrejä EuRoC-tietoaaineistosta asianmukaisilla muutoksilla. Parametrien hienosäätäminen voisi johtaa parempiin seurannan tuloksiin [23].

Viimeisimpänä testiympäristönä hyödynnetään EuRoC-tietoaaineistoa [21]. EuRoC sisältää kaksi sisätilaympäristöä sekä teollisuustehdas ympäristön [1], [10], [14]. Vicon-ympäristöt on jaettu kahteen pääasetelmaan, V1 ja V2 ympäristöihin [21]. Molemmat on toteutettu sisätiloissa, mutta esineiden ja huonekalujen sijoittelu eroaa toisistaan. Kumpaankin asetelmaan sisältyy kolme sekvenssiä, V101-V103- sekä V201-V203 -sekvenssit, jotka poikkeavat toisistaan valaistuksen ja liikkeen nopeuden suhteen. Näistä 01-sekvenssit ovat valoisimpia ja nopeudeltaan hitaimpia toteutuksia, kun taas 03-sekvenssit on toteutettu suuremmalla liikkeellä pimeämmässä ympäristössä [21]. Taulukossa 3.5 hyödynnetään molempia Vicon-huoneessa toteutettuja ympäristöasetelmia.

Taulukko 3.5: Stereo-SLAM-algoritmien siirtymävirheet valituissa EuRoC-tietoaaineiston sekvensseissä

SLAM	V101	V102	V103	V201	V202	V203
ORB-SLAM2 [10]	0.0350	0.0200	0.0480	0.0370	0.0350	-
LSD-SLAM [10]	0.0660	0.0740	0.0890	-	-	-
PL-SLAM [14]	0.0423	0.0459	0.0689	0.0609	0.0565	0.1261
SVO[1]	0.040	0.0400	0.0700	0.0500	0.0900	0.7900
VINS-fusion[1]	0.5500	0.2300	-	0.2300	0.2000	-
SP-SLAM [1]	0.0910	0.0680	0.1710	0.0630	0.0830	-

Taulukossa viivalla tarkoitetaan suorituksen epäonnistumista.

Algoritmien tulokset esitetään kuljetun reitin poikkeamasta metreissä. Taulukon 3.5 tulokset on kerätty kolmesta eri artikkelista [1], [10], [14], joissa hyödynnetään eri parametrejä käytetyssä UAV:ssa. Tämän myötä algoritmien tulokset eivät ole suoraan verrattavissa toisiinsa, vaan suuntaa antavia.

## 3.2 Matalastrukturinen ympäristö

Matalastrukturisella ympäristöllä viitataan tutkielmassa visuaalisesti vähäpiirteiseen matalatekstuuriseen ympäristöön, kuten peltoon, tundraan tai matalatekstuuriseen huoneeseen. Eli ympäristöihin, jotka ovat avainpisteiden kannalta puutteellisia, mikä mahdollistaa vastakkaisen ympäristöasetelman 3.1 luvussa käsiteltyyn korkeastrukturiseen ympäristöön. Myös tässä alaluvussa hyödynnetään eri tietoa-ineistoja, kuten KITTI:ä [20], Rosariota [22] sekä EuRoC:ia [21] tuloksien esittämiin.

Taulukko 3.6: Eri Stereo-SLAM-algoritmien tulokset matalastrukturisessa KITTI 01 -sekvenssissä

SLAM	$T_{rel}$	$R_{rel}$
L-SLAM [14]	-	-
LSD-SLAM [10]	2.36	0.36
PL-SLAM [14]	5.80	2.32
ORB-SLAM2 [10]	1.39	0.21

Taulukko 3.7: SLAM-algoritmien siirtymävirheet Gomez Ojedan ja muiden esittämässä testi-ympäristössä

Sekv.	L-SLAM	PL-SLAM	ORB-SLAM2
Helppo	0.1412	0.1243	0.1391
Keskitaso	0.1998	0.1641	-
Vaikea	0.1801	0.1793	-
Rot-vaikea	0.2247	0.2034	0.2910

Taulukossa 3.6 testituloksien siirtymävirheet jälleen prosenteissa sekä rotaatiovirheet asteissa. Taulukossa 3.6 yhdistetään kaksi artikkelia [10], [14], jotka käyttävät KITTI-tietoaainestoa testituloksiinsa. Artikkeleissa on hyödynnetty eri parametrejä KITTI-asetuksissa. Tämän takia algoritmien tulokset taulukossa 3.6 ei ole tarkoitettu vertailtavaksi keskenään, vaan tuoda yleisen kuvan Stereo-SLAM:in luotettavuudesta moottoritieympäristössä hyödyntäen eri algoritmejä. Gomez Ojedan ja muiden [14] esittämän testiaineiston lisäksi otetaan tarkasteluun tunnetumpi matalatekstuurinen sisätilyympäristö EuRoC-tietoaainestosta [21].

MH-ympäristöillä tarkoitetaan EuRoC-tietoaainestossa konehallia (engl. *machine hall*, MH) [1], [21]. Ympäristöt MH04 ja MH05 luokitellaan haastaviksi matalatekstuuriseksi sekä heikosti valaistuiksi ympäristöiksi [1]. MH03-ympäristö luokitellaan taas valoisaksi, mutta matalatekstuuriseksi ja nopealiikkeiseksi ympäristöksi [21].

Taulukko 3.8 esittää Stereo-SLAM-algoritmien konehalli-sekvenssien siirtymävirheet. Taulukkoon on mukaanluettu vain matalatekstuurisiet konehalli-sekvenssit.

Taulukko 3.8: Stereo-algoritmien siirtymävirheet EuRoC-tietoaineistossa

SLAM	MH03	MH04	MH05
ORB-SLAM2 [10]	<b>0.0280</b>	<b>0.1190</b>	<b>0.0600</b>
LSD-SLAM [10]	X	X	X
PL-SLAM [14]	0.0399	0.0641	0.0697
SVO[1]	0.0270	0.1700	0.1200
VINS-fusion[1]	0.3300	0.7800	0.5000

Viimeisenä tarkastellaan Rosario-tietoaineistoa [22], joka yhdistää matalatekstuurisien sekä matalastruktuurisen ulkoilmaympäristön. Sekvenssit on kuvattu maassa pellolla operoivalla robotilla 672 x 376 pikselin resoluutiolla ja 15 Hz:n stereokameralla [22]. Rosario-tietoaineisto ei hyödynnä lennokkia, mutta tarjoaa kuitenkin erinomaisen vastakkainasettelun rakennettuun ympäristöön. Taulukossa 3.9 esitetään kahden Stereo-SLAM-algoritmin kartoituksen tulokset kyseisessä ympäristössä.

Taulukko 3.9: Stereo-SLAM:in siirtymävirheet Rosario-tietoaineistossa [22].

Algoritmi		R01	R02	R03	R04	R05	R06	KI03	KI05	KI06
ORB-SLAM2	m	1.41	2.24	3.50	2.21	2.23	5.19	0.60	0.80	0.80
	%	0.23	0.70	2.06	1.45	0.68	0.73	0.11	0.04	0.06
S-PTAM	m	3.85	1.80	2.37	1.49	-	-	1.66	2.85	2.99
	%	0.63	0.56	1.40	0.98	-	-	0.30	0.13	0.24

R-sekvenssi = Rosario-sekvenssi, KI = KITTI-sekvenssi

Taulukon 3.9 siirtymävirheet on merkitty metreissä ja siirtymävirheen suhde lentoradan pituuteen prosenteissa. Tulokset on kerätty Rosario-tietoaineistosta [22]. Tuloksissa on hyödyttety kahta eri stereo-algoritmiä ORB-SLAM2 sekä S-PTAM. Tulokset esitetään muodossa: ORB-SLAM2:n siirtymävirhe sekvenssissä 01 oli 0.23 % kuljetusta liikeradasta eli virhematka oli 1.41 metriä.

# 4 Autonomisen droonin luotettavuus, haasteet ja ratkaisut

## 4.1 Tulosten analyysi

Edeltävässä luvussa tuotiin esille Stereo-SLAM:in tuloksia useassa eri ympäristössä, jossa jokaisessa oli samoja ja eriäviä Stereo-algoritmejä. Tulosten keskiarvot sekä jokaisen ympäristön yhteenlaskettu keskiarvo algoritmille on koottu yhteen taulukossa 4.1.

Taulukko 4.1: Stereo-algoritmien siirtymävirheiden lasketut keskiarvot hyödyntäen seitsemän eri testiympäristön tuloksia

Tietoaineisto strukturi yksikkö	KITTI [20] (korkea) RMSE, %	KITTI [20] (matala) RMSE, %	Blackbird [23] (korkea) RMSE, m	EuRoC [21] (korkea) RMSE, m	EuRoC [21] (matala) RMSE, m	Rosario [22] (matala) RMSE, m	PL [14] (matala) RMSE, m
ORB-SLAM2	0.658* [10]	1.39* [10]	5.92 [23]	0.035 [10]	0.069* [10]	2.797* [22]	0.215 [14]
LSD-SLAM	0.769* [10]	2.36* [10]	X	0.076 [10]	X	X	X
PL-SLAM	2.281* [14]	5.80* [14]	X	0.067* [14]	0.058* [14]	X	0.168* [14]
SVO	X	X	X	0.18* [1]	0.106* [1]	X	X
VINS	X	X	6.878 [23]	0.303 [1]	0.534* [1]	X	X
S-PTAM	0.97* [22]	X	X	X	X	2.378 [22]	X
L-SLAM	5.481 [14]	- [14]	X	0.0719 [14]	-	X	0.186* [14]

Tähdellä \* merkitään algoritmia, joka suoritti onnistuneesti kaikki tietoaineiston sekvenssit, X:llä merkitään puuttuvaa tulosta ja viivalla merkitään epäonnistunutta tulosta.

Taulukossa 4.1 esitetyt tulokset on saatu eri artikkeleista, jotka ovat hyödyntäneet seuraavia tietoaineistoja tuloksien esittämiseen: [20], [21], [22], [23]. Algoritmit ja niiden tulokset kyseisissä tietoaineistoissa on yhdistetty tutkielman kolmannen luvun eri taulukoista. Tulokset on ilmaistu laskemalla aineistojen sekvenssien RM-

SE (m) ja RMSE (%) keskiarvot riippuen kumpaa muotoa tietoinestossa käytetään. Taulukkoon 4.1 yhteenkerätyt testiympäristöt vaihtelevat tekstuuruudessa, matkassa, nopeudessa, ajassa, käännösten määrässä ja valoisuudessa.

Algoritmin 100 % onnistumisprosentilla tarkoitetaan, että drooni/agentti kykenee jäljittämään reitin alusta loppuun kyseisellä algoritmilla [1], [23]. Voidaan huomata tutkielmissa esitetyistä taulukoista, että oikein valitulla SLAM-algoritmilla saadaan kaikissa testiympäristöissä 100 % onnistumisprosentti poislukien Amado Antonini ja muiden julkaisema testiympäristö Blackbird [23].

Haastavaksi ympäristöksi voidaan luokitella heikosti valaistut, dynaamiset sekä nopeat ympäristöt [18], jonka perusteella Antonini ja muiden luoma testiympäristö on luotu. Blackbirdin [23] testituloksissa ei hyödynnetä silmukan sulkemista ja nopeudet yltyvät jopa 7 m/s. Lisäksi tuloksien saamiseen käytetään ainoastaan vakio-parametrejä ilman hienosäätöä. Näistä huolimatta oikein valitulla kamera-asennolla sekä algoritmilla saavutetaan 83.3 % - 100 % onnistumisia, kuten taulukosta 3.4 voidaan havaita.

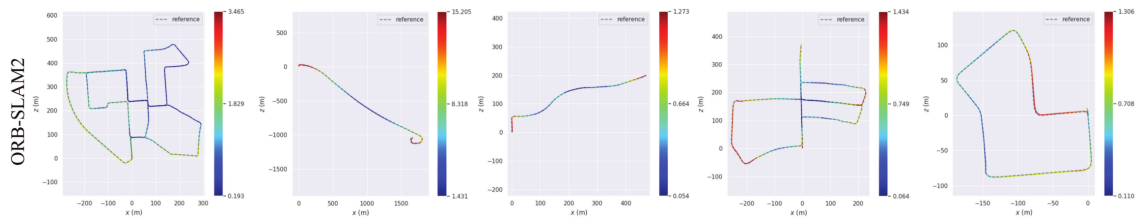
Tuloksista havaitaan myös selkeä ero kameran suuntauksen muuttuessa eteenpäin vakiosuuntauksen sijaan. Samalla algoritmilla ja samassa picasso-sekvenssissä onnistumisprosentti laskee 100 prosentista 57 prosenttiin, kun kyseessä on eteenpäin suunnattu kamera. Tuloksen selkeää pudotusta voi selittää droonin altistuminen suuremmalle rotaatiolle eteenpäin suunnatun kameran myötä. Kasvanut rotaatio aiheuttaa merkittävää perspektin muutosta [24], mikä vaikeuttaa avainpisteiden luotettavaa seuranta [18], [24]. Tästä syystä rotaatioherkkyys on tunnettu visuaalisten SLAM-menetelmien rajoite [15].

Yhteenveto-taulukosta 4.1 ilmenee, että ORB-SLAM2 suoriutuisi keskiarvollisesti pienemmällä virhearviolla (5.92 m) verrattuna VINS-algoritmin virhearvioon (6.878 m). Tulos on kuitenkin hieman harhaanjohtava, sillä ORB-SLAM2-algoritmin onnistumisprosentti vaihtelee sekvenssistä riippuen 14.3 % - 50 % välillä verrattaen

83.3 % - 100 % onnistumisprosenttiin VINS-algoritilla. Dynaamisissa ympäristöissä riittävien avainpisteiden poiminen yleisesti tuottaa haasteita visuaalisille menetelmille [8], [18]. VINS-algoritmi yhdistää täten myös linjapiirteiden tunnistamisen, joka J.Lee:n ja muiden [15] mukaan tehostaa paikantamisen tarkkuutta. Tämä saattaa osin selittää tulosten suurten eron algoritmien välillä. Tulokset ovat kuitenkin huomattavasti heikommät Blackbird-testiympäristössä kuin muissa taulukossa 4.1 esitetyissä ympäristöissä. Erittäin dynaamisissa lennoissa nykyiset algoritmit eivät pysty suorittamaan sensoreista saatua tietoa reaaliajassa [23], mikä selittää huomattavasti heikompaa tulosta. Dynaamisiin ympäristöihin yksi edistävää tekijä olisi hyödyntää havainnointimenetelmää, joka havaitsee ja poistaa dynaamisia kohteita vähentäen arviointivirheitä dynaamisissa ympäristöissä [7], [25].

KITTI-tietoaaineston tarjoamissa korkeastrukturisissa ympäristöissä tutkielmassa käytetyillä algoritmeilla saadaan 100 % onnistumisprosentti erittäin pienillä virhearvioilla. Lukuun ottamatta L-SLAM-algoritmia, joka vain linjapiirteiden havaintokyvyn myötä on selkeästi heikompi testatuissa ympäristöissä [14]. Parhaimmillaan saadaan jopa 0.4 % virhearvioita ORB-SLAM2 algoritmilla sekvenssissä 05, kuten taulukossa 3.3 esitetään. KITTI-sekvenssi 05 on etäisyydeltään noin 2200 metriä [26], joka vastaa 0.4 % siirtymävirheellä vain 8.8 metrin virhematkaa. Siirtymävirheellä ei tarkoiteta etäisyyttä kohteesta, vaan matkalla yhteenlasketuista pienistä heitoista, kuten kuvassa 4.1 visualisoidaan. Kuvassa 4.1 esitetään Stereo-SLAM:in arviot virheet todellisesta reitistä hyödyntäen ORB-SLAM2-algoritmia. Kuvissa siirtymävirhe merkitään metreissä [24].

Kuvissa sinisellä tarkoitetaan pientä virhettä, ja mikäli väri lähenee punaista virhemarginaali kasvaa. Matalatekstuurisissa ympäristöissä tuottavat selkeitä haasteita varsinkin avainpiste pohjaisille algoritmeille [24]. Tätä puoltaa myös yhteenvetotaulukosta 4.1 ja kuvasta 4.1 ilmenevä data, jossa matalatekstuurisissa ja -struktuurisissa reitit aiheuttavat enemmän vaikeuksia kuin korkeastrukturisissa, tekstuuri-rikkaat ja

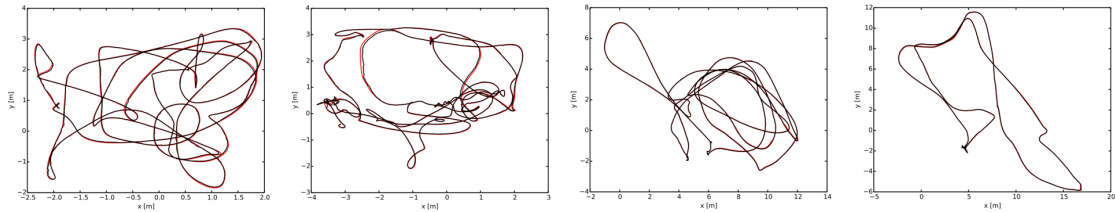


Kuva 4.1: Visualisoidut siirtymävirheet viidessä eri KITTI-sekvenssissä ©2025 [24].

silmukan sulkemismahdollisuuksia sisäistävät reitit. Korkeastrukturisiksi ympäristöiksi luokitelluissa KITTI-sekvensseissä Stereo-SLAM suoriutuu parhaiten valitulla algoritmilla keskiarvolla 0.66 % virhearviolla verrattaen 1.39 % virhesiirtymään matalastrukturisessa ympäristössä. Virhearvion eroon vaikuttaa sekvenssissä 01 avainpisteiden puuttellisuus, nopeus, matala kuvanlaatu sekä matalatekstuurisuus [10]. KITTI-sekvenssit 00, 02-10 perustuvat urbaaniseen ympäristöön korkeatekstuurisilla kuvilla, minkä myötä avainpistepohjaiset algoritmit kuten ORB-SLAM2 pärjäävät hyvin tuloksissa [14], [18].

EuRoC-testiympäristöt tuovat esille valon ja liikkeen sumennuksen vaikutuksen Stereo-SLAM-algoritmeille [21]. Oikein valitulla algoritmilla Stereo-SLAM:in onnistumisprosentti on jälleen 100 %. Avainpisteisiin pohjautuva ORB-SLAM2 tuottaa keskiarvallisesti parhaimmat tulokset, mutta ei kykene suorittamaan sekvenssiä V203, kuten taulukossa 3.5 esitetään. Tämä johtune sekvenssin suuresta perspektiivisestä liikkeestä sekä niukasti teksturoidusta ympäristöstä [24]. Avainpisteisiin pohjautuvat algoritmit, kuten ORB-SLAM2, on siten taipuvaisia menettämään seurannan [14]. Kuitenkin taulukossa 3.5 linjapiirteitä hyödyntävät algoritmit, kuten PL-SLAM, osoittautuu hyväksi vaihtoehdoksi 0.1261 metrin virhearviolla 86.1 metrin pituisessa sekvenssissä. Linjapiirteiden hyödyntämisellä pienennetään liikkeen sumennuksen vaikutusta [14]. Linjapiirteillä täydennetään siten ainoastaan avainpisteisiin pohjautuvia algoritmeja [14], [18]. Seuraavalla kuvalla 4.2 tuodaan visua-

lisesti esille Stereo-SLAM:in arvioiman reitin täsmällisyyttä EuRoC-sekvensseissä verrattuna todelliseen reittiin.



Kuva 4.2: Droonin arvioima lentoreitti (mustalla) verrattuna todelliseen reittiin (punaisella) V101, V202, MH03 ja MH05 EuRoC-ympäristöissä ©2011 IEEE.

EuRoC-testiaineistossa kuvatut matalastrukturiset sekä -tekstuurisets ympäristöt tuottivat jossain määrin enemmän haasteita kuin korkeastrukturisissa Vicon-sekvensseissä. MH03, MH04, ja MH05 luokitellaan kaikki nopealiikkeisiksi sekä MH04 ja MH05 sekvenssit luokitellaan lisäksi pimeiksi ympäristöiksi [21]. Matalatekstuurisets ja heikosti valaistut ympäristöt aiheuttavat avainpisteiden puutteellisuutta, mikä heikentää paikantamisen tarkkuutta visuaalisille menetelmille [1], [18]. Tämän myötä yhteenveto taulukossa 4.1 esitetty ORB-SLAM2:n tulokset heikenevät 0.035 virhesiirtymästä 0.069 metrin virhesiirtymään. Koska ympäristö on kuvattu sisätiloissa, linjapiirteitä hyödyntävillä algoritmeilla, vaikutus on vähäinen tai jopa positiivinen, jolloin suorituskyky saattaa parantua kuten SVO- ja PL-SLAM-algoritmeilla.

Tutkielmassa hyödynnettiin matalastrukturisten ympäristöjen tulosten esittelyyn myös Taihu Piren ja muiden julkaisemaa ROSARIO-tietoaineistoa [22]. Tuloksista viitaten taulukkoon 3.9 ilmenee jälleen Stereo-SLAM:in heikompi suorituskyky matalastrukturisissa ympäristöissä. ORB-SLAM2 algoritmillä saadaan 100 % onnistumisprosentti, mutta virhearviot heittelevät 0.23 % - 2.06 % välillä verrattuna myös maassa ajettuun KITTI-ympäristöön, jossa virhearviot vaihtelevat 0.04-0.11 % välillä [22]. Testiympäristöjen tulosten eroavaisuus pohjautuu eri haasteisiin, jota Rosario-testiympäristössä on mukaanluettu [22]. Näitä ympäristöjä ovat toistu-

va tekstuuri, kasvien joustava liike sekä valoisuuden muutokset, mitkä aiheuttavat kamerapohjaisen SLAM:in menettämään seurannan aiheuttaen virheen todennäköisyyttä Taihu Piren ja muiden mukaan.

## 4.2 Parannusmahdollisuudet

Kuten tutkielmassa esitetyissä tuloksista voidaan todeta kamerapohjaisen SLAM:in toiminta heikkenee tietyissä testiympäristöissä. Näitä ovat heikentyneen valoisuuden ympäristöt, tilanteet, jossa kuvat sumentuvat nopean liikkeen myötä, sekä matalatekstuurisat ympäristöt [1]. Robustisuuden parantamiseksi tutkijat ovat kehittäneet IMU:n edistämään SLAM:in tehokkuutta [1], [25]. Dajiang Lei ja muut [1] esittivät ORB-SLAM3 tuloksien eroja EuRoC-tietoaaineistoissa, kun algoritmi hyödyntää IMU:a ja kun ei. Pimeämmässä ja dynaamisissa ympäristöissä kuten MH04, MH05 sekä Vicon-huoneissa poislukien sekvenssi V101 on selkeää parannusta. IMU:lla algoritmi suorittaa keskiarvallisesti 0.035 metrin heitolla verrattuna ei IMU:a hyödyntävään ORB-SLAM3 0.084 metrin tulokseen. Ilman IMU:a toimiva algoritmi suoriutuu kuitenkin paremmin yksinkertaisemmissa ja valoisissa ympäristöissä, joihin Michael Burri ja muiden [21] julkaisemat sekvenssit MH01-MH03 ja V101 luetellaan. Tämän havainnon esitti myös Amado Antonini ja muut [23] heidän julkaisemassaan artikkelissa, jossa heidän testiaineistossaan haastavissa ympäristöissä parhaiten suoriutuva algoritmi suoriutui usein heikommin yksinkertaisemmissa lentoskenaarioissa. Muita tunnettuja sensorifuusioita on valotutkan yhdistäminen stereokameran kanssa. Valotutkan korkean hinnan ja painon myötä se ei kuitenkaan sovellu pieniin lennokkeihin [1].

Puolisuorien algoritmien edistäminen, kuten Christian Forsterin ja muiden [17] esittämä SVO, tarjoaa myös ratkaisumenetelmän SLAM-ongelmaan. SVO:n prosessi on kevyt ja nopea, mikä tarjoaa korkean nopeuden ja tarkan ajan suorituksen matalavirtaisissa alustoissa, kuten droneissa [24]. Taulukosta 4.1 huomataan-

kin, että SVO on ainoa algoritmi PL-SLAM:in lisäksi, joka suoriutuu matalastrukturisista EuRoC-testiympäristöistä 100 % onnistumisprosentilla. SVO:n heikompi tarkkuus kuitenkin varjostaa algoritmin suorituskykyä verrattuna muihin. Baosheng Zhang ja muut esittävät artikkelissaan [18] täydennetyn SVO-algoritmin nimeltä DynPL-SVO, joka hyödyntää myös linjapiirteitä yhdistäen siten avainpisteet, pikseli-intensiteetin ja linjapiirteet yhteen algoritmiin. Artikkelin mukaan tulokset parantuivat keskimäärin 20 % verrattuna muihin viimeistä tekniikkaa edustaviin SVO-malleihin. Tulokset viittaavat, että puolisuorien algoritmien edistämällä voitaisiin edistää vielä Stereo-SLAM:ia ilman IMU:a.

## 5 Pohdinta

Stereo-algoritmien suora vertailu toisiinsa korkea- sekä matalatekstuurisissa ympäristöissä todentui haastavaksi, koska julkisesti saatavia ajankohtaisia tietoaaineistoja, jotka sisältävät molemmat testiympäristöt ei juuri ollut saatavilla. Tämän myötä tutkielmassa hyödynnettiin myös tietoaaineistoja, joissa aluksena operoi maassa toimiva alus. Tuloksia voidaan pitää vertailukelpoisina, mutta ne eivät tarjoa koko totuutta. Drooni tarjoaa eri näkökulman, vaikka se operoisi lähellä maata. Droonin tarjoama korkeampi näkökulma voisi tarjota laajemmän kuvan, jolloin avainpisteitä olisi enemmän tarjolla. Laajemmalla kuvalla voitaisiin myös pienentää kuvan sumeutta nopeammissa liikkeissä, mikä tehostaisi Stereo-SLAM:in toimintakykyä eri ympäristöissä. On kuitenkin tiedostettava, että korkeampi näkökulma ei tarkoita aina laajeempaa kuvaa. Tämä tulee esiin esimerkiksi sisätiloissa, kuten käytävillä.

Haasteita tuloksien esittämisessä aiheutti myös eri parametrien käyttö, vaikka testiympäristö olisi ollut sama. Tämä ilmiö todentui useamman kerran esimerkiksi taulukkojen 3.2 ja 3.3 välillä. Molemmissa taulukoissa hyödynnettiin samoja KITTI-testiympäristöjä, mutta algoritmeissa käytetyt parametrit erosivat toisistaan. Tämä aiheuttaa sen, että tulokset eivät ole vertailukelpoisia keskenään. Tutkielman näkökulman kannalta kyseinen ilmiö ei ole niin merkittävä, sillä tutkielmassa tarkastellaan yleisellä tasolla Stereo-SLAM:in toimintakykyä, ei eri algoritmien.

Tietoaaineistot ovat kuitenkin selkeästi eritelty vain kameraa hyödyntäviin sekä muita sensoreita hyödyntäviin tietoaaineistoihin. Tutkielmasta jätettiin sensorifuusiot

ulkopuolelle, sillä tutkielmassa haluttiin keskittyä vain kamerapohjaisen SLAM:in riittävyteen ja tehokkuuteen kartan muodostamisessa sekä paikannuksessa. Tämän myötä algoritmejä, jotka hyödyntävät myös IMU:a tai muita sensoreita, kuten valotutkaa ei mukaan luettu taulukkojen testituloksiin.

Useissa tutkimuksissa tarkastellaan IMU:n integroimista Stereo-SLAM-järjestelmiin, mutta sen lisääminen ei ole kaikissa tilanteissa optimaalisin ratkaisu. Kuten Amado Antoninin ja muiden esittämässä artikkelissa [23] mainitaan nykypäiväiset IMU:t ovat vielä erittäin parametriherkkiä eri lentoskenaarioissa. Tulokset osoittaa, että vaikeissa lentoskenaarioissa optimaaliseksi osoittautuneet parametrit tuottavat heikompia tuloksia yksinkertaisimmissa lentoskenaarioissa. Parametriherkkyyden myötä ainoastaan kameralla toimivat järjestelmät toimivat siinä mielessä tasaisemmin. Tämä havainto voi osittain myös selittää Dajian Lei:n ja muiden esittämiä testituloksia [1], joiden mukaan pelkästään stereosensoria käyttävä ORB-SLAM3 suoriutui paremmin yksinkertaisista sekvensseistä.

Parempien tulosten saamiseksi olisi siten ideaalista löytää ratkaisu parametriherkkyyden ratkaisemiseksi. Näitä ratkaisuja voisivat muun muassa olla laskenta-tehon kasvattaminen. Tämä mahdollistaisi tiheämmän optimoinnin, joka vähentäisi parametriherkkyyttä. Stereo-SLAM:in edistäminen ilman IMU:a voisi myös olla mahdollista. Yökameraan perustuva SLAM voisi tarjota toimivan ratkaisun pimeissä toimintaympäristöissä. Julkisesti saatavilla olevia tietoaaineistoja on kuitenkin rajallisesti tai ei lainkaan. Tästä johtuen tätä lähestymistapaa ei sisällytetty tutkielmaan. Myös ilman IMU:a toimivissa algoritmeissa törmätään prosessointitehon tuottamaan kulmakiveen. Kuten Amaldo Antonini ja muiden esittämässä artikkelissa [23] todetaan, korkealaatuiset kamerat tuovat selkeän edun paikantamisessa. Nykyaikaiset algoritmit eivät kuitenkaan heidän mukaan kykene prosessoimaan tietoa tarpeeksi nopeasti dynaamisissa ympäristöissä. Tästä johtuen prosessointitehon edistäminen on vieläkin avainosassa Stereo-SLAM:in jatkokehityksen kannalta.

## 6 Yhteenveto

SLAM on kehitelty vaihtoehtoiseksi menetelmäksi satelliittipaikannukseen perustuville paikannusmenetelmille. Tutkielmassa tuodaan esille useampi SLAM-menetelmä vastaten ensimmäiseen tutkimuskysymykseen. Näitä menetelmiä ovat valotutkaan perustuva, monokulaariseen kameraan perustuva sekä syvyyskameraan perustuva SLAM. Jokainen vaihtoehtoinen SLAM-menetelmä ilmenee jollain määrin ongelmalliseksi, johon stereokamera tarjoaa tehokkaamman vaihtoehdon. Ongelmia ovat muun muassa valotutkan herkkyys valolle ja sen hinta [3], [9], syvyyden mittauksen puute monokulaarisessa kamerassa [9], [10] sekä syvyyskameran herkkyys kirkkaalle valolle. Syvyyskamera olisi ollut sensoreista toiseksi ideaalisin vaihtoehto SLAM-menetelmälle, mutta stereokameran laajempien käyttömahdollisuuksien vuoksi stereokamera valittiin tutkielman pääsensoriksi

Tutkielmassa tarkastellut Stereo-SLAM-menetelmät tuottavat odotusten mukaista suorituskykyä, mutta jättivät aukon tulevaisuuden kehitysmahdollisuuksille. Korkeastrukturisissa ympäristöissä Stereo-SLAM:in tulokset ylsivät 100% onnistumisprosenttiin vähintään kahdella tutkielmassa esitetyllä algoritmilla sekä alle 1 % virhearvioon, kuten taulukossa 4.1 on esitetty. Varsinkin algoritmit, kuten ORB-SLAM2, jotka perustuvat avainpisteisiin suoriutuivat hyvin korkeastrukturisissa ympäristöissä. Edeltä esitetyistä tuloksista poissuljettiin erittäin dynaaminen Blackbird-tietoaineisto, koska sen ääriolosuhteet ja suuret nopeudet eivät ole suoraan verrattavissa muihin tutkielmassa käytettyihin ympäristöihin.

Kun ympäristö muuttuu matalastrukturiseksi tuloksissa ilmenee laskukaava. Linjapiirteitä havaitsevat algoritmit, kuten PL-SLAM suoriutuu yleisesti paremmin yltäen 100% onnistumisprosenttiin testatuissa matalastrukturisissa ympäristöissä. Linjapiirteitä huomioivat algoritmit ovat siten erittäin hyödyllisiä matalastrukturisissa sisätilaympäristöissä [14]. Kuitenkaan kaikissa ympäristöissä, kuten matalatekstuurisessa KITTI moottoritie sekvenssissä 01, linjapiirteiden havainnoinnilla ei pystytä parantamaan tulosta.

Tuloksista huomataan dynaamisuuden ja valoisuuden olevan suurimmat muuttajat Stereo-SLAM:in tuloksiin. Erittäin dynaamisessa Blackbird-ympäristössä huomataan suuri pudotus ORB-SLAM2:n luotettavuudessa. VINS-algoritmi ilman IMU:a suoriutuu hieman luotettavammin, mutta vaikeimmissa ympäristöissä onnistumisprosentti on vaivaiset 83.3 % oikein valitulla kameran suuntauksella. Valoisuuden vaikutus tulee ilmi taulukossa 3.8, jossa tulokset tipahtavat vähintään 43 % nopeasta mutta valoisasta MH03-sekvenssistä, nopeaan mutta pimeään MH05-sekvenssiin.

Vaikka tarkkuuden tulokset heikentyvät matalastrukturisissa ympäristöissä, useampi tutkielmassa esitetty algoritmi kykeni suoriutumaan 100 % onnistumisprosentilla verrattuna korkeastrukturiseen ympäristöön. Huonommalla tuloksella tarkoitetaan, että oikein valitulla algoritmilla saavutetaan silti vaikeissa skenaariossa alle 1.5 % virhearvioita. Stereo-SLAM:illa on kuitenkin parempi kyky toimia korkeastrukturisessa ympäristössä algoritmista riippumatta. Hyödyntämällä myös linjapiirteitä tai pikseli-intensiteettiä havaitsevia algoritmeja voidaan parantaa huomattavasti Stereo-SLAM:in luotettavuutta myös matalastrukturisissa sekä dynaamisemmissa ympäristöissä. Voidaan siten todeta, että nyky Stereo-SLAM-algoritmeilla on kyky toimia luotettavasti molemmissa toimintaympäristöissä, vaikka ympäristömuuttujat olisivat haastavia. Korkeastrukturinen ympäristö luokitellaan kuitenkin testitulosten myötä ideaalimmaksi runsaiden piirteiden myötä.

# Lähdeluettelo

- [1] D. Lei, L. Luo, S. Chen ja W. Li, "PL-ALF: A Novel Point-Line Feature Autonomous Localization and Flight Framework Based on Multisensor Fusion and Optimization", *IEEE Transactions on Instrumentation and Measurement*, vol. 74, 2025. DOI: 10.1109/TIM.2024.3522670.
- [2] R. Giubilato, W. Stürzl, A. Wedler ja R. Triebel, "Challenges of SLAM in extremely unstructured environments: the DLR Planetary Stereo, Solid-State LiDAR, Inertial Dataset", *IEEE Robotics and Automation Letters*, vol. 7, 2022, ISSN: 2377-3766. DOI: 10.1109/LRA.2022.3188118.
- [3] N. Abdelaziz ja A. El-Rabbany, "INS/LIDAR/Stereo SLAM Integration for Precision Navigation in GNSS-Denied Environments", *Sensors*, vol. 23, 2023, ISSN: 14248220. DOI: 10.3390/s23177424.
- [4] V. Vemulapati ja D. Chen, "FSLAM: an Efficient and Accurate SLAM Accelerator on SoC FPGAs", teoksessa *FPT 2022 - 21st International Conference on Field-Programmable Technology, Proceedings*, Hong Kong: Institute of Electrical ja Electronics Engineers Inc., 2022. DOI: 10.1109/ICFPT56656.2022.9974562.
- [5] A. Hussain, A. R. Memon, H. Wang, Y. Wang, Y. Miao ja X. Zhang, "S-VIT: Stereo Visual-Inertial Tracking of Lower Limb for Physiotherapy Rehabilitation in Context of Comprehensive Evaluation of SLAM Systems", *IEEE*

- Transactions on Automation Science and Engineering*, vol. 18, 2021, ISSN: 15583783. DOI: 10.1109/TASE.2020.3010415.
- [6] J. Qiao, J. Guo ja Y. Li, "Simultaneous localization and mapping (SLAM)-based robot localization and navigation algorithm", *Applied Water Science*, vol. 14, 2024, ISSN: 21905495. DOI: 10.1007/s13201-024-02183-6.
- [7] I. A. Kazerouni, L. Fitzgerald, G. Dooly ja D. Toal, *A survey of state-of-the-art on visual SLAM*, 2022. DOI: 10.1016/j.eswa.2022.117734.
- [8] D. Sharafutdinov et al., "Comparison of modern open-source visual SLAM approaches", 2023. DOI: 10.48550/arXiv.2108.01654.
- [9] Y. Li et al., "A Review of Simultaneous Localization and Mapping Algorithms Based on Lidar", *World Electric Vehicle Journal*, vol. 16, 2025, ISSN: 20326653. DOI: 10.3390/wevj16020056.
- [10] R. Mur-Artal ja J. D. Tardos, "ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras", *IEEE Transactions on Robotics*, vol. 33, 2017, ISSN: 15523098. DOI: 10.1109/TR0.2017.2705103.
- [11] Y. Lu, Z. Xue, G. S. Xia ja L. Zhang, "A survey on vision-based UAV navigation", *Geo-Spatial Information Science*, vol. 21, 2018, ISSN: 10095020. DOI: 10.1080/10095020.2017.1420509.
- [12] S. Zhang, L. Zheng ja W. Tao, "Survey and Evaluation of RGB-D SLAM", *IEEE Access*, vol. 9, 2021, ISSN: 21693536. DOI: 10.1109/ACCESS.2021.3053188.
- [13] M. Kasper, S. McGuire ja C. Heckman, *A Benchmark for Visual-Inertial Odometry Systems Employing Onboard Illumination*. Macau, China: IEEE, 2019, ISBN: 9781728140049. DOI: 10.1109/IR0S40897.2019.8968554.

- 
- [14] R. Gomez-Ojeda, F. A. Moreno, D. Zuñiga-Noël, D. Scaramuzza ja J. Gonzalez-Jimenez, "PL-SLAM: A Stereo SLAM System Through the Combination of Points and Line Segments", *IEEE Transactions on Robotics*, vol. 35, 3 2019, ISSN: 19410468. DOI: 10.1109/TR0.2019.2899783.
- [15] J. Lee ja S. Y. Park, "PLF-VINS: Real-Time Monocular Visual-Inertial SLAM with Point-Line Fusion and Parallel-Line Fusion", *IEEE Robotics and Automation Letters*, vol. 6, s. 7033–7040, 2021, ISSN: 23773766. DOI: 10.1109/LRA.2021.3095518.
- [16] T. Pire, T. Fischer, J. Civera, P. D. Cristoforis ja J. J. Berlles, *Stereo Parallel Tracking and Mapping for robot localization*. Hamburg, Germany: IEEE, 2015, ISBN: 9781479999941. DOI: 10.1109/IR0S.2015.7353546.
- [17] C. Forster, Z. Zhang, M. Gassner, M. Werlberger ja D. Scaramuzza, "SVO: Semidirect Visual Odometry for Monocular and Multicamera Systems", *IEEE Transactions on Robotics*, vol. 33, s. 249–265, 2017, ISSN: 15523098. DOI: 10.1109/TR0.2016.2623335.
- [18] B. Zhang, X. Ma, H. J. Ma ja C. Luo, "DynPL-SVO: A Robust Stereo Visual Odometry for Dynamic Scenes", *IEEE Transactions on Instrumentation and Measurement*, vol. 73, 2024, ISSN: 15579662. DOI: 10.1109/TIM.2023.3348882.
- [19] J. Enge, J. Stuckler ja D. Cremers, *Large-Scale Direct SLAM with Stereo Cameras*. Hamburg, Germany: IEEE, 2015, ISBN: 9781479999941. DOI: 10.1109/IR0S.2015.7353631.
- [20] a. U. A. Geiger P. Lenz, "Are we ready for autonomous driving? The KITTI vision benchmark suite," Providence, RI, USA: IEEE, 2012, ISBN: 9781467312288. DOI: 10.1109/CVPR.2012.6248074.

- 
- [21] M. Burri et al., "The EuRoC micro aerial vehicle datasets", *International Journal of Robotics Research*, vol. 35, s. 1157–1163, 2016, ISSN: 17413176. DOI: 10.1177/0278364915620033.
- [22] T. Pire, M. Mujica, J. Civera ja E. Kofman, "The Rosario dataset: Multisensor data for localization and mapping in agricultural environments", *International Journal of Robotics Research*, vol. 38, 6 2019, ISSN: 17413176. DOI: 10.1177/0278364919841437.
- [23] A. Antonini, W. Guerra, V. Murali, T. Sayre-McCord ja S. Karaman, "The Blackbird UAV dataset", *International Journal of Robotics Research*, vol. 39, 10-11 2020, ISSN: 17413176. DOI: 10.1177/0278364920908331.
- [24] Y. Liu, T. Jia, S. Guo, Y. Liu, K. Zhang ja Z. Du, "A Robust and Accurate Stereo SLAM Based on Learned Feature Extraction and Matching", *IEEE Transactions on Instrumentation and Measurement*, vol. 74, 2025, ISSN: 15579662. DOI: 10.1109/TIM.2025.3614846.
- [25] S. Wen, S. Tao, X. Liu, A. Babiarz ja F. R. Yu, "CD-SLAM: A Real-Time Stereo Visual-Inertial SLAM for Complex Dynamic Environments with Semantic and Geometric Information", *IEEE Transactions on Instrumentation and Measurement*, vol. 73, s. 1–8, 2024, ISSN: 15579662. DOI: 10.1109/TIM.2024.3396858.
- [26] M. Aladem ja S. A. Rawashdeh, "Lightweight visual odometry for autonomous mobile robots", *Sensors (Switzerland)*, vol. 18, 2018, ISSN: 14248220. DOI: 10.3390/s18092837.