

VILNIAUS UNIVERSITETAS

Rokas  
JUREVIČIUS

# Bepiločio orlaivio vizualinės lokalizacijos grįstos dalelių filtru tyrimas skrydžiams mažame aukštyje

**DAKTARO DISERTACIJOS SANTRAUKA**

Technologijos mokslai,  
Informatikos inžinerija (07T)

---

VILNIUS 2019

Disertacija rengta 2014 - 2018 metais Vilniaus Universitete, Duomenų mokslo ir skaitmeninių technologijų institute.

**Mokslinis vadovas:**

**dr. Virginijus Marcinkevičius** (Vilniaus Universitetas, technikos mokslai, informatikos inžinerija - 07T)

Gynimo taryba:

Pirmininkas – **prof. habil. dr. Julius Žilinskas** (Vilniaus Universitetas, technikos mokslai, informatikos inžinerija - 07T)

Nariai:

**dr. Saeed Yahyanejad** (Joanneum Research Forschungsgesellschaft mbH, technikos mokslai, informatikos inžinerija - 07T)

**prof. dr. Tomas Krilavičius** (Vytauto Didžiojo Universitetas, technikos mokslai, informatikos inžinerija - 07T)

**doc. dr. Tomas Luneckas** (Vilniaus Gedimino Technikos Universitetas, technikos mokslai, elektros inžinerija - 01T)

**prof. dr. Rimantas Vaicekaskas** (Vilniaus Universitetas, technikos mokslai, informatikos inžinerija - 07T)

Disertacija ginama viešame Gynimo tarybos posėdyje 2019 m. XX mėn. XX d. XX val. Vilniaus Universitete, Duomenų mokslo ir skaitmeninių technologijų instituto XX auditorijoje. Adresas: , tel. +370 ; el. paštas XX .

Disertaciją galima peržiūrėti Vilniaus Universiteto bibliotekoje ir VU interneto svetainėje adresu: [www.vu.lt/lt/naujienos/ivykiu-kalendorius](http://www.vu.lt/lt/naujienos/ivykiu-kalendorius)

VILNIUS UNIVERSITY

Rokas  
JUREVIČIUS

# Investigation of Particle Filter Based Visual Localization for Unmanned Aerial Vehicle Flights at Low-Altitude

**SUMMARY OF DOCTORAL DISSERTATION**

Technological sciences,  
Informatics engineering (07T)

---

VILNIUS 2019

The dissertation work was carried out at Vilnius University Institute of Data Science and Digital Technologies from 2014 to 2018.

**Academic supervisor**

**Dr. Virginijus Marcinkevičius** (Vilnius University, technical sciences, informatics engineering - 07T)

This doctoral dissertation will be defended in a public meeting of the Dissertation Defence Panel:

Chairman – **Prof. Habil. Dr. Julius Žilinskas** (Vilnius University, technical sciences, informatics engineering - 07T)

Members:

**Dr. Saeed Yahyanejad** (Joanneum Research Forschungsgesellschaft mbH, technical sciences, informatics engineering - 07T)

**Prof. Dr. Tomas Krilavičius** (Vytautas Magnus University, technical sciences, informatics engineering - 07T)

**Assoc. Prof. Dr. Tomas Luneckas** (Vilnius Gediminas Technical University, technical sciences, electrical engineering - 01T)

**Prof. Dr. Rimantas Vaicekaskas** (Vilnius University, technical sciences, informatics engineering - 07T)

The dissertation shall be defended at a public meeting of the Dissertation Defence Panel at XX on XX in Room XX of the Vilnius University Institute of Data Science and Digital Technologies.

Address: street, house No., Room No., city, Lithuania

Tel. +370; e-mail:

The text of this dissertation can be accessed at the libraries of Vilnius University, as well as on the website of Vilnius University: [www.vu.lt/lt/naujienos/ivykiu-kalendorius](http://www.vu.lt/lt/naujienos/ivykiu-kalendorius)

# SANTRAUKA

## 1. Įvadas

Tobulėjant įterptinėms kompiuterinių skaičiavimų sistemoms, kompiuterinės regos ir dirbtinio intelekto algoritmai vis plačiau naudojami autonominėms robotinėms platformoms – bepiločiams orlaiviams (toliau – BPO), antžeminiams robotams, miniatiūriniais povandeniniams laivams ir kitoms išmanioms robotinėms platformoms. Šių algoritmų naudojimas įgalina robotus vykdyti naujas ir išmanias funkcijas, pasiekti autonomiškumą ir svarbiausia – jie tampa saugesniais naudojant realiomis sąlygomis. Autonominių skrydžių metu naudojamas GPS signalas gali būti trikdomas ar prarastas, dėl šios priežasties lėktuvas gali būti perimtas ar sudaužytas. Tam, kad būtų užtikrintas BPO skrydžių saugumas ir atsparumas išoriniams poveikiams būtina taikyti lokalizacijos metodus nenaudojančius išorinių signalų. Įprastos auto-pilotavimo sistemos, kurios yra naudojamos BPO, neturi alternatyvios lokalizacijos sistemos leidžiančios orlaiviui saugiai naviguoti dingus GPS signalui. Alternatyvūs lokalizacijos metodai turėtų lokalizuoti orlaivį pasitelkdami tik įrangą orlaivyje. Šiuo metu pažangiausi kompiuterinės regos algoritmai gali lokalizuoti orlaivį naudojant orlaivyje įmontuotos kameros vaizdus. Orlaivio pozicijos nustatymo procesas apdorojant optinės kameros vaizdus vadinamas – vizualine lokalizacija. Vizualinės lokalizacijos algoritmai suteikia galimybę saugiai tęsti autonominį skrydį praradus GPS signalą.

### 1.1 Problemos formulavimas

BPO autonominiai skrydžiai neįmanomi jei GPS signalas yra neprieinamas arba sutrikdytas. Ši problema ypač aktuali karinių situacijų metu, kai bet kokie radijo signalai yra blokuojami. Lokalizuoti orlaivį galima radijo bangomis grįstais metodais, tačiau visi radijo bangų signalai gali būti sutrikdyti ir šie metodai tampa nepanaudojami.

Kompiuterinės regos sritis sprendžia orlaivių lokalizacijos problemą, kuomet GPS signalas yra neprieinamas. Orlaivio pozicija erdvėje apskaičiuojama apdorojant optinių kamerų vaizdo srautus taikant optinio srauto, vizualinės odometrijos, vienalaikės lokalizacijos ir žemėlapių sudarymo (angl. *Simultaneous Localization And Mapping*, toliau – SLAM) arba vaizdų registravimo algoritmus. Nors šie algoritmai pasiekia stebėtinai gerus rezultatus skrendant pastatų viduje ar prie žemės paviršiaus (skrydžio aukštis iki 100 metrų), tačiau jie retai taikomi nedidelio

aukščio skrydžiams (100 – 3000 metrų). Vizualiniai SLAM algoritmai pasiekia tikslią orlaivio pozicijos lokalizaciją be besikaupiančių paklaidų, tačiau šie algoritmai geriausius rezultatus rodo skrendant prie žemės paviršiaus (5 – 15 metrų aukštyje) arba patalpose. Karo pramonėje raketų navigacijai be GPS signalo naudojama lokalizacijos technologija „TERCOM“ (angl. *Terrain Contour Matching*). Ši technologija naudoja aukščio jutiklį ir derina aukščio parodymus su *a priori* žinomų aukščių žemėlapiu. „TERCOM“ naudojama ir šiandien, tačiau ji nėra labai tiksli, todėl ji paprastai naudojama raketoms, kurioms nereikalinga labai tiksli pozicionavimo sistema. „DSMAC“ (angl. *Digitized Scene-Mapping Area Correlator*) sistema veikia labai panašiai kaip „TERCOM“, tik vietoje aukščio parodymų naudojamas skaitmeninės kameros vaizdas ir ortofotografiniai vietovės žemėlapiai. „DSMAC“ technologijos trūkumas – prastas atsparumas pokyčiams žemėlapiuose. Tam, kad sistema veiktų tiksliai, reikalingi prieš pat skrydį sudaryti palydoviniai žemėlapiai (netgi kitokiu kampu krentantys šešėliai gali paveikti navigacijos tikslumą).

Šioje disertacijoje nagrinėjami kompiuterinės regos algoritmai tinkami BPO lokalizacijai nedideliame aukštyje (100-3000 metrų) ir siūlomas naujas algoritmas, kuris pagerintų lokalizacijos tikslumą, lyginant su šiuo metu geriausias vizualinės lokalizacijos algoritmais.

## 1.2 Tyrimo objektas

Tyrimo objektas – kompiuterinės regos, vaizdų apdorojimo ir dalelių filtravimo algoritmai taikomi bepiločių orlaivių vizualinei lokalizacijai, kurie yra tinkami vykdymui įterptinėse platformose realiu laiku.

## 1.3 Tyrimo tikslas ir uždaviniai

Šio tyrimo tikslas – pasiūlyti naują vizualinės lokalizacijos algoritmą, tinkamą bepiločio orlaivio lokalizacijai be GPS signalo nedidelio aukščio (100-3000 metrų) ir didelio atstumo ( $> 1$  kilometras) skrydžiams. Jis veiktų įterptinėse skaičiavimų platformose ir būtų tikslesnis, greitesnis ir atsparesnis netikslumams žemėlapyje lyginant su pažangiausiais lokalizacijos algoritmais. Šiam tikslui pasiekti formuluojami šie uždaviniai:

1. Palyginti ir identifikuoti tinkamiausią vaizdų panašumo matą žemės paviršiaus nuotraukų panašumui apskaičiuoti, nedidelio aukščio BPO skry-

džio metu.

2. Palyginti dalelių atrankos metodus naudojamus dalelių filtruose tam, kad būtų identifiкуotas mažiausiai skaičiavimo resursų reikalaujantis atrankos metodas.
3. Pasiūlyti ir palyginti vaizdų panašumo į tikimybę konversijos funkcijas siekiant pagerinti dalelių filtro lokalizacijos algoritmo tikslumą.
4. Pasiūlyti naują dalelių filtro lokalizacijos algoritmą (ar jo modifikaciją), įvertinti jo tikslumą, greitaveiką, atsparumą netikslumams žemėlapiuose ir gebėjimą sumažinti besikaupiančias vizualios odometrijos paklaidas.
5. Palyginti pasiūlytą dalelių filtro lokalizacijos algoritmą su pažangiausiais vizualiniais SLAM algoritmais.
6. Pasiūlyti naują metodą įterptinių lygiagrečių skaičiavimo platformų palyginimui, kuris leistų identifiкуoti energetiškai efektyviausią platformą, tinkamą kompiuterinės regos uždaviniams.

#### 1.4 Naudojami moksliniai tyrimo metodai

1. Atlikta literatūros apžvalga, analizuojami naujausi moksliniai straipsniai, identifiкуojami pažangiausi algoritmai, sprendžiantys BPO vizualinę lokalizaciją be GPS signalo.
2. Eksperimentinių tyrimu metu panaudotas kiekybinis ir kokybinis informacijos rinkimas, matuojamas algoritmų ar atskirų algoritmo komponentų veikimas.
3. Statistiniai metodai, tokie kaip Stjudento t-testas, Šapiro-Vilko (Shapiro-Wilk) testas, buvo panaudoti gautų rezultatų patikimumui įvertinti taikant statistinį hipotezės tikrinimą.
4. Buvo naudojamas konstruktyvinis tyrimas, kuriuo siekiama pasiūlyti patobulinimus egzistuojantiems sprendimams ir pasiūlyti naujus metodus gerinančius esamas teorijas.
5. Atliekant eksperimentus ir realizuojant pasiūlytą algoritmą, bei jo komponentus buvo taikomi programinės įrangos kūrimo metodai.

#### 1.5 Tyrimo mokslinė vertė

1. Pasiūlyta vizualinio dalelių filtro lokalizacijos algoritmo modifikacija. Pasiūlyta nauja dalelių filtro komponentė – vaizdo panašumo į tikimybę konversijos funkcija. Pamatuota funkcijos parametro daroma įtaka lokalizacijos

- tikslumui, vykdymo greičiui ir atsparumui netikslumams žemėlapyje. Disertacijoje parodoma, kad keičiant funkcijos parametro vertę galima pasiekti algoritmo tikslumo ir atsparumo (žemėlapio netikslumams) derinimą.
2. Pasiūlytas dalelių filtro lokalizacijos algoritmas pagerino pažangiausių ORB-SLAM ir SVO algoritmų lokalizacijos tikslumą. Pritaikius adaptyvų dalelių skaičių ir vaizdų panašumo konversijos į tikimybę funkciją, dalelių filtro lokalizacijos algoritmo tikslumas ir greitis buvo ženkliai pagerinti.
  3. Pasiūlytas įterptinių skaičiavimo platformų palyginimo metodas leido identifikuoti energetiškai efektyviausią lygiagrečių skaičiavimų platformą, tinkamą kompiuterinės regos algoritmų vykdymui. Palygintos trys platformos – platforma su NVIDIA TX1 įterptiniu grafiniu procesoriumi identifikuota greičiausia ir energetiškai efektyviausia.

## 1.6 Tyrimo praktinė vertė

1. Pasiūlytas nauja dalelių filtro lokalizacijos algoritmo modifikacija. Šio algoritmo realizacija yra viešai prieinama kitiems tyrėjams, norintiems jį palyginti ar išplėsti.
2. Eksperimentiniais rezultatais parodoma, kad panaudojus parametrizuojamą vaizdų panašumo į tikimybę konversijos funkciją galima valdyti algoritmo atsparumą žemėlapio netikslumams arba didinti lokalizacijos tikslumą. Naudojant logistinę konversijos funkciją ir keičiant funkcijos parametro vertę pasiekiamas lokalizacijos tikslumo ir atsparumo derinimas, t. y. galime pagerinti tikslumą arba pagerinti lokalizacijos atsparumą netikslumams vaizduose.
3. Sukurtas naujas duomenų rinkinys, kuriame pateikiami orlaivio kameros vaizdai ir jo pozicijos, simuliuojant bepiločio orlaivio skrydį virš ortofotografinio žemėlapio robotikos simulatoriuje. Duomenų rinkinys tinkamas vizualinės odometrijos, vizualinės lokalizacijos ir SLAM algoritmų vertinimui ir testavimui.
4. Skirtingi dalelių atrankos metodai buvo palyginti su KL divergencijos (angl. *Kueller-Leiblach divergence*, t. y. matas, naudojamas įvertinti dviejų skirstinių panašumui) dalelių atranka. Šis dalelių atrankos metodas buvo pasirinktas tinkamiausiu, nes su nežymiais lokalizavimo tikslumo praradimais pagreitina algoritmo veikimą.
5. Pasiūlytas naujas metodas įterptinių skaičiavimų platformų energetinio efektyvumo vertinimui. Palygintos kelios pažangiausios įterptinės lygiagrečių



skaičiavimų platformos, iš kurių platforma naudojanti NVIDIA TX1 – parodė geriausias energetinio efektyvumo rezultatus ir kainos su greitimeika santykį.

### 1.7 Tyrimo ginamieji teiginiai

1. Logistinė vaizdų panašumo į tikimybę konversijos funkcija pagerina pasiūlyto dalelių filtro lokalizacijos algoritmo tikslumą ir atsparumą netikslumams žemėlapyje.
2. Pasiūlyta dalelių filtro lokalizacijos algoritmo modifikacija pagerina vizualinės odometrijos tikslumą ir sumažina besikaupiančią odometrijos paklaidą.
3. Pasiūlyta dalelių filtro lokalizacijos algoritmo modifikacija yra tikslesnė už pažangiausią vizualinio SLAM algoritmą ORB-SLAM.
4. Vizualinei BPO lokalizacijai tinkamiausias vaizdų panašumo matas yra Pirsono (Pearson) koreliacijos koeficientas (arba normalizuotas koreliacijos koeficientas) lyginant su kvadratų skirtumų sumos ir kryžminės koreliacijos matais.
5. KLD atrankos metodas pagreitina dalelių filtro lokalizacijos algoritmo veikimą mažindamas lokalizacijai reikalingą dalelių skaičių.
6. Pasiūlytas įterptinių skaičiavimo sistemų energetinio efektyvumo palyginimo metodas identifikuoja greičiausią ir energetiškai efektyviausią platformą kompiuterinės regos algoritmams vykdyti.

### 1.8 Tyrimo aprobavimas ir publikavimas

Disertacijos rezultatų pagrindu buvo paruošti 6 moksliniai straipsniai, publikacijų sąrašą galima rasti 5 skyriuje. Tyrimo rezultatai pristatyti 3 tarptautinėse konferencijose:

1. Jurevičius R. Energy Efficient Platform for Sobel Filter Implementation in Energy and Size Constrained Systems. The 3rd Workshop on AIEEE'15, Ryga, Latvija. 2015 m. Lapkričio 13–14 d.
2. Jurevičius R. Comparison of Image Similarity Functions and Sampling Algorithms in Vision-Based Particle Filter for UAV Localization. International Congress on Informatics CSIST'2016, Minskas, Baltarusija. 2016 m. Spalio 24–27 d.
3. Jurevičius R. Application of Vision-Based Particle Filter and Visual Odometry for UAV Localization. The 25th International Conference in Central

Europe on Computer Graphics, Visualization and Computer Vision WSCG '2017, Pilzenas, Čekijos Respublika. 2017 m. Gegužės 29–Birželio 2 d.

## 2. Tyrimo metodologija

### 2.1 Energetiškai efektyviausios įterptinės platformos lyginimo metodas

Šiuolaikiniai kompiuteriai yra pakankamai galingi sudėtingų kompiuterinės regos algoritmų vykdymui. Tačiau šios platformos yra didelės ir gali naudoti šimtus vatų elektros energijos. Energetiškai ribotose sistemose, tokiose kaip baterijomis maitinamas orlaivis, svarbu pasiekti kuo geresnį kompiuterinės regos algoritmų tikslumą su mažiausiomis energijos sąnaudomis. Šiuo metu populiariausi trys įterptinių skaičiavimų platformų tipai: daugelio branduolių procesoriai (angl. *multi-core CPU*), grafiniai procesoriai (angl. *Graphics Processing Unit*, GPU) ir vartotojo programuojamos loginių elementų matricos (angl. *Field Programmable Gate Array*, FPGA).

Šiame tyrime yra lyginama daugelio branduolių platforma Parallella (naudojanti Epiphany spartinamąjį procesorių Epiphany) su Radxa Rock2 (kuri naudoja ARM Mali T764 grafinį procesorių) ir Airvision Core X1 (kuri naudoja NVIDIA Tegra X1 grafinį procesorių su 256 CUDA branduoliais). Platformos yra palyginamos pagal energijos kiekį (džiauliais) sunaudotą vienos nuotraukos Sobelio (Sobel) filtro skaičiavimams. Sobelio filtras skaičiuojamas atliekant dvi konvoliucijos operacijas: vieną vertikaliai, kitą horizontaliai filtravimui:

$$I_v = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix} * T \quad (1)$$

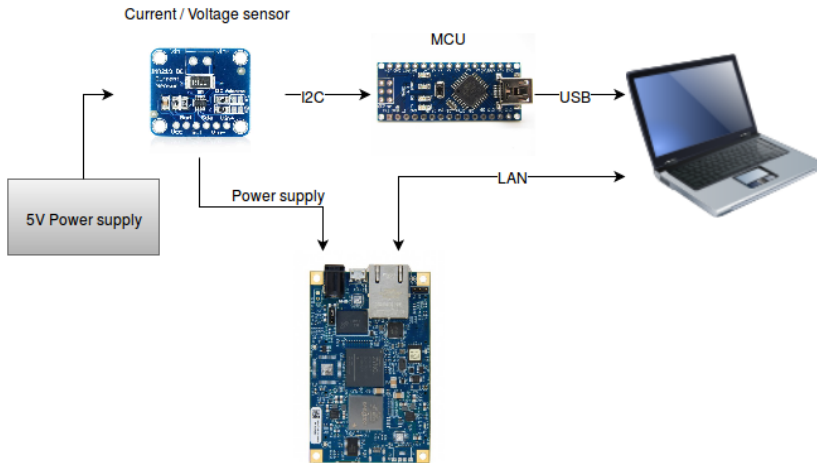
$$I_h = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * T \quad (2)$$

čia:

- $T$  pradinė nuotrauka,
- $I_v$  ir  $I_h$  yra nuotraukos gaunamos po vertikalios ir horizontalios Sobelio filtro apskaičiavimo,
- simbolis "\*" žymi konvoliucijos operaciją.

Skaičiuojant Sobelio filtrą, matuojamos trijų programos veikimo fazių laiko trukmės:

**Rašymas.** Matuojamas laikas kiek trunka įkelti nuotrauką į spartinančiojo prietaiso atmintį.



Pav. 1: Įtampos ir srovės jutiklio pajungimas prie įterptinės skaičiavimų platformos.

**Vykdymas.** Sobelio filtro skaičiavimo trukmė.

**Skaitymas.** Gautų rezultatų nuskaitymas į centrinio procesoriaus atmintį iš spartinančiojo prietaiso atminties.

Įtampos ir srovės jutiklis INA219, prijungtas maitinimo šaltinio grandyje, yra naudojamas energijos sąnaudoms matuoti. Programos vykdymo metu 1 kHz dažniu registruojami įtampos ir srovės matavimai ir galutinės energijos sąnaudos gaunamos integruojant įtampą ir srovę:  $E = \int_0^t P(t)dt$ . Jutiklio pajungimas pateikiamas pav. 1. Algoritmas 1 yra naudojamas atlikti Sobelio filtro vykdymo trukmės ir energijos sąnaudų matavimams.

## 2.2 Duomenų rinkiniai vizualinės lokalizacijos algoritmų vertinimui

Tyrimo metu buvo sukurti du nauji duomenų rinkiniai tinkami vizualinės lokalizacijos algoritmų vertinimui skrydžiams nedideliame aukštyje (200-3000 metrų).

**Testinio skrydžio duomenų rinkinys.** Šis duomenų rinkinys buvo sukurtas atliekant maždaug 1 kilometro skrydį su lėktuvu „Spartan” Lentvaryje netoli Vilniaus. Skrydžio metu kameros vaizdai buvo įrašyti 90 kadrų per sekundę dažniu, kiekvienam kadru saugomos orlaivio koordinatės ir pokrypiai erdvėje. Šio nedidelio duomenų rinkinio užtenka patikrinti algoritmo veikimą realiomis

---

**Algoritmas 1** Įterptinių skaičiavimo platformų energetinio efektyvumo matavimo algoritmas
 

---

Įvestis: Nuotrauka  $T$

Išvestis: Vertikaliai filtruota nuotrauka  $I_v$ , horizontaliai filtruota nuotrauka  $I_h$ , rašymo laikas  $t_w$ , vykdymo laikas  $t_e$ , skaitymo laikas  $t_r$ , sunaudota energija  $E$

**function** MEASURESOBELFILTER( $T$ )

*startPowerMeasurementCapture()* ▷ Atskiroje gijoje pradedamas elektros įtampos ir srovės parametrų nuskaitymas

$t_w^{start} = timeNow()$  ▷ Išsaugomas dabartinis laikas

*upload(T)* ▷ Nuotraukos krovimo į spartintuvo atmintį

$t_w = timeNow() - t_w^{start}$  ▷ Paskaičiuojamas rašymo laikas

$t_e^{start} = timeNow()$

$I_v, I_h = performSobel(T)$  ▷ Vykdomas Sobelio filtro algoritmas spartintuve

$t_e = timeNow() - t_e^{start}$  ▷ Paskaičiuojamas vykdymo laikas

$t_r^{start} = timeNow()$

*download(T)* ▷ Nuotrauka siunčiama atgal į centrinio procesoriaus atmintį

$t_r = timeNow() - t_r^{start}$  ▷ Pamatuojamas skaitymo laikas

*endPowerMeasurementCapture()*

$E = retrieveEnergy()$  ▷ Suintegruojamos įtampos ir srovės matavimai gaunant energijos sąnaudas

**return**  $\langle I_v, I_h, t_w, t_e, t_r, E \rangle$

---

sąlygomis, tačiau duomenų kiekis yra nepakankamas įvertinti kitus lokalizacijos algoritmo aspektus – tikslumą skrendant įvairiomis trajektorijomis, atsparumą netikslumams žemėlapyje, naudojimą skirtingose aplinkose.

**Simuliuotų skrydžio duomenų rinkinys.** Šis duomenų rinkinys buvo sudarytas naudojant robotikos simulatorių „Gazebo“. Duomenų rinkinyje pateikiami duomenys iš 12 skirtingų skrydžių. Skrydžiai atlikti 200 ir 300 metrų aukščiuose virš miškingų ir urbanistinių teritorijų. Pasirinktos trys skrydžio trajektorijos: tiesės, apskritimo ir kvadrato. Skrydžių metu vaizdai iš kamerų buvo registruojami 50 kadrų per sekundę dažniu, su kiekvienu kadru saugoma orlaivio koordinatės ir pokrypiai erdvėje. Skrydžiai buvo atliekami simuliuotoje aplinkoje, kurios paviršiuje projektuojamas ortofotografinis žemėlapis. Šis duomenų rinkinys yra viešai prieinamas.

### 2.3 Dalelių filtru grįsta lokalizacija

Lokalizacija *a priori* žinomame žemėlapyje gali būti atliekama taikant dalelių filtro algoritmą. Lokalizacijos algoritmai, grįsti dalelių filtrais, pasiekia tikslius rezultatus be pozicijos dreifo. Dalelių filtras yra neparimetrinė Bajeso filtro realizacija. Dalelių filtrai naudojami apskaičiuoti sistemų būseną erdvėje naudojant elementus vadinamus dalelėmis. Lokalizacijos atveju, viena dalelė atitinka robotą (arba orlaivio) poziciją erdvėje. Bajeso teorema yra taikoma apskaičiuojant naujausią dalelių filtro būseną, kai gaunami nauji jutiklių duomenys. Dalelės yra filtruojamos siekiant atrinkti daleles, kurios yra artimiausios realiai orlaivio pozicijai. Prieš tai minėtos dalelės yra apibrėžiamos kaip baigtinis elementų rinkinys:

$$S_t := P_t^{(1)}, P_t^{(2)}, \dots, P_t^{(n)}, \quad (3)$$

čia:

- $S_t$  yra dalelių rinkinys laiko momentu  $t$ ,
- $P_t^{(i)}$  yra  $n$ -toji ( $1 \leq i \leq n$ ) dalelė rinkyje laiko momentu  $t$ ,
- $n$  yra dalelių kiekis.

Dalelių skaičius  $n$  gali būti fiksuotas arba specifiniais atvejais jis gali būti kintamas. Dalelių filtrai su kintamu dalelių skaičiumi vadinami adaptyviais dalelių filtrais.

### 2.4 Vaizdų panašumo matai

Dviejų vaizdų (žymėkime vaizdus  $I$  ir  $T$ ) panašumo vertinimui reikalingas skaitinis vaizdų panašumo metas. Vaizdų panašumo įvertis  $R$  laikomas matu jei įvertio vertė auga didėjant priklausomybei tarp dviejų vaizdų. Matas yra laikomas panašumo matu jei tenkina šias sąlygas:

1. Turi ribas:  $R(I, T) \leq R_0$ , kur  $R_0 = R(I, I)$ .
2. Refleksyvumas:  $R(I, T) = R_0$  jeigu ir tik jeigu  $I = T$ .
3. Simetrija:  $R(I, T) = R(T, I)$ .
4. Trikampė nelygybė:  $R(I, T)R(T, U) \leq [R(I, T) + R(T, U)]R(I, U)$

Vaizdų panašumo matai naudojami įvertinti ar orlaivio pozicijos hipotezė atitinka realią orlaivio poziciją. Panašumo matas skaičiuojamas tarp vaizdo iš lėktuvo kameros ir žemėlapio iškarpos, atitinkančios orlaivio pozicijos hipotezę. Vienas iš populiarių matų naudojamų įvertinti panašumui – normalizuota kvadratų skirtumų suma (dar žinoma kaip normalizuotas  $L_2$  normatyvas), ji apskaičiuojama naudojant šią formulę:

$$R = 1 - \frac{\sum_{x=0}^w \sum_{y=0}^h (T(x,y) - I(x,y))^2}{\sqrt{\sum_{x=0}^w \sum_{y=0}^h T(x,y)^2 \cdot \sum_{x=0}^w \sum_{y=0}^h I(x,y)^2}}, \quad (4)$$

čia:

- $w, h$  yra nuotraukos plotis ir aukštis,
- $x, y$  yra nuotraukos pikselio koordinatės  $X$  ir  $Y$  ašimis.

Šioje formulėje suma yra atimama iš vieneto, kad būtų patenkinta panašumo mato apibrėžimo sąlyga nr. 1, kitaip šis įvertis būtų laikomas skirtingumo matu. Kitas populiarius vaizdų panašumo matas – kryžminė koreliacija:

$$R = \frac{\sum_{x=0}^w \sum_{y=0}^h (T(x,y) \cdot I(x,y))}{\sqrt{\sum_{x=0}^w \sum_{y=0}^h T(x,y)^2 \cdot \sum_{x=0}^w \sum_{y=0}^h I(x,y)^2}}$$

Trečias panašumo matas naudojamas šioje disertacijoje yra Pirsono koreliacija arba normalizuota kryžmine koreliacija. Šis matas yra kryžminės koreliacijos plėtinys, kuriame papildomai atimamas nuotraukos pikselio verčių vidurkis. Vidurkio atėmimas suteikia matui atsparumą kontrasto ir ekspozicijos nesutapimams:

$$R = \frac{\sum_{x=0}^w \sum_{y=0}^h (T'(x,y) \cdot I'(x,y))}{\sqrt{\sum_{x=0}^w \sum_{y=0}^h T'(x,y)^2 \cdot \sum_{x=0}^w \sum_{y=0}^h I'(x,y)^2}}, \quad (5)$$

čia:

- $T'(x,y) = T(x,y) - \frac{\sum_{x'=0}^w \sum_{y'=0}^h T(x',y')}{w \cdot h}$
- $I'(x,y) = I(x,y) - \frac{\sum_{x'=0}^w \sum_{y'=0}^h I(x',y')}{w \cdot h}$
- $w, h$  nuotraukos plotis ir aukštis atitinkamai.

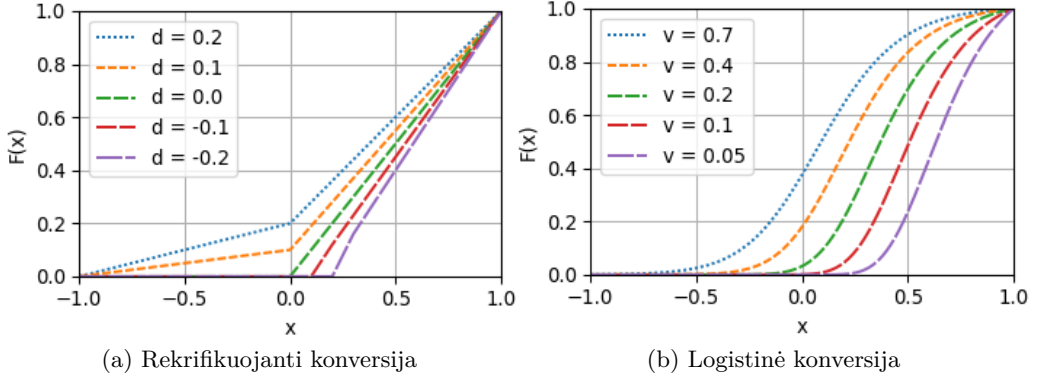
## 2.5 Vaizdų panašumo į tikimybę konversijos funkcija

Disertacijoje yra siūloma nauja dalelių filtro lokalizacijos algoritmo komponentė – vaizdo panašumo į tikimybę konversijos funkcija  $F(x, d)$ . Konversijai siūlomos šios dvi funkcijos:

1. rektifikuojanti konversijos funkcija:

$$F(x, d) = \begin{cases} 0, & \text{if } d < 0 \text{ and } x \leq |d| \\ (1 + |d|)(x - |d|) + d^2, & \text{if } d < 0 \text{ and } x > |d| \\ d(1 + x), & \text{if } d \geq 0 \text{ and } x \leq 0 \\ x(1 - d) + d, & \text{if } d \geq 0 \text{ and } x > 0 \end{cases}, \quad (6)$$

čia  $x$  panašumo mato vertė,  $d$  konversijos funkcijos parametro vertė.



Pav. 2: Rektifikuojančios ir logistinės konversijos funkcijų perdavimo grafikai.

2. logistinė konversijos funkcija:

$$F(x, v) = \frac{L(x, v)}{L(1, v)} \quad (7)$$

čia  $v$  konversijos funkcijos parametro vertė,  $L(x, v) = \frac{1}{(1+e^{-5x})^{\frac{1}{v}}}$ .

Pav. 2 pateikti konversijų funkcijų verčių grafikai režiuose  $[-1; 1]$ . Pasirinkti būtent šie režiai, kadangi Pirsono koreliacija yra naudojama apskaičiuoti vaizdų panašumui, o jos režiai yra  $[-1; 1]$ .

## 2.6 Dalelių filtro lokalizacija su diskriminavimu pagal Pirsono koreliaciją

Šioje disertacijoje pasiūlyta dalelių filtro lokalizacijos algoritmo modifikacija, kuri yra realizuota naudojant KL divergenciją. Divergencija apskaičiuoja reikiamą dalelių kiekį patikimai lokalizacijai taikant šią formulę:

$$n = \frac{1}{2\varepsilon} Z_{k-1, 1-\delta}^2 = \frac{k-1}{2\varepsilon} \left\{ 1 - \frac{2}{9(k-1)} + \sqrt{\frac{2}{9(k-1)}} z_{1-\delta} \right\}^3, \quad (8)$$

čia:

- $z_{1-\delta}$  yra normalinio skirstinio  $N(0, 1)$  viršutinis  $1 - \delta$  kvantilis;
- $\varepsilon$  yra diskretaus dalelių tikimybės skirstinio viršutinė riba;
- $k$  dalelėmis užimtų erdvių kiekis.

**Algoritmas 2** Pasiūlytas Dalelių filtro lokalizacijos algoritmas

Įvestis: dalelių rinkinys  $S_{t-1}$  gaunamas iš praeitos iteracijos, naujausia nuotrauka iš kameros  $T$ , konversijos funkcijos parametras  $v$  (jei naudojamas)

Išvestis: nuspėta orlaivio pozicija  $\langle x', y', \theta'_{yaw} \rangle$  žemėlapiu koordinatinių sistemoje

**function** FILTERPARTICLES( $S_{t-1}, T, v$ )

$H = 0, k = 1, S_t = \emptyset, i = 0$

**do**

$P^{(i)} \sim S_{t-1}$   $\triangleright$  Atrenkama dalelė iš dalelių rinkinio

$PropagateParticle(P^{(i)})$   $\triangleright$  Dalelė yra pajudinama taikant judesio modelį

$I = ExtractMapImage(P^{(i)})$   $\triangleright$  Paimama atitinkama žemėlapiu dalis pagal

dalelės poziciją

$r_t^{(i)} = CalcSimilarity(T, I)$   $\triangleright$  Paskaičiuojamas vaizdų panašumo matas

taikant Pirsono koreliaciją

$s_t^{(i)} = F(r_t^{(i)}, v)$   $\triangleright$  Perskaičiuojama tikimybė iš panašumo mato taikant

konversijos funkciją

$H = H + s_t^{(i)}$

$S_t = S_t \cup P^{(i)}$

$\triangleright$  Dalelė įdedama į naują dalelių rinkinį

**if**  $P^{(i)}$  patenka į neužimtą erdvę **then**

$bin = užimta$

$\triangleright$  Erdvė žymima kaip užimta

$k = k + 1$

$\triangleright$  Padidinamas užimtų erdvių skaitiklis

$i = i + 1$

$\triangleright$  Padidinamas dalelių skaitiklis

**while**  $i < \frac{1}{2\epsilon} Z_{k-1, 1-\delta}^2$

$\triangleright$  Kartojama kol pasiekama K-L riba

$n = i$

$\triangleright$  Išsaugojame dalelių skaičių

$x' = 0, y' = 0, \theta'_{yaw} = 0$

**for**  $i = 1 .. n$  **do**

$b_t^{(i)} = s_t^{(i)} / H$

$\triangleright$  Paskaičiuojamas dalelės svoris

$x' = x' + (b_t^{(i)} * x_t^{(i)})$

$\triangleright$  Visų dalelių koordinatinių svertinė suma

$y' = y' + (b_t^{(i)} * y_t^{(i)})$

$\triangleright x_t^{(i)}, y_t^{(i)}$  yra  $i$ -osios dalelės koordinatė žemėlapyje

$\theta'_{yaw} = \theta'_{yaw} + (b_t^{(i)} * \theta_t^{(i)})$

$\triangleright$  Nuspėtos pozicijos judėjimo kryptis  $\theta'_{yaw}$

**return**  $\langle x', y', \theta'_{yaw} \rangle$



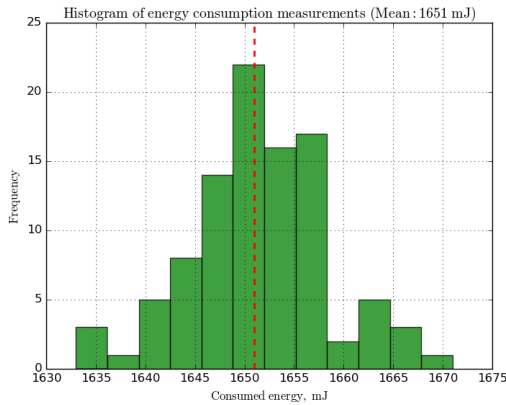
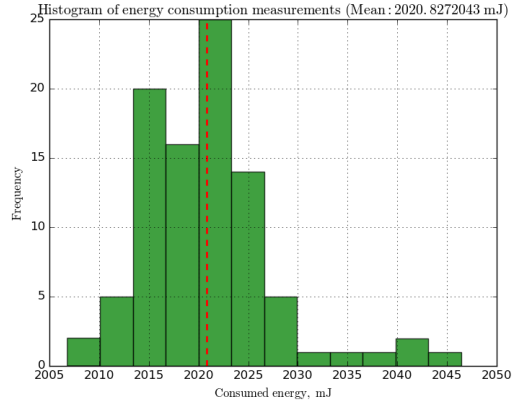
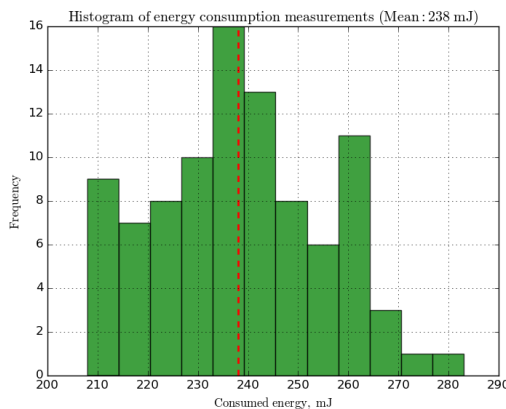
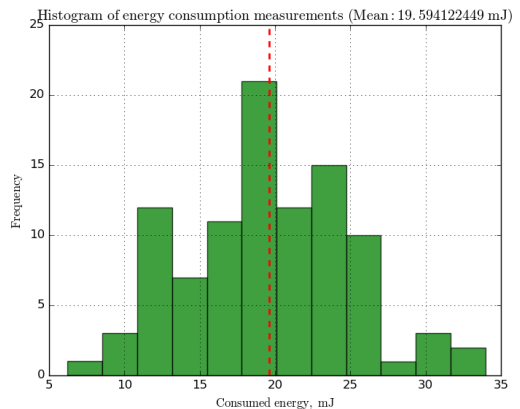
Algoritmas nr.2 pateikiama detali pasiūlyto dalelių filtro lokalizacijos algoritmo modifikacija. Pirmos iteracijos metu pradinis dalelių rinkinys  $S_0$  yra tolygiai paskirstomas 300 metrų spinduliu aplink *a priori* žinomą pradžios tašką ir dalelėms suteikiama pirminė atrankos tikimybė lygi 1. Todėl pirmos iteracijos metu visos dalelės turi vienodą atrinkimo tikimybę. KL divergencijos skaičiavimui naudojamos erdvės, kurios gaunamos paskirsčius žemėlapiu plotą į dvimatį tinklėlį, kurio žingsnis yra 5 metrai. Jeigu bent viena dalelė patenka į žemėlapiu tinklelio kvadratą – ta erdvė laikoma užimta. Pirsono koreliacija naudojama apskaičiuoti vaizdų panašumo mato vertę tarp kameros vaizdo ir žemėlapiu iškarpos. Vaizdų panašumo mato vertė  $R$  perskaičiuojama į dalelės tikimybę  $b^{(i)}$  naudojant konversijos funkciją. Logistinė konversijos funkcija su parametro verte 0,2 yra rekomenduojama bendruoju atveju. Perskaičiuota tikimybė yra proporcinga dalelės išgyvenimo tikimybei kitos iteracijos metu. Pritaikius judesio modelį atrinktos dalelės yra pastumiamos pagal jutiklių pamatuotą judesį. Judesys yra matuojamas apdorojant kameros vaizdus vizualinės odometrijos algoritmu. Ši algoritmo modifikacija pavadinta Dalelių filtro lokalizacija su diskriminavimu pagal Pirsono koreliaciją (angl. *Discriminatory Pearson Correlation based Particle Filter Localization, DPC-PFL*). Parametrizuojamos perskaičiavimo funkcijos leidžia pasiekti dalelių diskriminavimą suteikiant mažesnes arba didesnes dalelių tikimybes, priklausomai nuo Pirsono koreliacijos vertės.

### 3. Eksperimentiniai rezultatai

#### 3.1 Įterptinių skaičiavimo platformų palyginimas

Sobelio filtro algoritmo realizavimas trijose platformose išdėstytas 2.1 skyriuje. Atlikus eksperimentus buvo gautas spartinančių įrenginių greičio ir energetinio efektyvumo palyginimas. Pagal rezultatus nustatyta, kad Parallella platformoje Epiphany spartinantis procesorius atlieka skaičiavimus 2,5 karto lėčiau nei vieno branduolio ARM procesorius. Mali T764 grafinis procesorius skaičiavimus atlieka 2 kartus greičiau nei vieno branduolio ARM procesorius, o NVIDIA Tegra X1 grafinis procesorius atlieka skaičiavimus 15–50 kartų greičiau, priklausomai nuo įvesties nuotraukos dydžio (didėjant įvesties duomenų kiekiui spartinimas didėja).

Pav. 3 pateiktos skaičiavimo platformų sunaudotos energijos histogramos kiekvienai iš 100 iteracijų, apskaičiuojant Sobelio filtrą įvesties nuotraukai, kurios rezoliucija yra 1920x1080 pikselių. Raudona linija žymi vidutinės energijos

(a) *Parallella* su Epiphany-16.(b) *ARM Cortex A9* procesorius.(c) *Radxa Rock2* su ARM Mali T764 GPU.(d) *Airvision Core X1* su Tegra X1 GPU.

Pav. 3: Sunaudotos energijos histogramos kiekvienai iš lyginamų platformų.

snaudų vertes. Pagal rezultatus, Radxa Rock2 platforma su Mali T764 grafiniu procesoriumi sunaudoja 6,85 karto mažiau energijos tiems patiems skaičiavimams nei Parallella platforma su 16 branduolių spartinančiuoju procesoriumi Epiphany-16, o platforma Airvision Core X1 su grafiniu procesoriumi NVIDIA Tegra X1 sunaudojama netgi 84,2 karto mažiau energijos.

### 3.2 Vaizdo panašumo matų palyginimas

Šiame skyriuje pristatomas vaizdo panašumo matų palyginimas. Palyginimas atliekamas naudojant vaizdus iš duomenų rinkinio surinkto bandomojo skrydžio

Lentelė 1: Comparison of image similarity measures.

| Vertė                          | NCC           | CC     | SSD    |
|--------------------------------|---------------|--------|--------|
| Top 5 minimali metrikos vertė  | <b>0.0122</b> | 0.7185 | 0      |
| Top 5 vidutinė metrikos vertė  | <b>0.0550</b> | 0.7471 | 0.0000 |
| Top 5 maksimali metrikos vertė | <b>0.1253</b> | 0.7703 | 0.0000 |
| Top 5 minimumo tikslumas       | <b>2.92%</b>  | 3.21%  | 3.42%  |
| Top 5 vidurkio tikslumas       | <b>6.45%</b>  | 2.70%  | 3.42%  |
| Top 5 maksimumo tikslumas      | <b>25.00%</b> | 6.25%  | 3.42%  |

metu. Lyginami trys matai: Pirsono koreliacija (NCC), kryžminė koreliacija (CC) ir skirtumų kvadratų suma (SSD). Panašumo matas skaičiuojamas tarp nuotraukos iš įrašytos vaizdinės medžiagos skrydžio metu ir žemėlapių iškarpos. Žemėlapių iškarpa koreguojama – atliekami mastelio aukščio pakeitimai, žemėlapis sukamas pagal skrydžio kryptį ir pridamas triukšmas naudojant Gauso ir druskos ir pipirų (angl. *Salt and pepper*) algoritmus. Druskos ir pipirų algoritmas atsitiktinai parenka 5% pikselių ir juos paverčia visiškai baltais ir 5% visiškai juodais. Gauso triukšmas nuotraukai yra pritaikomas sugeneravus atsitiktines pikselių vertes naudojant dispersiją su vidurkiu  $\mu = 25$  ir standartiniu nuokrypiu  $\sigma = 45$  ir gautas vertes pridėjus prie kiekvieno nuotraukos pikselio. Vaizdų panašumo matas skaičiuojamas nespalvotoje gamoje, dėl to spalvoti vaizdai yra paverčiami nespalvotais. Palyginimas atliekamas sugeneruojant 500 hipotetinių pozicijų (dalelių) ir paskaičiuojamas vaizdų panašumo matas su vienu vaizdu. Vaizdų panašumo matai lyginami skaičiuojant tikslumą: kiek dalelių gavo aukščiausius įverčius, kurios buvo arčiausiai realios orlaivio pozicijos. Lentelėje 1 pateikti rezultatai rodo, kad Pirsono koreliacija suteikia daugiausiai aukštų panašumo įverčių hipotezėms esančioms arti realiosios orlaivio pozicijos.

Šioje lentelėje pateiktos Top 5 dalelių (5 dalelių, kurios yra arčiausiai realiosios orlaivio pozicijos) mažiausia, vidutinė ir maksimali panašumo vertės, paskaičiuotos su lyginamais matais. Nors Pirsono koreliacijos (NCC) minimumo tikslumo vertė yra prastesnė nei kitų matų, tačiau vidurkio ir maksimumo tikslumai yra geresni (atitinkamai 2,4 ir 4,0 karto geriau nei artimiausio kryžminės koreliacijos (CC) mato). Dėl šios priežasties Pirsono koreliacija bus naudojama skaičiuojant vaizdų panašumo matą siūlomame dalelių filtro lokalizacijos algoritme.

Pirsono koreliacijos vertė yra priklausoma nuo vaizdų pasukimo, dėl to buvo

įvertinta šio mato paklaida esant netiksliems orlaivio kryptiems matavimams. Įvertinti mato jautrumą krypties paklaidoms buvo panaudota 500 nuotraukų iš bandomojo skrydžio vaizdų ir skaičiuojama koreliacijos vertė sukant žemėlapyje vaizdą aplink centro tašką, o realų vaizdą išlaikant tokį pat, taip imituoju netikslią skrydžio kryptį. Eksperimento rezultatas – koreliacijos vertė kinta vidutiniškai 10% režiuose esant skrydžio krypties paklaidai +/- 2 laipsniai. Tokia paklaidos vertė yra priimtina ir nereikalauja papildomų priemonių mažinant skrydžio krypties paklaidas.

### 3.3 Dalelių atrankos algoritmų palyginimas

Šiuo eksperimentu siekiama identifikuoti greičiausią dalelių atrankos algoritmą, kurio tikslumas nebūtų prastesnis nei įprastų metodų. Eksperimentas atliekamas simuliuotoje aplinkoje, pradėjus skrydį nuo atsitiktinės pozicijos žemėlapyje, aplink pradinę poziciją sugeneruojamos dalelės 312 metrų spinduliu. Atrankos algoritmai vertinami skaičiuojant procentinę išraišką sėkmingų lokalizacijų po 50 dalelių filtro algoritmo iteracijų. Lokalizacija laikoma sėkminga, jei nuspėta pozicija po 50 iteracijų yra ne toliau nei 15 metrų nuo tikrosios pozicijos. 2 lentelėje pateikiami eksperimento rezultatai. Atmetimo ir svarbos algoritmai naudoja fiksuotą dalelių skaičių, kuomet KLD atranka naudoja kintamą dalelių skaičių pagal dalelių pasiskirstymą žemėlapyje. Algoritmo realizacija paleidžiama 100 kartų kompiuteryje su Intel i5-4200M procesoriumi, vykdymo trukmė paskaičiuojama suvidurkinus veikimo laiką. Pagal rezultatus, KLD atrankos algoritmas pasiekia 94% sėkmingų lokalizacijų per 2,6 karto trumpesnę vykdymo laiką.

Lentelė 2: Dalelių atrankos algoritmų palyginimas

| Algoritmas | Galutinis dalelių skaičius | Vidutinė trukmė, s | Sėkmingos lokalizacijos |
|------------|----------------------------|--------------------|-------------------------|
| Atmetimo   | 500                        | 215                | 90 %                    |
| Svarbos    | 500                        | 219                | 99 %                    |
| KLD        | 150                        | 81.8               | 94 %                    |

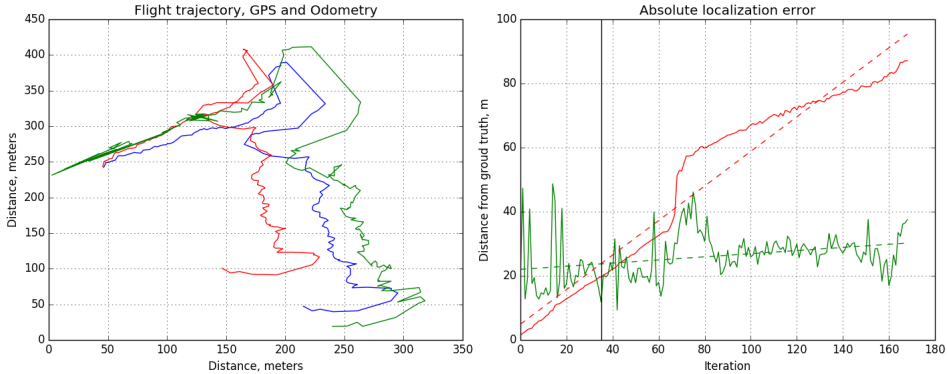
### 3.4 Vaizdo panašumo į tikimybę konversijos funkcijų palyginimas

Šioje disertacijoje pasiūloma nauja vizualinio dalelių filtro lokalizacijos algoritmo komponentė – vaizdų panašumo į tikimybę konversijos funkcija. Siūlomos dvi parametrinės konversijos funkcijos: rektifikuojanti ir logistinė. Šiame eksperimente matuojamas dalelių filtro lokalizacijos algoritmo tikslumas, vykdymo greitis ir atsparumas netikslumams žemėlapyje, naudojant skirtingas konversijos funkcijas su skirtingomis parametro vertėmis. Šiam eksperimentui naudojamas simuliuotas duomenų rinkinys ir kiekvienai konversijos funkcijos ir jos parametro vertės kombinacijai atlikta 120 eksperimentinių skrydžių.

**Tikslumas** matuojamas skaičiuojant Euklidinį atstumą tarp realios orlaivio pozicijos, registruotos duomenų rinkinyje, ir algoritmo prognozuojamos pozicijos. Tiksliausias algoritmas išrinktas naudojant rangavimo metodą. Vidutinis lokalizacijos tikslumas naudojamas suranguoti konversijos funkcijas kiekvienam iš skrydžio scenarijų, geriausią tikslumą pasiekusiai funkcijai suteikiamas 1 taškas, antrai – 2 taškai ir t.t. Funkcija surinkusi mažiausiai taškų yra laikoma tiksliausia. Pagal gautus rezultatus logistinė funkcija su parametro verte 0,2 pasiekė geriausius rezultatus, logistinė funkcija pagerino lokalizacijos tikslumą 43% lyginant su Softmax funkcija.

**Greitis** matuojamas vidutiniu įvertintų dalelių skaičiumi (algoritmo iteracijų) per vieną eksperimentinį skrydį. Vidutinis įvertintų dalelių skaičius yra tiesiogiai proporcingas algoritmo veikimo laikui. Matuojant vidutinį įvertinimų skaičių atmetamas programos vykdymo stochastiškumas ir matavimas tampa nepriklausomas nuo techninės įrangos. Softmax konversijos funkcija lokalizacijos veikimą pagreitino vidutiniškai 7% lyginant su tiesine normalizacija. Geriausiai suranguota logistinė konversijos funkcija su parametro verte 0,2 pagreitino lokalizaciją 3,18 karto lyginant su tiesinės normalizacijos funkcija ir 2,96 karto lyginant su Softmax funkcija.

**Atsparumas** netikslumams žemėlapyje vertinamas pagal lokalizacijos tikslumo standartinių nuokrypį skirtingo amžiaus žemėlapiuose. USGS pateikia tų pačių regionų žemėlapius kartą per 2 metus, nuo 2008 iki 2014 metų. Naudojantis šiais žemėlapiais galima palyginti kaip kinta lokalizacijos tikslumas senstant žemėlapiams. Rektifikuojančios funkcijos su parametro vertėmis 0,2, 0,1 ir 0 ir logistinės funkcijos su parametro vertėmis 0,7 ir 0,4 reliatyvus standartinis



(a) Algoritmų atkurtos orlaivio skrydžio trajektorijos.

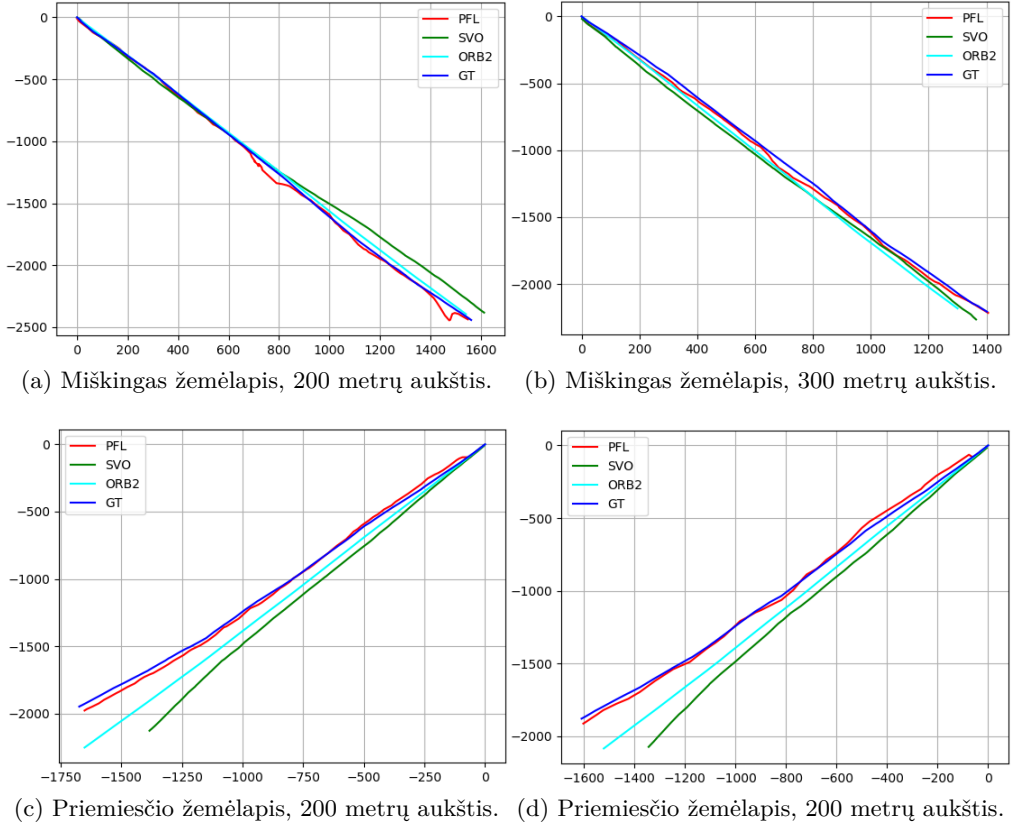
(b) Pozicijos paklaida.

Pav. 4: Pasiūlyto dalelių filtro algoritmo DCP-PFL (žalia) palyginimas su vizualine odometrija SVO (raudona). Atkurtos skrydžio trajektorijos ir lokalizacijos paklaidų grafikai skaičiuojant nuo pozicijos gautos naudojant GPS (mėlyna).

nuokrypis naudojant bet kokio amžiaus žemėlapius neviršija 10%. Šios funkcijos leidžia lokalizuoti orlaivį su labai panašia paklaida nepaisant netikslumų esančių senesniuose žemėlapiuose. Kitų funkcijų tikslumas ženkliai prastėja naudojant senesnius žemėlapius, tačiau pasiekia geresnį tikslumą naujuose žemėlapiuose.

### 3.5 Palyginimas su pažangiausiais algoritmais

Pasiūlytas dalelių filtro algoritmas palyginamas su pažangiausiu ORB-SLAM algoritmu naudojant simuliuotų skrydžių duomenų rinkinį. Eksperimentai atlikti tik scenarijuose, kuriuose orlaivis skrenda tiesios linijos trajektorija. 5a, 5b, 5c ir 5d pav. pateiktos atkurtos skrydžių trajektorijos naudojant ORB-SLAM, SVO ir pasiūlytą algoritmą; realioji orlaivio pozicija gauta iš simuliuotų GPS duomenų. 3 lentelėje pateiktos lokalizacijos paklaidų vertės kiekvienam iš skrydžio scenarijų. Pagal šiuos rezultatus ORB-SLAM ir SVO tikslumas yra labai panašus (vidutiniškai ORB-SLAM yra 9% tikslesnis už SVO), tačiau pasiūlytas algoritmas yra 2,6 karto tikslesnis nei ORB-SLAM ir 3,1 karto tikslesnis nei SVO algoritmas.



Pav. 5: Skrydžio trajektorijos atkurtos naudojant SVO, ORB-SLAM ir pasiūlytu dalelių filtro algoritmu (PFL).

Lentelė 3: Lokalizacijos tikslumo palyginimas tarp SVO, ORB-SLAM ir pasiūlyto algoritmo (PFL).

| Scenario | Paklaida, m |       |        | Reliatyvi paklaida |         |         |
|----------|-------------|-------|--------|--------------------|---------|---------|
|          | SVO         | PFL   | ORB    | PFL/SVO            | ORB/SVO | PFL/ORB |
| FL-200   | 48.36       | 27.60 | 49.56  | 75.26%             | -2.43%  | 79.61%  |
| FL-300   | 45.00       | 17.08 | 58.55  | 163.53%            | -23.14% | 242.88% |
| UL-200   | 130.95      | 21.93 | 118.48 | 497.03%            | 10.53%  | 440.17% |
| UL-300   | 129.88      | 21.06 | 86.05  | 516.84%            | 50.94%  | 308.67% |
| Average  |             |       |        | 313.16%            | 8.97%   | 267.83% |

## 4. Bendrosios išvados

1. Pirsono koreliacijos koeficientas (arba normalizuotas koreliacijos koeficientas) yra tinkamiausias vaizdų panašumo matas bepiločio orlaivio lokalizacijai, nedidelio aukščio skrydžiuose ir nenaudojant GPS signalo, lyginant su kvadratų skirtumų suma ir kryžmine koreliacija. Ji suteikia aukštesnius panašumo įverčius hipotezėms, kurios yra arčiau realios orlaivio pozicijos, 4 kartus dažniau lyginant su kryžmine koreliacija.
2. Pasiūlytas dalelių filtro lokalizacijos algoritmas, realizuotas KLD atrankos pagrindu, veikia 2,6 karto greičiau nei lokalizacijos algoritmai, pagrįsti atmetimo ar svarbos atrankų metodais.
3. Dalelių filtro lokalizacijos algoritmas, naudojantis logistinę vaizdų panašumo į tikimybę konversijos funkciją su parametro verte 0,2, pagerina lokalizacijos tikslumą 43% palyginus su Softmax konversijos funkcija.
4. Pasiūlytas dalelių filtro lokalizacijos algoritmas 2 kartus pagerina lokalizacijos tikslumą lyginant su pažangiausiu vizualinės odometrijos algoritmu – SVO. Dalelių filtro lokalizacijos algoritmas sumažina besikaupiančias odometrijos paklaidas iki 11 kartų.
5. Pasiūlytas dalelių filtro lokalizacijos algoritmas (naudojantis KLD atranką, logistinę konversijos funkciją su parametro verte 0,2 ir Pirsono koreliaciją) pagerina lokalizacijos tikslumą 2,6 karto lyginant su pažangiausiu vizualiniu SLAM algoritmu – ORB-SLAM.
6. Pasiūlytas įterptinių skaičiavimo platformų energetinio efektyvumo palyginimo metodas identifikavo, kad Airvision Core X1 su NVIDIA Tegra X1 grafiniu procesoriumi yra energetiškai efektyviausia platforma kompiuterinės regos algoritmų vykdymui orlaivyje. Ši platforma yra 142 kartus greitesnė ir 84 kartus energetiškai efektyvesnė nei vieno branduolio ARM procesorius ir 29 kartus greitesnė ir 12 kartų energetiškai efektyvesnė nei artimiausia lygiagrečių skaičiavimų platforma Radxa Rock 2 su ARM Mali T764 grafiniu procesoriumi.



# **Investigation of Particle Filter Based Visual Localization for Unmanned Aerial Vehicle Flights at Low-Altitude**

## **Research Context**

During the recent years, the development of powerful embedded computing platforms enables the use of computer vision and artificial intelligence onboard robotic platforms, such as Unmanned Air Vehicles (abbr. UAV) (e.g. drones), Unmanned Surface Vehicles (abbr. USV), Unmanned Underwater Vehicles (abbr. UUV) and others. These capabilities take a step forward towards intelligent, autonomous, secure, and most importantly safe robots. In the case of UAVs, flight safety and security are important issues, since UAVs are highly dependent on GPS signal which can be jammed or spoofed. The UAVs play an important role in the military, law enforcement, and rescue services by providing critical intelligence from above. Conventional autopilot systems fail to navigate safely if the GPS signal is lost or the UAV navigates in a GPS-denied environment, and there is no available alternative localization method. UAV should estimate its own position without the need of GPS signal or other radio signal based localization techniques. Therefore, visual localization, the process of pose estimation relatively to a known environment using optical sensors, may solve the problem of navigation in GPS-denied scenarios. Visual odometry calculated on imagery from a downward looking camera on a UAV can solve the GPS-Denied localization. Such a solution may deal with this problem to certain limitations: the visible area of the camera must contain enough visual features for tracking throughout the flight. Visual odometry has a fundamental issue of positioning drift and errors add up to infinity, within infinite flight time.

## **Statement Of The Problem**

Autonomous flight of UAV is highly dependent on precise pose estimation of the aircraft body. Consequently, flights are impossible to perform or fail if the GPS signal is lost or jammed. This is an especially worrying problem for military UAVs since the GPS signal is completely unavailable in such scenarios. The field of computer vision is tackling the problem by proposing visual localization algorithms, that process video stream from onboard cameras and estimates aircraft

pose in space. Recently developed algorithms – visual odometry, SLAM, optical flow, and image registration – are able to provide backup localization for UAVs. Although these techniques show stunning results in indoor or outdoor environments at near-ground ( $< 100$  meters) altitude, not many researchers apply these algorithms in low altitude flights (between 100-3000 meters). Most researchers focus on micro UAVs, that are only capable of flying at near-ground altitudes or indoor environments since they are cheaper and more widely available.

Problems faced at near-ground altitude is quite different than during low altitude, long distance flights that are far more expensive to perform. To provide precise and drift-free positioning for low altitude UAV flights, this thesis analyses map relative localization techniques using an optical sensor. Improvements to current techniques also proposed to speed-up execution, reduce localization drift, improve precision and robustness to inaccuracies in maps, therefore, localization could be feasible in outdated maps.

## **Research Object**

The object of this research is visual localization of a UAV using computer vision, image processing, and Particle filtering algorithms, that suitable for execution onboard UAV with embedded computer hardware.

## **Research Aim And Objectives**

The research aim is to develop new visual localization algorithm, that is capable of UAV pose estimation in GPS-Denied environments during low altitude and long-distance flights, running on an embedded-flight computer that outperforms existing methods in localization accuracy, execution speed, and robustness to errors in the aerial imagery.

The following tasks were performed to accomplish the research aim:

1. Compare image similarity measures to identify the most suitable measure for matching terrain images captured during a low altitude flight for UAV localization.
2. Compare particle sampling techniques used in Particle filters to identify which sampling technique which requires the least computations for Particle filter localization algorithm.
3. Propose and compare image similarity to particle probability conversion function to improve the accuracy of a Particle filter localization algorithm.

4. Propose a new visual Particle filter localization algorithm based on the results of previous tasks and evaluate the ability of the algorithm to reduce the accumulating error of Visual odometry.
5. Compare the proposed Particle filter localization algorithm against state-of-the-art SLAM algorithm.
6. Propose a new method for evaluation of embedded computing platforms to identify the most energy efficient platform currently available that suitable for computer vision tasks onboard UAV.

## **Research Methods**

1. Literature review is performed on the latest scientific papers to identify, select and evaluate state-of-the-art algorithms solving the stated problem.
2. Quantitative and qualitative information gathering was performed to create datasets used for experiments and experimental data describing the performance of the proposed solution or its components.
3. Statistical methods, e.g. Student's t-test, Shapiro-Wilk test, are used to perform confirmatory data analysis, ensuring the reliability of data and experimental setup.
4. Constructive research was used to propose improvements in the real-world of the problem and propose new methods to improve the theory.
5. Software development methods were used in the experimental part of this thesis, implementing localization algorithms and proposing new methods to improve existing algorithms.

## **Scientific Contributions Of The Research**

1. A new algorithm of visual localization based on Particle filter proposed. A new component for Particle filter algorithm introduced — image similarity to the probability conversion function. The effects of the function on accuracy, execution speed and robustness were measured in the experimental section of this thesis. The ability to achieve trade-off of accuracy for robustness by using function parametrization presented.
2. Proposed Particle filter localization algorithm was able to outperform state-of-the-art algorithms — SVO and ORB-SLAM in the means of localization accuracy.

3. A benchmark method for comparison of embedded computing platforms is proposed to identify the most energy efficient platform capable of common computer vision tasks. Three platforms compared, and a platform based on NVIDIA TX1 GPU identified as the fastest and most energy efficient platform for image processing onboard UAV.

## **Practical Value Of The Research**

1. New localization algorithm based on Particle filter was developed and its source code is publicly available for other researchers to benchmark against and use in the development of UAV backup localization systems.
2. The experimental results have shown that by using image similarity to probability density conversion function in an adaptive Particle filter localization algorithm, it is possible to control the accuracy and robustness of the localization.
3. A new dataset of aerial imagery was created and published with open access. The dataset contains images from a simulated alongside with sensor data that can be used to develop and evaluate optical flow, visual odometry, localization, SLAM, and other vision algorithms. To the best of our knowledge, it is by far the largest dataset of low altitude aerial imagery.
4. Different particle sampling techniques were evaluated and KLD sampling technique showed significant improvement in localization algorithm execution runtime with similar accuracy compared to other techniques.
5. The new method of embedded computing platform comparison was used to identify the Nvidia Tegra X1 as the most energy-efficient computing platform from currently most advanced available platforms.

## **Defensive Claims**

1. The use of logistic conversion function allows the proposed localization algorithm to achieve higher accuracy and improve robustness against inaccuracies in maps of the environment.
2. Proposed localization algorithm based on Particle filter is more accurate than Visual odometry and has reduced the accumulating error.
3. Proposed localization algorithm based on Particle filter is more accurate than state-of-the-art algorithm ORB-SLAM.

4. Pearson Correlation Coefficient is the most suitable image similarity measure for visual localization of a UAV compared against the sum of squared differences and cross-correlation measures.
5. KLD sampling algorithm speeds-up the execution of Particle filter based visual localization by reducing the number of particles used for localization without affecting positioning accuracy.
6. The proposed method for energy efficiency measurement allows the comparison of embedded computing platforms and identifies the most energy efficient platform.

## General Conclusions

1. Pearson Correlation Coefficient (or Normalized Correlation Coefficient) is the most suitable image similarity measure for localization in low altitude GPS-Denied aerial flight, compared to Sum of Squared Differences and Cross-Correlation. It provides higher similarity values to hypotheses that are closer to the ground truth location compared to the other image similarity measures. The usage of Normalized Correlation Coefficient improves the detection probability of a particle that is close to the ground truth 4 times compared to Cross-Correlation.
2. Particle filter localization algorithm using KLD sampling technique performs 2.6 times faster compared to rejection and importance sampling-based Particle filter localizations with a similar localization success rate.
3. Particle filter based localization algorithm using logistic conversion function with parameter value 0.2 achieved 43% more accurate localization and is 3 times faster compared to the same algorithm using baseline Softmax conversion function.
4. The Proposed particle filter localization algorithm reduces the localization error 2 times compared to state-of-the-art Visual odometry algorithm SVO. Additionally, the algorithm reduces the accumulating error slope of odometry 11 times.
5. Proposed localization algorithm based on Particle filter, KLD sampling, logistic conversion function with parameter value 0.2 and Normalized Cross Correlation achieves 2.6 times better localization accuracy compared against state-of-the-art Visual SLAM ORB-SLAM algorithm.
6. The proposed method for measuring the energy efficiency of computing platforms identified the *Airvision Core X1* with *NVIDIA Tegra X1* GPU as the

most energy efficient platform for computing onboard UAV. It is ~142x times faster and ~84x more energy efficient than single-core ARM CPU and ~29x times faster and ~12x more energy efficient than second best platform *Radxa Rock 2* with *ARM Mali T764* GPU.

## 5 Doktoranto publikacijos disertacijos tema

### Publikacijos periodiniuose leidiniuose

1. Jurevičius, R., Marcinkevičius, V., and Šeibokas J. (2018). Robust GPS Denied Localization for UAV Using Particle Filter and Visual Odometry. Machine Vision and Applications, Springer. ISSN: 1432-1769 [Under Review] [IF: 1.306]
2. Jurevičius. R. (2018). AIR: A Dataset Of Aerial Imagery From Robotics Simulator. Robotics And Automation Letters, IEEE. ISSN: 2377-3766 [Under Review]
3. Jurevičius, R., and Marcinkevicius, V. (2016). Energy Efficient Platform for Sobel Filter in Energy and Size Constrained Systems. Baltic Journal of Modern Computing, 4(1), 2015, p. 79-88. ISSN: 2255-8950

### Publikacijos recenzuojamuose konferencijų leidiniuose

4. Jurevičius, R., and Marcinkevičius, V. (2017). Application Of Vision-Based Particle Filter and Visual Odometry for UAV Localization. WSCG '2017: short communications proceedings: The 25th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision 2016 in co-operation with EUROGRAPHICS: University of West Bohemia, Plzen, Czech Republic May 29 – June 2 2017, p. 67–71. ISBN: 978-80-86943-45-9
5. Jurevičius, R., Marcinkevičius, V., and Taujanskas, V. (2016). Comparison of Image Similarity Functions and Sampling Algorithms in Vision-Based Particle Filter for UAV Localization. Proceedings of CSIST 2016, p. 109–114. ISBN: 978-985-566-369-1
6. Jurevičius, R., and Marcinkevičius, V. (2015). Energy Efficient Platform for Sobel Filter Implementation in Energy and Size Constrained Systems. In Information, Electronic and Electrical Engineering (AIEEE), 2015 IEEE 3rd Workshop on Advances in (pp. 1–5). IEEE. ISBN: 978-1-5090-1201-5

Vilniaus universiteto leidykla  
Universiteto g. 1, LT-01513 Vilnius  
El. p. [info@leidykla.vu.lt](mailto:info@leidykla.vu.lt),  
[www.leidykla.vu.lt](http://www.leidykla.vu.lt)  
Tiražas XX egz.