



**Vilnius  
universitetas**

**Vilnius  
universitetas**

# **Autonominis nežinomos patalpų aplinkos-terpės išžvalgymas mažais bepiločiais orlaiviais**

Mantas Briliauskas  
Vilnius, 2024  
II metai II pusmetis

# Doktorantūra

Doktorantas: Mantas Briliauskas

**Tema:** Autonominis nežinomos patalpų aplinkos-terpės išžvalgymas mažais bepiločiais orlaiviais

**Darbo vadovas:** prof. dr. Virginijus Marcinkevičius

**Doktorantūros pradžios ir pabaigos metai:** 2022 – 2026

**Studijų metai:** 2

# Studijų planas

Vilniaus  
universitetas

Studijų metai	Egzaminai			Dalyvavimas konferencijose				Publikacijos					
	Planas	Įvykdyta	Būklė	Tarptautinėse		Nacionalinėse		Su citav. rodikliu			Be citav. rodiklio		
				Planas	Įvykd.	Planas	Įvykd.	Planas	Įvykd.	Būklė	Planas	Įvykd.	Būklė
I (2022/2023)	1	2	Išlaikyta, 10 / Išlaikyta, 9	0	0	1	2	0	0				
II (2023/2024)	2	2	Išlaikyta, 9 / Išlaikyta, 10	0	0	1	0	0	1	išspausdinta			
III (2024/2025)	1			1				1					
IV (2025/2026)	0			1				1					

# Ataskaitinis studijų pusmetis

Vilniaus  
universitetas

2023/2024 (II pusmetis)	Planas	Įvykdyta	Pastaba
Egzaminai	Gilieji neuroniniai tinklai	Gilieji neuroniniai tinklai, 2024 birželis	Išlaikyta
Dalyvavimas konferencijose tarptautinėse	1	ECML PKDD '24, 2024 m. rugsėjo 9–13 d., Vilnius	Sudalyvauta
Dalyvavimas vasaros stovyklose doktorantų	1	Vilniaus universiteto doktorantų vasaros stovykla, 2024 m. birželio 17–21 d. Vilnius, botanikos sodas	Sudalyvauta
Publikacijos	-	-	-

# Mokslinių tyrimų ir disertacijos rengimo etapai

<p><b>2.2. Teorinis tyrimas:</b></p> <ol style="list-style-type: none"><li>1. Vienalaikės lokalizacijos ir žemėlapių sudarymo metodų analizė ir tyrimas.</li><li>2. Navigacijos metodų analizė ir tyrimas.</li><li>3. Autonominio vietovės išžvalgymo metodų analizė ir tyrimas.</li></ol>	<p>2024 m. kovo mėn. – 2024 m. gegužės mėn.</p>	<ol style="list-style-type: none"><li>1. Vienalaikės lokalizacijos ir žemėlapių sudarymo metodų panaudojant skatinamąjį mokymą tyrimas.</li><li>2. Simuliacinės aplinkos tyrimui parinkimas.</li></ol>
<p><b>2.3. Empirinis tyrimas:</b></p> <ol style="list-style-type: none"><li>1. Skirtingų algoritmų palyginimas.</li></ol>	<p>2024 m. birželio mėn. – 2024 m. rugsėjo mėn.</p>	<ol style="list-style-type: none"><li>1. Simuliacinės aplinkos tyrimui paruošimas. Techninės įrangos paruošimas. Pradėtas, bet nebaigtas, programinės derinimas.</li></ol>
<ol style="list-style-type: none"><li>2. Įgyvendintų algoritmų modifikacijos, ar naujų algoritmų kūrimas, sprendžiant apibrėžtus uždavinius.</li><li>3. Sukurtų modifikacijų eksperimentinis tyrimas analizuojant jų efektyvumą.</li></ol>	<p>2024 m. spalio mėn. – 2025 m. vasario mėn.</p>	

# Tyrimo objektas, tikslas ir uždaviniai

## Objektas:

- autonominio nežinomos patalpų aplinkos-terpės išžvalgymo algoritmai mažiems bepiločiams orlaiviams panaudojant skatinamąjį mokymą.

## Tikslas:

- sukurti efektyvų algoritmą autonominiam nežinomos patalpų aplinkos-terpės išžvalgymui skirtą mažiems bepiločiams orlaiviams panaudojant skatinamąjį mokymą.

## Uždaviniai:

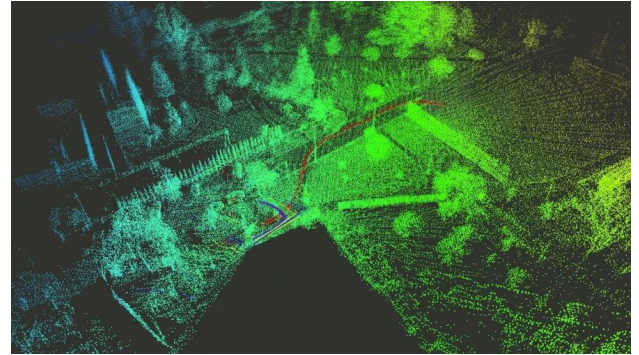
- pažangiausių autonominio nežinomos patalpų aplinkos-terpės išžvalgymo algoritmų panaudojant skatinamąjį mokymą apžvalga ir palyginimas;
- skatinamojo mokymo taikymas mažo bepiločio orlaivio stabilizavimui mažame aukštyje;
- pažangiausio algoritmo modifikavimas sukuriant naują algoritmą;
- sukurto algoritmo palyginimas su pažangiausiu algoritmu.

# SLAM

Angl. *Simultaneous Localization and Mapping*

Tikslas - robotui judant erdvėje ir panaudojant iš sensorių gautą informaciją sudaryti aplinkos žemėlapij:

1. nuskaityta sensorių informacija;
2. įvertinamas judesio modelis;
3. atliekamas pozicijos filtravimas:
  - a. KF, EKF, dalelių filtrai (*particle filters*);
4. papildomas aplinkos žemėlapis.



# Active SLAM

Vilniaus  
universitetas

Active SLAM = SLAM + automatinis kito navigacijos tikslo parinkimas.

POMDP problema.

Duota: drono ir orientyrų pozicijos.

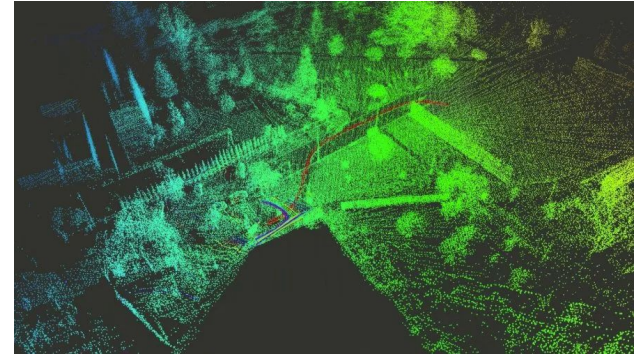
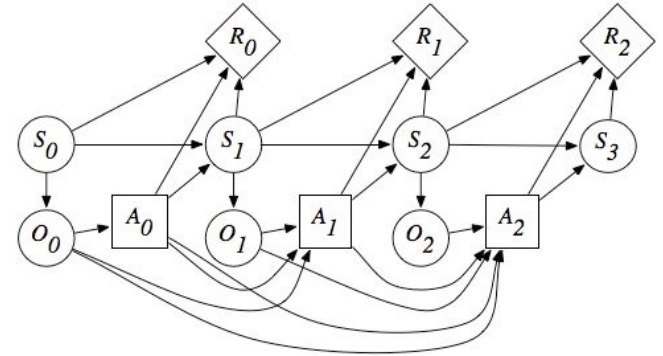
Active SLAM:

1. navigacijos tikslų-kandidatų parinkimas;
2. navigacijos tikslų-kandidatų įvertinimas;
3. navigacijos tikslo atrinkimas;
4. navigacija su SLAM.

Iki:

1. išvalgyta apibrėžta aplinkos dalis;
2. žemėlapių neapibrėžtumas (*uncertainty*) žemiau užsibrėžtos ribos;
3. baigiasi užduočiai skirtas laikas ( $t > T$ ).

Išvalgymo-eksplotacijos (angl. *exploration-exploitation*) problema.







**Vilniaus  
universitetas**

**Vilniaus  
universitetas**

# **Atlikta: taškų debesies segmentavimas su DNN**

# Tyrimas dalykui “Giliaji neuroniniai tinklai”

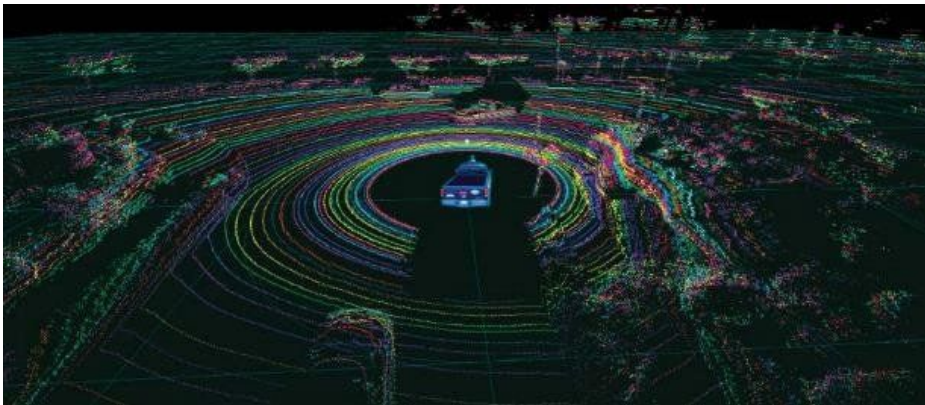
- Dalyko vadovas: prof. dr. Povilas Treigys
- Tema: “Statinių/dinaminių objektų segmentavimas 3D taškų debesyse giliaisiais neuroniniais tinklais”
- Uždaviniai:
  - apžvelgti literatūrą;
  - pasirinkti algoritmą;
  - jį įgyvendinti;
  - ištirti tikslumą statinėms/dinaminės kliūtims atpažinti (pvz. sienos, ir patalpoje judantys objektai).



# Sensoriaus judesio mokymasis be mokytojo

Angl. *self-supervision for ego-motion*:

- srauto pokyčių aptikimas dėl sensoriaus judesio;
- statiniai taškai *pasaulio koordinačių sistemoje*;
- Kabsch tapatinimas \*\*.



Rotation  
+  
Translation



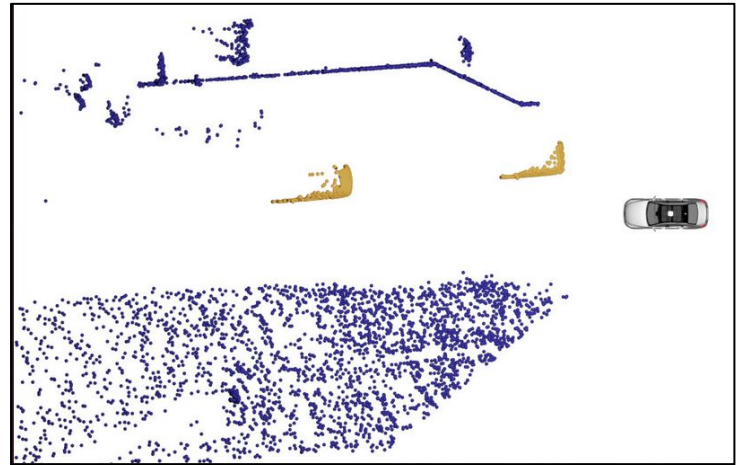
\*\*

\* Hirz, Mario & Walzel, Bernhard. (2018). Sensor and object recognition technologies for self-driving cars. Computer-Aided Design and Applications. 15. 1-8. 10.1080/16864360.2017.1419638.

\*\* [https://www.youtube.com/watch?v=nCs\\_e6fP7Jo](https://www.youtube.com/watch?v=nCs_e6fP7Jo)

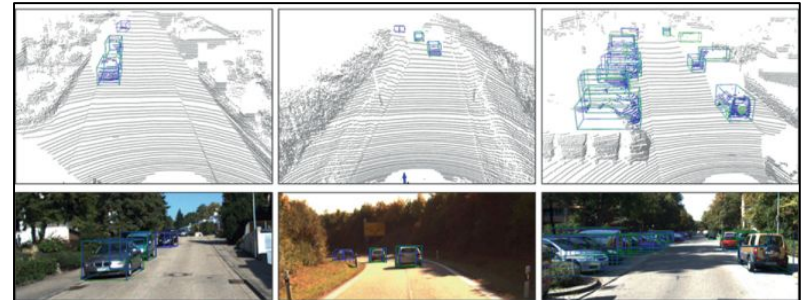
# SLIM metodas

- Scenos srautas  $F_t$  išmokstamas RAFT bloke.
- Taškų klasifikacija:
  - statiniai taškai: Kabsch tapatumas:
    - $T_t$  matricos paieška;
    - $l_2$  paklaidos mažinimas  $P_{t+1} - T_t P_t$ .
  - dinaminiai taškai:
    - $l_2$  paklaidos mažinimas  $P_{t+1} - (P_t + F_t)$ .



# SLIM įgyvendinimas ir testavimas

- kodas: [https://github.com/mercedes-benz/selfsupervised\\_flow](https://github.com/mercedes-benz/selfsupervised_flow)
- NN karkasas: Tensorflow;
- mokymo įrenginys: GPU;
- dalis PointPillar ir Kabsch įgyvendinti C++;
- mokymo resursai: VU HPC, GPU resursai;
- duomenų rinkinys: nuScenes taškų debesų srautas;
  - KITTY: sutapatinti taškai;
- ~35000 mokymo iteracijų (~4 dienos).



# Rezultatai

- Dinaminiai taškai
  - AEE **0.0954** (“average Euclidean error”)
  - AccR **0.7623** (“point ratio where EE < 0.1m”)
  - ROutl **0.0623** (“point ratio where EE > 0.3m”)
- Statiniai taškai
  - AEE **0.1031**
- 50-50 (vidurkis)
  - AEE **0.0993**

Table 3: Self-supervised training & evaluation on nuScenes

	Moving			Stat.	50-50
	AEE↓	AccR↑	ROutl↓	AEE↓	AEE↓
Zero	0.6381	0.1632	0.5783	0.5248	0.5814
PPWC	0.3539	0.2543	0.3848	0.1974	0.2756
PF	0.7399	0.0000	0.9364	<b>0.0570</b>	0.3985
Ours	<b>0.1050</b>	<b>0.7365</b>	<b>0.0240</b>	0.0925	<b>0.0987</b>

# Algoritmo tobulinimo galimybės

- Pritaikytas 2.5D, ne 3D
  - Autoriai nerekomenduoja naudoti su FlyingThings3D duomenų rinkiniu
  - Tobulinti arba pakeisti PointPillars 3D pritaikytu enkoderiu
- Neišnaudotas duomenų srauto potencialas
  - Taškų dinamikos nustatymas
    - $P_{t+1} = P_t + F_t$
    - $P_{t+2} = P_{t+1} + F_{t+1} = P_{t+1} = P_t + F_t + F_{t+1}$
    - $P_t = P_0 + \sum_i F_i + \epsilon$
  - Pritaikyti RNN visame SLIM



**Vilnius  
universitetas**

**Vilnius  
universitetas**

**Dabar: simulatorius**



# Simuliacinė aplinka

## *Unreal Engine 5:*

- atkuriamą tikrovę atitinkanti grafika;
- žemėlapių kūrimo įrankis *Unreal Editor*;
- “fizikos be grafinio atvaizdavimo” režimas.

## *AirSim (Colosseum):*

- įskiepis kvadrotoriaus skrydžio simuliacijai;
- RGB, RGBD kameros;
- biblioteka kvadrotoriaus valdymui iš python kodo.



\* <https://paperswithcode.com/dataset/airsim>

\*\* <https://github.com/Sollimann/airsim-client>

# Simuliacinės aplinkos panaudojimas

Sukurtas python modulis *AirSimDrone*:

- supaprastintas drono valdymas;
- RGB kameros vaizdas;
- RGBD kameros vaizdas.

Liko pridėti:

- ORB-SLAM3\*\*\* palaikymą.



\* <https://paperswithcode.com/dataset/airsim>

\*\* <https://github.com/Sollimann/airsim-client>

\*\*\* [https://github.com/UZ-SLAMLab/ORB\\_SLAM3](https://github.com/UZ-SLAMLab/ORB_SLAM3)



**Vilnius  
universitetas**

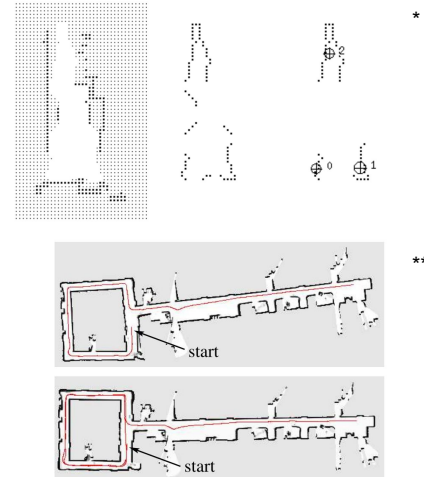
**Vilnius  
universitetas**

**Toliau: žvalgybos algoritmų įgyvendinimas**

# Žvalgybos algoritmai įgyvendinimui

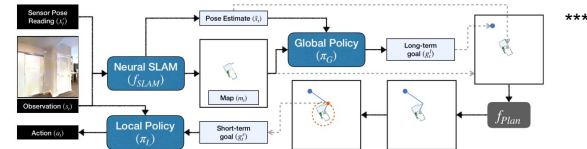
Klasikiniai algoritmai:

1. pasienio (angl. *frontier*) žvalgyba<sup>\*</sup>;
2. entropijos mažinimas žvalgybos-panaudos (angl. *exploration-exploitation*) metodu<sup>\*\*</sup>;
3. žvalgyba hierarchiniu skatinamuoju mokymu<sup>\*\*\*</sup>.



Naujausi pasiekimai:

4. smalsumu (angl. *curiosity*) pagrįsta žvalgyba<sup>\*\*\*\*</sup>.



\* B. Yamauchi, "A frontier-based approach for autonomous exploration", Proceedings 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation CIRA'97. 1997, pp. 146-151, doi: 10.1109/CIRA.1997.613851.

\*\* C. Stachniss, "Exploration with active loop-closing for FastSLAM," 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Japan. 2004, pp. 1505-1510 vol.2, doi: 10.1109/IROS.2004.1389609.

\*\*\* S. C. Devendra et al., "Learning to Explore using Active Neural SLAM", Proceeding 2020 ICLR. 2020, doi: 10.48550/arXiv.2004.05155.

\*\*\*\* N. Botteghi et al., "Curiosity-Driven Reinforcement Learning Agent For Mapping Unknown Indoor Environments", ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. 2021, V-1-2021, 129–136, doi: 10.5194/isprs-annals-V-1-2021-129-2021.

# Ačiū už dėmesį.

Darbo autorius dėkoja Vilniaus universiteto Matematikos ir informatikos fakulteto Informacinių technologijų atviros prieigos centrui už suteiktus HPC išteklius šio darbo skaičiavimams atlikti.