



Kauno technologijos universitetas

Elektros ir Elektronikos fakultetas

Mikromobilumo priemonių pavojingų manevrų identifikavimo metodų tyrimas

Baigiamasis magistro projektas

Dominykas Simpukas

Projekto autorius

Doc. Pranas Kuzas

Vadovas

Kaunas, 2024



Kauno technologijos universitetas

Elektros ir elektronikos fakultetas

Mikromobilumo priemonių pavojingų manevrų identifikavimo metodų tyrimas

Baigiamasis magistro projektas (6211EX012)

Dominykas Simpukas

Projekto autorius

Doc. Pranas Kuzas

Vadovas

Doc. Šarūnas Kilius

Recenzentas

Kaunas, 2024



Kauno technologijos universitetas

Elektros ir elektronikos fakultetas

Dominykas Simpukas

Mikromobilumo priemonių pavojingų manevrų identifikavimo metodų tyrimas

Akademinio sąžiningumo deklaracija

Patvirtinu, kad:

1. baigiamąjį projektą parengiau savarankiškai ir sąžiningai, nepažeisdamas kitų asmenų autoriaus ar kitų teisių, laikydamasis Lietuvos Respublikos autorių teisių ir gretutinių teisių įstatymo nuostatų, Kauno technologijos universiteto (toliau – Universitetas) intelektinės nuosavybės valdymo ir perdavimo nuostatų bei Universiteto akademinės etikos kodekse nustatytų etikos reikalavimų;
2. baigiamajame projekte visi pateikti duomenys ir tyrimų rezultatai yra teisingi ir gauti teisėtai, nei viena šio projekto dalis nėra plagijuota nuo jokių spausdintinių ar elektroninių šaltinių, visos baigiamojo projekto tekste pateiktos citatos ir nuorodos yra nurodytos literatūros sąrašė;
3. įstatymų nenumatytų piniginių sumų už baigiamąjį projektą ar jo dalis niekam nesu mokėjęs;
4. suprantu, kad išaiškėjus nesąžiningumo ar kitų asmenų teisių pažeidimo faktui, man bus taikomos akademinės nuobaudos pagal Universitete galiojančią tvarką ir būsiu pašalintas iš Universiteto, o baigiamasis projektas gali būti pateiktas Akademinės etikos ir procedūrų kontrolieriaus tarnybai nagrinėjant galimą akademinės etikos pažeidimą.

Dominykas Simpukas

Patvirtinta elektroniniu būdu

Dominykas Simpukas. Mikromobilumo priemonių manevrų identifikavimo metodų tyrimas. Magistro baigiamasis projektas / vadovas doc. Pranas Kuzas; Kauno technologijos universitetas, Elektros ir elektronikos fakultetas.

Studijų kryptis ir sritis (studijų krypčių grupė): Elektronikos inžinerija, inžinerijos mokslai.

Reikšminiai žodžiai: Mikromobilumas, pavojingas manevras, signalų apdorojimas, požymis.

Kaunas, 2024. 74 p.

Santrauka

Baigiamajame darbe tiriama įvairūs mikromobilumo priemonės pavojingi manevrai, galintys sužaloti vairuotoją ar kitus eismo dalyvius. Kuriamą sistemą, gebanti surinkti įvairių jutiklių duomenis, išbandomi keli metodai skirti aptikti tam tikrus manevrus iš gautų jutiklių duomenų, juos atpažinti ir nustatyti pavojingumo lygį. Darbo tikslas – ištirti dažniausiai sutinkamus mikromobilumo priemonės pavojingus ar neatsargius manevrus, juos identifikuoti, jutiklių pagalba juos įrašyti ir taikant kelis algoritmus juos atpažinti, klasifikuoti ir palyginti tarpusavyje. Algoritmai turėtų būti nereikalaujantys didelių našumo resursų, tinkami taikyti žemadažnių signalų atpažinimui, o skaičiavimai atliekami kaip galima greičiau.

Pirmoje dalyje analizuojami dažniausiai naudojami jutikliai mobilumo uždaviniuose, aprašomi jų veikimo principai ir matuojamas dydis. Taip pat analizuojami literatūros šaltiniai ir atlikti tyrimai, susiję su objekto judesiu ir įvairių manevrų atpažinimu. Galiausiai apžvelgiama įmonės, kurios gamina išmaniuosius paspirtukus su įdiegtomis saugumo savybėmis, teikiama informacija.

Antroje dalyje aptariami du naudoti duomenų surinkimo metodai - išmaniuoju telefonu ir sukurta įterptinė sistema. Pradžioje aprašoma mobiliojo telefono eksperimento stendas, skirtas surinkti akcelerometro duomenis, atliekamas gautų duomenų atvaizdavimas ir pirminė analizė, identifikuojami šio stendo pliusai ir minusai, vėliau pereinama prie specializuoto stendo kūrimo, kuris išsprendė pirmosios sistemos problemas ir minusus. Aprašoma detali antrojo stendo struktūra ir sudedamosios dalys, aptariamas algoritmas. Toliau identifikuojami pavojingi manevrai, aprašomi jų požymiai, bei atliekami eksperimentai skirti nustatyti manevro pavojingumą ir atlikimo galimybes, sudaroma manevrų duomenų bazė. Aprašoma eksperimentų požymių charakterizavimo seka, palyginami filtravimo metodai ir išrenkamas optimaliausias filtras. Taip pat skaičiuojama standartinė deviacija manevro požymio kokybei nustatyti. Galiausiai aprašomi 5 naudoti algoritmai manevrų aptikimui.

Trečioje dalyje palyginami darbo rezultatai, kurie gauti „*Matlab*“ programinio paketo pagalba. Tirti metodai palyginami teisingo ir klaidingo manevro aptikimo atvejais. Iš gautų rezultatų išrenkami kiekvienam tirtam pavojingam manevrui po algoritmą ar metodą, kuris parodė geriausius rezultatus.

Simpukas Dominykas. Research on Methods for Maneuvering Identification of Micromobility Vehicle. Master's Final Degree Project / supervisor Assoc. Prof. Pranas Kuzas; Faculty of Electrical and Electronics Engineering, Kaunas University of Technology.

Study field and area (study field group): Electronics Engineering, Engineering Sciences.

Keywords: Micro-mobility, dangerous maneuver, signal processing, feature.

Kaunas, 2024. 74 p.

Summary

This work analyses various dangerous maneuvers that are performed with micro-mobility vehicles, which can harm the user or pedestrians. A system is developed, which is capable of collecting data from various sensors, several methods are evaluated, which are used in detecting maneuvers from sensor data, identifying them and determining their danger level. The aim of this work – conduct research on the most common micro-mobility dangerous maneuvers, identify them, with the help of sensors record them and using selected algorithms, identify, classify, and compare them. The algorithms should require minimal computational resources, be suitable for signal recognition, and perform calculations as quickly as possible.

The first part analyzes the most common sensors used in mobility tasks, describing their principles of operation and measured quantities. Literature sources and related research on object motion and maneuver recognition are also analyzed. Finally, an overview of a company which produces smart scooters with integrated safety features is provided.

The second part discusses two data recording methods that were used: a smartphone and developed embedded system. First, the mobile phone set up is described for collecting accelerometer data, highlighting its pros and cons, followed by data visualization and preliminary analysis. Second, the development process of the second data acquiring system is provided, which solves the problems of the first system. The detailed structure and components of the second experiment stand are described along with the algorithm of entire system. Later, dangerous maneuvers are identified, their characteristics described, and experiments are conducted to determine the danger level of each maneuver, a database is created. The sequence of feature extraction from experiment data is described, filtering methods are compared, and the optimal filtering method is selected. Additionally, the standard deviation of each maneuver feature is calculated to determine its overall quality. Finally, five algorithms used for maneuver detection are described.

In the third part, the results, obtained using “*Matlab*” software package, are analyzed. The examined methods are compared in case of correct and incorrect maneuver detection. From the obtained results, the algorithm or method that showed the best results for each researched dangerous maneuver is selected.

Turinys

Lentelių sąrašas	7
Paveikslų sąrašas	8
Santrumpų sąrašas	10
Įvadas.....	11
1. Teorinė tyrimo dalis	13
1.1. Mikromobilumo priemonių tipai	13
1.2. Dažniausiai naudojamos sistemos judėjimo uždavinių sprendimuose	14
1.2.1. Akcelerometras	14
1.2.2. Giroskopas	15
1.2.3. Momentinio greičio matuokliai	15
1.3. Mobiliumi telefonu pagrįsta neblaivaus vairuotojo atpažinimo sistema	17
1.4. Kritimo aptikimo sistema esanti apyrankėje	18
1.5. Žemės drebėjimo aptikimo algoritmų palyginimas naudojant mobilaus telefono jutiklius	19
1.6. „Neuron Mobility“ siūlomas produktas.....	22
2. Metodinė tyrimo dalis	23
2.1. Tyrimo struktūrinė schema.....	23
2.2. Pavojingų manevrų identifikavimas ir aprašymas.....	23
2.3. Eksperimento maketas	25
2.3.1. Duomenų surinkimo sistema pasitelkiant išmanųjį telefoną	25
2.3.2. Duomenų surinkimo sistema pasitelkiant įterptinę sistemą	29
2.3.3. Eksperimento stendo algoritmas duomenų surinkimui	31
2.4. Duomenų bazės sudarymas	33
2.5. Duomenų paruošimas ir požymių charakterizavimas.....	38
2.6. Standartinės deviacijos skaičiavimas	45
2.7. Slenkančio vidurkio filtro modeliavimas	47
2.8. Manevrų identifikavimo metodai	49
2.8.1. Kros-koreliacijos metodas	49
2.8.2. Signalo energijos metodas	51
2.8.3. Gaubtinės analizės metodas.....	53
2.8.4. Slenkstinės vertės metodas	54
2.8.5. Dinaminio laiko deformavimo metodas	56
3. Eksperimentų rezultatai	58
3.1. Metodų gebėjimas teisingai aptikti manevrą.....	58
3.2. Metodų atsparumas klaidingiems manevrų aptikimams	59
3.3. Rezultatų apibendrinimas ir pasiūlymai, kaip juos gerinti	62
Išvados	64
Literatūros sąrašas	65
Priedai.....	68
1 priedas. Kros-koreliacijos metodo funkcija „Matlab“ programoje	68
2 priedas. DTW metodo funkcija „Matlab“ programoje.....	70
3 priedas. Ribinės vertės metodo funkcija „Matlab“ programoje	72
4 priedas. Gaubtinės vidutinės vertės metodo funkcija „Matlab“ programoje	73
5 priedas. Vidutinės energijos metodo funkcija „Matlab“ programoje.....	74

Lentelių sąrašas

1 lentelė. Paskaičiuotos kros-koreliacijos pikų vertės prie 3 manevrų.....	51
2 lentelė. Paskaičiuotos DTW vertės prie 3 manevrų.....	57
3 lentelė. 3 kliūčių ir 2 kliūčių apvažiavimo manevro metodų rezultatai.....	58
4 lentelė. Šuolio nuo bortelio manevro metodų rezultatai.....	59
5 lentelė. Blogo kelio dangos nustatymo metodų rezultatai.....	59
6 lentelė. Visų manevrų klaidingų aptikimų rezultatai.....	60
7 lentelė. 3 kliūčių požymių aptikimo 2 kliūčių apvažiavimo eksperimentuose rezultatai.....	60
8 lentelė. 3 ir 2 kliūčių apvažiavimo manevrų aptikimo, kai amplitudė yra mažesnė, rezultatai.....	61
9 lentelė. 3 ir 2 kliūčių apvažiavimo manevrų aptikimo esant tvarkingam važiavimui rezultatai	61
10 lentelė. Bendri 3 ir 2 kliūčių apvažiavimo algoritmų klaidingų manevrų nustatymo rezultatai ..	62
11 lentelė. Blogos kelio dangos klaidingo nustatymo rezultatai	62
12 lentelė. Tinkamiausių naudoti algoritmų skirtingiems pavojingiems manevrams rezultatai	62

Paveikslų sąrašas

1 pav. Mikromobilumo transporto pavyzdžiai [3].....	13
2 pav. 3 ašių MEMS akcelerometro struktūra ir vienos ašies principinė schema [5].....	15
3 pav. Principinė giroskopo schema [4]	15
4 pav. Principinė enkoderio veikimo schema [8]	16
5 pav. Neblaivaus vairuotojo aptikimo sistemos modelis [12].....	18
6 pav. Sprendinių medžio struktūra [17]	20
7 pav. LDA pavyzdys [18]	20
8 pav. Atraminių vektorių klasifikatoriaus veikimo pavyzdys [19]	21
9 pav. K-arčiausių kaimynų algoritmo pavyzdys [20]	21
10 pav. "Neuron Mobility" įmonės identifikuoti pavojingi manevrai [22].....	22
11 pav. Tyrimo struktūrinė schema	23
12 pav. Pavojingo manevro "Vinguriavimas" principinė schema.....	24
13 pav. Pavojingo manevro "Šuolis nuo bortelio" principinė schema	25
14 pav. Pavojingo manevro "Greitas važiavimas prasta danga" principinė schema	25
15 pav. „Physics Toolbox Suite“ programėlės vaizdas	26
16 pav. Įrašytų duomenų pavyzdys	26
17 pav. Gautas manevras laike	27
18 pav. Telefono pozicija eksperimento metu.....	28
19 pav. Šuolio nuo bortelio Y ašies signalo FFT rezultatas	28
20 pav. Galutinai surinkta eksperimento sistema	30
21 pav. Eksperimento stendo komponentai	30
22 pav. Jutiklio vektorių vizualizacija	31
23 pav. Eksperimento stendo algoritmas	33
24 pav. Sinc formos signalas [27]	34
25 pav. Manevro schema	35
26 pav. Vienetinio impulso formos signalas	36
27 pav. Manevro schema	37
28 pav. Sudarytos duomenų bazės iškarpa	38
29 pav. 2 kliūčių apvažiavimo eksperimento neapdoroti akcelerometro X, Y, Z ašių duomenys	39
30 pav. 2 kliūčių apvažiavimo eksperimento akcelerometro ir giroskopo ašių FFT rezultatai.....	39
31 pav. 2 kliūčių apvažiavimo eksperimento akcelerometro X ašies filtravimo palyginimai.....	40
32 pav. 2 kliūčių apvažiavimo eksperimentų akcelerometro duomenys sinchronizuoti laike	41
33 pav. 2 kliūčių apvažiavimo eksperimentų akcelerometro Y ašių sinchronizuoti duomenys.....	42
34 pav. Charakterizuoti manevrų požymiai.....	43
35 pav. Šuolio nuo bortelio neapdoroti akcelerometro Z ašies duomenys.....	43
36 pav. Šuolio nuo bortelio eksperimentų akcelerometro Z ašių sinchronizuoti duomenys.....	44
37 pav. Šuolio nuo bortelio manevro požymis	44
38 pav. Apskaičiuotos standartinės deviacijos	45
39 pav. Atvaizduotos standartinės deviacijos su papildoma piko deviacija.....	46
40 pav. Trijų kelio dangų važiavimo FFT palyginimas	48
41 pav. Slenkamojo vidurkio filtrų palyginimas	49
42 pav. 3 kliūčių apvažiavimo akcelerometro Y ašies kros-koreliacijos rezultatas	50
43 pav. Energijos vidutinės vertės priklausomybė nuo kelio dangos.....	52
44 pav. Blogos kelio dangos signalo gaubtinės atskaitų tarpų palyginimai	53

45 pav. Gaubtinės vidutinės vertės priklausomybė nuo kelio dangos	54
46 pav. Ribinės vertės vizualizacija	55
47 pav. DTW atstumų matricos ir optimalaus kelio pavyzdys.....	56
48 pav. DTW algoritmo rezultato pavyzdys [35]	57
49 pav. Kros-koreliacijos metodu sėkmingai aptiktas manevras visose trijose ašyse.....	59
50 pav. DTW metodu klaidingai aptiktas manevras dvejose ašyse.....	61

Santrumpų sąrašas

Santrumpos:

IMU – Inercinis matavimo blokas (angl. *Inertial Measurement Unit*);

MEMS – Mikroelektromechaninė sistema (angl. *Micro-ElectroMechanical System*);

GNSS – Globali navigacinė palydovų sistema (angl. *Global Navigation Satellite System*);

HAR – Žmogaus elgsenos atpažinimas (angl. *Human Activity Recognition*);

ML – Mašininis mokymasis (angl. *Machine Learning*);

RMS – Vidutinė kvadratinė vertė (angl. *Root Mean Square*);

FFT – Greitoji Furjė transformacija (angl. *Fast Fourier Transform*);

SPI – Nuosekli periferijos sąsaja (angl. *Serial Peripheral Interface*);

I2C – Integruota schema (angl. *Inter-Integrated Circuit*);

UART – Universali asinchroninė imtuvo-siųstuvo sąsaja (angl. *Universal Asynchronous Receiver – Transmitter*);

FIFO – Pirmas įeina, pirmas išeina (angl. *First In, First Out*);

RIR – Ribotos impulsinės reakcijos;

ŽD – Žemų dažnių;

DTW – Diskretinis laiko deformavimas (angl. *Discrete Time Warping*).

Įvadas

Šiais laikais mobilumas tapo labai svarbiu žmonijos laimėjimu, be kurio negalėtų egzistuoti šiuolaikinė ekonomika ir visa mūsų visuomenė. Mobilumas visais laikais buvo tobulinamas ir ieškoma vis efektyvesnių metodų, kaip iš taško A nusigauti į tašką B. Technologijoms tobulėjant atsirado garo varikliai, dyzeliniai varikliai, bei elektriniai varikliai. Pastarųjų variklių bumą galima stebėti jau paskutinius kelis dešimtmečius, jie yra intensyviai tobulinami ir reklamuojami, nes tai yra tvarus, švarus ir netaršus metodas išgauti judėjimą. Vienas naujausių ir ypač išpopuliarėjusių metodų keliauti yra elektrinės mikromobilumo priemonės (angl. *Micro-mobility*).

Mikromobilumas - tai sąvoka, kuri vartojama apibūdinti lengvas, kompaktiškas, nedidelių gabaritų mobilumo priemones, skirtas įveikti trumpiems atstumams, dažniausiai urbanistinėse vietovėse [1]. Dažniausiai tokios transporto priemonės naudoja elektrinius variklius ir elektros baterijas, kaip galios šaltinį. Šis elektrinių transporto priemonių populiarumas prasidėjo rinkoje atsiradus pigiems ir vartotojams lengvai prieinamiems paspirtukams. Pastaruoju metu kiekvienas gyventojas gali vienai dienai ar ilgesniam laikui iškeisti brangų, kuro reikalaujantį ir kamščiuose stovėti verčiantį automobilį į kelis šimtus eurų kainuojantį ir namie elektra kraunamą paspirtuką. Iš vartotojo pusės, paspirtukas pasiekia nemažą greitį, turi pakankamai didelę bateriją, kurios paprastam vartotojui užtenka įveikti reikiamą atstumą bei juo galima važiuoti dviračių takais. Taip pat paminėtina, kad šios priemonės yra tvarios ir netaršios. Tačiau yra ir blogoji visų šių technologijų dalis. Kadangi šios transporto priemonės yra ganėtinai naujai paplitusios ir išpopuliarėjusios, joms nereikia jokių vairuotojo teisių, techninių apžiūrų ar kitų reikalavimų, jomis gali naudotis neapmokyti nepilnamečiai, vaikai ar neblaivūs asmenys. Nėra jokios kontrolės ar reikalavimų, kas gali vairuoti tokias priemones, o kadangi, su jomis galima išvystyti tikrai didelius greičius (modifikuoti variantai gali važiuoti ir 30 ar 50 km/h greičiu), tai ima kelti pavojų kitiems eismo dalyviams. Dar vienas bendras jų bruožas yra tai, kad šias priemones naudoja ne tik dviračių takais ar gatvėmis, o ir pėsčiųjų takais, alėjomis, nes ne visuose miestuose dviračių takų infrastruktūra yra gera. Dėl visų šių priežasčių daugėja įvairių incidentų, kuriuose dalyvauja elektrinių priemonių vartotojai. Tai ypač padidėjo atsiradus greitai išnuomojamiems ir lengvai prieinamiems paspirtukams. Žmonės nesaugodami ne savo turto dažnai keliose naudojasi tokiais paspirtukais, sukelia papildomą apkrovą varikliui, baterijai, taip pat agresyviai jais naudojasi, šokinėja nuo šaligatvio bortelių, per vėlai ir staigiai stabdo, taip sukeldami pavojų pėstiesiems, važinėja dideliu greičiu pėsčiųjų takais, pavojingai manevruoja tarp pėsčiųjų. Atlikdami šiuos manevrus vartotojai sukelia pavojų ne tik sau bet ir aplinkiniams [2]. Dėl to dauguma valstybių ėmė riboti mikromobilumo priemonių vartojimą, kai kur jas uždraudė naudoti šaligatviuose, kai kur joms reikia teisių, o kai kur, atsirado ir visiškas draudimas naudoti šias priemones. Ir nors tai atrodo logiškas ir racionalus sprendimas, tai sukelia daug neigiamų emocijų. Dėl kai kurių vartotojų neatsakingo ir neatsargaus naudojimo ir elgesio, nukenčia likusi visuomenės dalis, nors tai ir daug pranašumų turinti mobilumo rūšis, padedantis ir taupantis visų laiką ir pinigus. Tačiau draudimas nėra pati geriausia išeitis. Egzistuoja ir kitų metodų kaip pažaboti neatsakingus vartotojus, o nusižengusius, individualiai nubausti, taip atsirastų kompromisas, tarp saugumo ir taisyklių laikymosi, bei efektyvios ir tvarios mobilumo priemonės naudojimo.

Pagrindinė problema, su kuria susiduriama yra ta, kad reikia būdo atskirti atsakingus vartotojus nuo pažeidėjų, nes tik taip būtų galima išspręsti šią problemą visiškai neuždraudžiant ir neapribojant mikromobilumo priemonių. Dauguma dabartinių priemonių stokoja saugumo sistemų, kurios gebėtų sekti vartotojo elgseną naudojantis mikromobilumo priemone, tačiau rinkoje pradeda atsirasti gamintojų, kurie siūlo savo transporto priemones, kurios geba atpažinti pavojingus ir neatsargius manevrus.

Pagrindinis **darbo tikslas** - naudojantis įvairių inercinių jutiklių duomenimis (akcelerometru, giroskopu bei momentinio greičio matuokliu) sukurti duomenų surinkimo sistemą ir nesaugių manevrų nustatymo ir stebėjimo algoritmą, kuris seks vartotojo važiavimo elgseną ir maneversus. Ištirti kelis esamus atpažinimo algoritmus, juos palyginti, įvertinti jų privalumus ir trūkumus, bei išrinkti tinkamiausią. Šiam darbui buvo iškelti tokie uždaviniai:

1. ištirti esamus judesio atpažinimo metodus, pagrįstus inertinių jutiklių duomenų analize;
2. sukurti testavimo standą ir duomenų surinkimo sistemą elektrinio paspirtuko manevrų skaitmenizavimui inertiniais ir bekontakčiais jutikliais;
3. išanalizuoti naudojamus algoritmus signalų atpažinime, susijusius su manevravimu ir judesiu;
4. išskirti ir identifikuoti pavojingus maneversus, jų rūšis, požymius bei pagrindinius parametrus;
5. apdoroti duomenų surinkimo sistemos įrašytų signalų pavyzdžius naudojant matematinės signalų analizės algoritmus ir realizuoti pavojingų manevrų atpažinimo algoritmus, juos palyginti, išrinkti tinkamiausią kiekvienam manevrui.

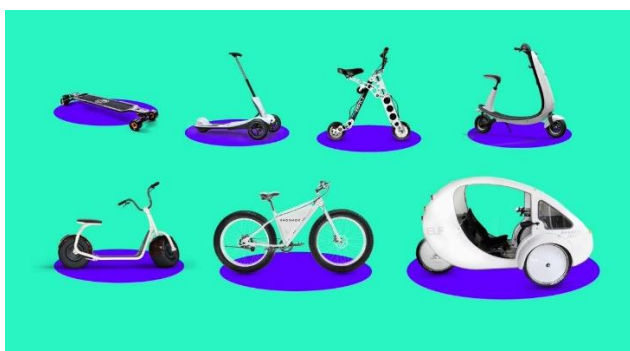
1. Teorinė tyrimo dalis

Šioje dalyje bus aptariamos mikromobilumo transporto priemonių rūšys, taip pat išrenkama tolimesnio tyrimo metu naudojama mikromobilumo priemonė. Taip pat aptariami literatūros šaltiniai, kurie aprašo ir tiria įvairius algoritmus ar sistemas, gebančias nustatyti tam tikrus judesius ar įvykius, taip pat bus analizuojama esama rinka ir sprendimai, naudojami nustatyti netaisyklingus mikromobilumo priemonių naudojimo atvejus. Bus aptartos pagrindinės sistemos ir jutikliai, naudojami tokio pobūdžio uždavinių sprendimuose, analizuojami jų veikimo principai ir pateikiamos principinės schemas.

1.1. Mikromobilumo priemonių tipai

Kiekviena mikromobilumo priemonė yra savita ir unikali, dažnai, tai daugumai nematytai atrodanti transporto priemonė, o kartais tai tiesiog patobulinta ir atnaujinta senai pažįstama priemonė. Kelios pagrindinės ir dažniausiai sutinkamos mikromobilumo priemonės išvardintos žemiau:

- elektriniai paspirtukai - tai maži, dažnai ranka pakeliami ar net sulankstomi, paspirtukai, kurie turi elektros variklį, skirtą padėti vartotojui judėti arba visiškai perimti judėjimą. Turi elektros baterijas, kurių talpos užtenka įveikti nuo 5 iki 65 km priklausomai nuo modelio, gamintojo ir naudojamo variklio bei modifikacijų. Tinka tiek pramoginiam važinėjimui, tiek verslo ar darbo tikslams kaip dalinis ar visiškas pakaitalas automobiliui.
- elektriniai dviračiai - kaip ir paprasti dviračiai, gali būti minami, tačiau esant poreikiui, elektrinis variklis gali padėti minti (minant į kalną arba nuo kalno) arba pilnai perimti riedėjimą. Dažniausiai brangesni už paspirtukus, tačiau daug saugesni ir stabilesni, gali nuvykti didesnius atstumus, išvystyti didesnius greičius. Egzistuoja elektrinės pavaros, kurias uždėjus ant paprasto dviračio, galima jį paversti elektriniu, t. y. pavara padės vartotojui minti dviratį arba trumpam laikui perims ratų sukimą ir palaikys greitį.
- elektriniai vienaračiai – nestabiliausia transporto priemonė, nes visas prietaisas realizuotas viename rate (baterija, motoras, valdikliai) ir neturi jokių stabilumo ar laikymosi priemonių. Dažnai skirti pramoginiam važiavimui, tačiau ima populiarėti ir kaip transporto priemonės pakaitalas maisto ir siuntinių pristatymo srityje.
- riedžiai - panašiai kaip ir vienaratis, neturi jokių stabilumą suteikiančių atramų, naudojama be rankų, vairavimas ir važiavimas pagrįstas vartotojo kūno pasvirimu į norimą pusę, nepatyrę vartotojai gali neišlaikyti pusiausvyros ar pasiekę per didelį greitį kristi.



1 pav. Mikromobilumo transporto pavyzdžiai [3]

Aukščiau išvardintos priemonės yra dažnai sutinkamos kasdieniniame gyvenime, tačiau viena iš jų yra labiausiai paplitusi ir išpopuliarėjusi, tai elektrinis paspirtukas. Nors paspirtukus yra įsigijęs neretas asmuo, verslai, teikiantys nuomos paslaugas, išgyvena aukso amžių ir savo paspirtukų parkus

plečia. Dėl šios priemonės prieinamumo ir populiarumo elektrinis paspirtukas ir buvo pasirinktas, kaip šio tyrimo objektas, pagrindinė mikromobilumo priemonė, kurios manevrai bus tiriami.

1.2. Dažniausiai naudojamos sistemos judėjimo uždavinių sprendimuose

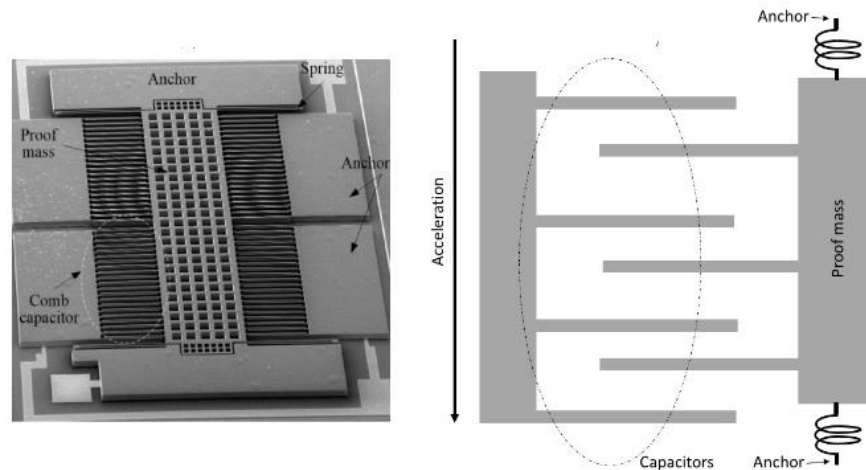
Daugumoje tirtų mokslinių straipsnių duomenų surinkimui yra naudojama IMU sistema (angl. *Inertial Measurement Unit*). Tai prietaisas, kuris dažniausiai susideda iš giroskopo - jutiklio, skirto aptikti kampinį greitį bei akcelerometro - jutiklio, kuris matuoja žemės traukos jėgą, bei jėgą, kuri veikia objektą šiam judant. Kartais IMU blokai turi magnetometrą - jutiklį gebantį aptikti magnetinį lauką, kuris veikia objektą, tačiau paminėtina tai, kad toks jutiklis šiame tyrime nebus reikalingas. Naudodamasis visais šiais jutikliais IMU blokas geba apskaičiuoti objektą veikiančią gravitacijos jėgą, jo pagreitį bei orientaciją erdvėje. Atitinkamai nuo panaudojimo srities, IMU blokai yra kelių klasių:

- vartotojo/automobilinė klasė;
- industrinė klasė;
- karinė klasė;
- navigacinė klasė.

Visos šios klasės skiriasi kaina, tikslumu ir panaudojimo sritimis. Nagrinėjamu atveju tiksliausia būtų naudoti pirmąją klasę, nes tai turėtų būti pigi, plačiai prieinama sistema [4].

1.2.1. Akcelerometras

Pagrindinis jutiklis, kurio duomenys bus naudojami, kuriant algoritmą, tai akcelerometras. Šis jutiklis dažniausiai naudojamas aptikti inercinę akceleraciją bei greičio pokytį per laiką, jis dažniausiai naudojamas išmaniuosiuose telefonuose, automobilių sistemose bei įvairiose judesio nustatymo sistemose. Dažniausiai naudojami MEMS (angl. *Micro-ElectroMechanical System*) jutikliai, kurių veikimas pagrįstas spyruoklės principu. Tai labai maža mechaninė sistema, kuri sudaryta iš tam tikros masės objekto, prikabinto prie spyruoklės. Masė gali laisvai judėti spyruoklės suspaudimo ir ištempimo kryptimi, tai vadinama jutimo ašimi, šių ašių dažniausiai yra 3, kad būtų galima pamatuoti objekto 3 judėjimo laisvės laipsnius (X, Y ir Z ašyse). Objektui pradedant greitėti, masė nori išlaikyti savo poziciją dėl inercijos, taigi spyruoklė pradeda deformuotis (susitraukti arba išsitempti, atitinkamai nuo judėjimo krypties). Tokiu būdu sukuriama jėga spyruoklėje, kuri yra proporcinga akceleracijai. Akceleracija matuojama per inercinės masės poslinkį dažniausiai talpiniu būdu. Kiekvienam laisvės laipsniui (ašiai) yra naudojamas atskiras „šukų“ komplektas.

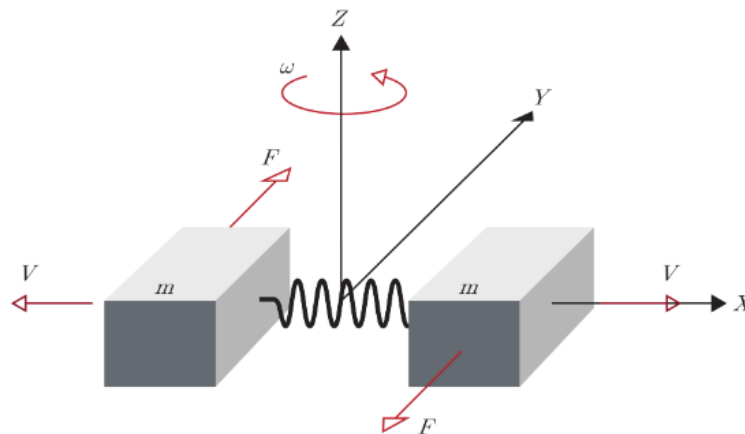


2 pav. 3 ašių MEMS akcelerometro struktūra ir vienos ašies principinė schema [5]

Akceleracija matuojama G vienetais, kur vienas G yra $9,81 \text{ m/s}^2$.

1.2.2. Giroskopas

Giroskopas naudojamas objekto kampiniam greičiui nustatyti. Šis jutiklis tai atlieka pasitelkdamas Koriolio efektą. Koriolio efektas yra jėga veikianti kūną ant besisukančios platformos. Platformai sukantis pagal laikrodį, inercinė jėga veiks kūną į kairę pusę, tolstant nuo platformos centro ši jėga didės. Jėgos didėjimas tolstant nuo centro ir yra Koriolio akceleracija. Panaudojus šį fizikinį principą ir MEMS sistemą galima apskaičiuoti objekto pasisukimą erdveje naudojant laisvai judančios masės struktūrą. Ši struktūra yra pritvirtinta ir pastoviai vibruoja tam tikroje ašyje, pakitus objekto orientacijai ar atsiradus kampiniam greičiui, jutiklis registruoja šį pasisukimą [6]



3 pav. Principinė giroskopo schema [4]

Giroskopo kampinis greitis matuojamas $^\circ/\text{s}$ (dps) arba rad/s .

1.2.3. Momentinio greičio matuokliai

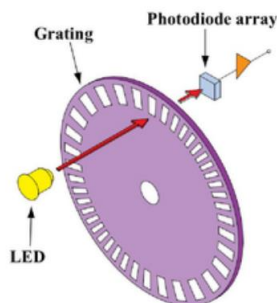
Dar vienas svarbus parametras, reikalingas įvertinti objekto judėjimą, yra momentinis greitis. Kadangi mikromobilumo priemonė juda nepastoviu greičiu, negalima vertinti greičio, kaip nekintančio dydžio, taigi reikalingas momentinis greitis tam tikru laiko momentu, o matavimo dažnis turėtų būti parenkamas toks, kad būtų gaunama kuo geresnė greičio rezoliucija ir greičio grafikas būtų kuo tikslesnis. Vienas iš metodų yra matuoti nuvažiuotą atstumą per tam tikrą laiką. Tai galima atlikti

naudojantis GNSS modulį, priimant koordinates ir jas verčiant į nuvažiuotą atstumą per laiką, tačiau šis metodas nebūtų tinkamas uždaroje patalpose, tuneliuose, taip pat priimamos koordinatės būtų ganėtinai lėtu dažniu, egzistuoja ir pozicijos registravimo paklaidos, kurios pablogins rezultatus. Kitas metodas yra naudoti akcelerometro duomenis. Vidutinė akceleracija yra išreiškiama, kaip vidutinio greičio diferencialas padalintas iš laiko diferencialo:

$$\vec{a} = \frac{d\vec{v}}{dt} \quad (1)$$

Tai parodo, kad pagreitis ir greitis yra priklausomi vienas nuo kito, taigi turint akceleraciją, galima išsivesti greičio formulę, atlikus akceleracijos integravimą. Taip pat atlikus dvigubą pagreičio integravimą, galima gauti nuvažiuotą atstumą [7]. Šis metodas yra plačiai naudojamas pozicijai nustatyti naudojant IMU bloką. Tačiau šis metodas turi didelį trūkumą, be papildomo apdorojimo, jis veikia tik idealiomis sąlygomis, nes realus IMU blokas, o konkrečiai, akcelerometras, turi tam tikrą matavimo paklaidą ir gaunama akceleracija turi paklaidas, taigi atliekant skaičiavimus ir integravimą bus gaunamas netikslus rezultatas. Skaičiuojant nuvažiuotą atstumą dėl šios pastovios paklaidos bus gaunamas vis didesnis nuokrypis ir paklaida, nes atliekant dvigubą integravimą akcelerometro duomenims ir skaičiuojant nuvažiuotą atstumą, paklaidos kaupiasi, nes naujos pozicijos nustatymui naudojami seni duomenys, kurie taip pat yra su paklaida. Dėl šių priežasčių, judančio objekto greičiui nustatyti dažniausiai naudojami enkoderiai.

Enkoderis - tai elektromechaninis prietaisas, kuris konvertuoja mechaninį judesį ar poziciją į elektrinį signalą. Enkoderiai dažniausiai naudojami aptikti besisukančių kūnų poziciją ar greitį, tačiau egzistuoja ir tiesiniai enkoderiai, kurie matuoja tiesiškai judančio kūno greitį ar poziciją. Rotaciniai enkoderiai (angl. *Rotary encoders*) matuoja kampinę poziciją arba greitį pasitelkiant atraminius taškus, esančius ant besisukančio objekto. Tai dažniausiai būna vienodu atstumu išpjautos kiaurymės (tada naudojami optiniai enkoderiai) arba įmontuoti magnetai (tada naudojami Halo efekto enkoderiai). Žinant atraminių taškų kiekį ir atstumą tarp jų, objektui besisukant, enkoderiu galima užfiksuoti, kiek atraminių taškų per matuojamą laiką enkoderis aptiko, taip apskaičiuoti besisukančio objekto greitį. Žinant kiek taškų aptiko, galima nustatyti objekto pasisukimą, arba nukeliatą atstumą, tačiau tikslumas priklausys nuo atraminių taškų kiekio, kuo jų daugiau, pozicija bus nustatyta tiksliau, tačiau esant labai dideliame taškų kiekiui, reikia įvertinti enkoderio galimybes išskirti taškus esant dideliame greičiui. Dar vienas metodas naudoti vieną atraminį tašką, tokiu būdu nustatyti tikslios objekto pozicijos nepavyks, tačiau galima apskaičiuoti apsisukimų skaičių per tam tikrą laiko periodą ir iš to apskaičiuoti visą nueitą kelią. Principinė enkoderio struktūra pateikta toliau:



4 pav. Principinė enkoderio veikimo schema [8]

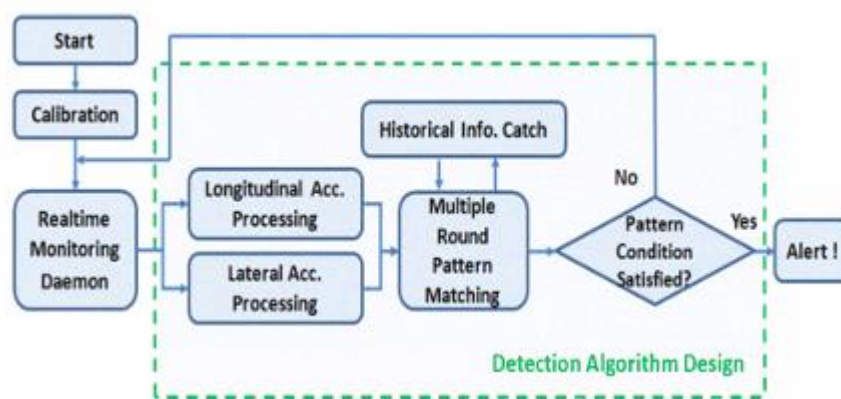
Enkoderio pagalba išmatuotas greitis dažniausiai matuojamas *RPM* - (angl. *Revolutions Per Minute*) apsisukimais per minutę, arba *km/h* – kilometrais per valandą (galimi ir kiti, išreiškiami matavimo vienetai pvz.: *cm/min*, *mm/s*, *mm/ms* ir t.t.).

Aukščiau išvardintos sistemos yra pagrindinės ir dažniausiai naudojamos judėjimo ir manevrų atpažinime bei klasifikavime. Iš šių jutiklių duomenų galima nustatyti įvairius parametrus: objekto laisvės laipsnius, judėjimo trajektoriją, esamą poziciją, prognozuojamą poziciją. Pagal atitinkamą užduotį galima stebėti tik dominančius parametrus ar požymius duomenyse. Tam reikalinga gerai žinoti, kokie įvykiai yra reikalingi, ko reikia ieškoti duomenyse, ir į ką nekreipti dėmesio. Todėl kiekvienas uždavinys yra savitas ir skirtingas, negalima skirtingiems požymiams taikyti vienodų kriterijų ar apdorojimo metodų, reikia gerai išanalizuoti esamą problemą ir ieškoti geriausių sprendimo metodų. Sekančiuose skyriuose bus apžvelgta keletą metodų ir algoritmų, naudojamų atpažinti konkrečius požymius ar įvykius, kurie nebūtinai tiks šio darbo analizuojamai temai, tačiau padės geriau suprasti, kokia įvykių seka yra naudojama išgauti rezultata.

1.3. Mobilium telefonu pagrįsta neblaivaus vairuotojo atpažinimo sistema

Kasmet neblaivių vairuotojų sukelti eismo įvykiai nusineša labai daug gyvybių įvairiose šalyse. Vien Jungtinėse Amerikos Valstijose, kasdien žūsta apie 37 žmonės, kurie patenka į neblaivaus vairuotojo sukeltą eismo įvykį [9]. Tai labai didelė problema, kurią pažaboti efektyvių priemonių ieškoma ir šiai dienai. Vieną sprendimo metodą pasiūlė Kinijos ir JAV universitetų mokslininkų atliktas tyrimas, kuriame būtų panaudojamas mobilusis telefonas, aptikti neblaivų vairuotoją. Alkoholis ar kitos svaiginančios medžiagos, visų pirma, paveikia vartotojo smegenų veiklą. Neblaivus asmuo pasižymi lėtesniu reakcijos laiku, ilgesniu situacijos vertinimo ir sprendimo priėmimo laiku bei koordinacijos praradimu. Stabdymo laikas prailgėja iki 2,2 sekundžių esant apsisvaiginus alkoholiu [10]. Taip pat neblaivus vairuotojas sunkiai geba išlaikyti automobilį ar kitą transporto priemonę kelio linijų režiuose, galima stebėti netolygų vairavimą ar vinguriavimą esant tiesiam keliui. Taip pat alkoholis veikia emocijas, taigi neblaivūs vairuotojai gali jaustis per daug pasitikintys savimi, dėl ko dažnai stebimas greičio viršijimas, vėlesnis stabdymas ar kelių ženklų nepaisymas [11]. Visi šie bruožai apibūdina neblaivaus vairuotojo elgseną, kurią galime atpažinti vizualiniu būdu. Išanalizuotame straipsnyje buvo iškelta idėja naudoti mobilųjį telefoną, kaip prietaisą, skirtą atpažinti neblaivų vairuotoją. Kadangi šiais laikais kiekvienas turi išmanųjį telefoną, o jame yra įmontuota daug įvairių jutiklių, tarp kurių yra ir anksčiau aptartas IMU blokas, jo pagalba galima stebėti vairuotojo manevrus ir aptikus požymius, rodančius, kad vairuotojas yra neblaivus ir imtis tam tikrų saugumo priemonių. Pagrindiniai bruožai, kuriuos galima stebėti akcelerometro pagalba ir nustatyti ar vairuotojas yra neblaivus, tai nesugebėjimas vairuoti automobilio tiesiai, negebėjimas važiuoti tarp eismo linijų, dreifavimas link vieno ar kito kelkraščio bei posūkio atlikimas itin dideliu lanku. Taip pat galima stebėti neįprastus akceleracijos ir lėtėjimo požymius, itin staigius greitėjimo ar stabdymo, bei negebėjimo išlaikyti pastovų greitį bruožus. Straipsnyje pasiūlyta idėja neprisiriša prie konkretaus mobilaus telefono modelio ir nereikalauja papildomos aparatinės dalies, visas sistema susideda iš keturių dalių:

1. duomenų surinkimo dalis (vidinis IMU blokas);
2. kalibravimo sistema;
3. duomenų apdorojimas;
4. perspėjimo sistema.



5 pav. Neblaivaus vairuotojo aptikimo sistemos modelis [12]

Sistema veikia autonomiškai, jai nereikia papildomo vartotojo valdymo ar paleidimo, ji automatiškai pradeda veikti fone vartotojui nežinant, kai yra pajuntama, kad mobilusis telefonas yra judančiame automobilyje. Kadangi telefonas gali būti padėtas bent kokia pozicija, o nuo to priklausys akcelerometro duomenys, bus naudojami giroskopo duomenys nustatyti telefono orientaciją, ir pagal tai nuspręsti, kur yra judėjimo kryptis. Tai yra svarbu, nes akcelerometro pozicijai esant nežinomai ar besikeičiančiai, jo ašių duomenys gali rodyti vis kintančius reikalingus duomenis. Tyrimo algoritmas skaičiuoja dviejų tipų akceleraciją: šoninę ir išilginę. Vėliau naudodamas šias dvi vertes, vykdo požymių atpažinimą, jas klasifikuoti ir priskirti įprastam vairavimui arba neblaivaus vairuotojo vairavimui. Rezultate, autoriai palygina algoritmo klaidingai įvertintus manevrus ir praleistus manevrus. Pagrindinė problema, telefonu grįstoje aptikimo sistemoje, yra ta, kad telefonas turėtų būti stacionarioje vietoje, kitaip matavimo duomenys bus labai netikslūs ir klaidingi [12].

Šiame straipsnyje nebuvo analizuojami naudojami algoritmai signalų atpažinime, tačiau buvo iškelta labai inovatyvus sprendimas panaudoti mobilųjį telefoną. Šią idėją galime pritaikyti ir kuriamoje sistemoje, kadangi norint naudotis nuomojamų paspirtukų paslaugomis, reikia turėti išmanųjį telefoną. Jo pagalba galima stebėti vartotojų manevrus ir elgseną važiuojant paspirtuku, tačiau šis metodas nebūtų pilnai tinkamas stebėti nuosavų paspirtukų vartotojus.

1.4. Kritimo aptikimo sistema esanti apyrankėje

Dar vienas išanalizuotas straipsnis buvo kritimo manevro aptikimo sistema, esanti ant žmogaus riešo. Straipsnis gvildena senų žmonių, kurie gyvena vieni, problemą. Ši problema yra labai opi, kadangi vis daugiau ir daugiau asmenų gyvena vieno nario šeimos ūkyje, dėl priežasčių, tokių, kaip ankstyvos mirtys, bei labai padidėjusiu skyrybų skaičius. Lietuvoje, pagal pateiktus duomenis, 45% asmenų, kurie yra 65 metų ir vyresni, gyvena vieni [13]. Šis skaičius kelia labai didelę problemą, dėl jų pačių saugumo, kadangi seni žmonės yra padidėjusioje rizikos grupėje, jie dažniau serga įvairiomis ligomis ir įvykus nelaimei, bei nesant šalia kitų asmenų, tai gali lemti ir asmens mirtį. Dėl šių priežasčių, straipsnio autoriai nusprendė sukurti ir iširti privatumo nepažeidžiančią sistemą ant asmens riešo, kuri pasitelkdama IMU bloko duomenis gebės aptikti senyvo amžiaus asmens staigų kritimą. Straipsnis palygina kelias duomenų surinkimo sistemas, tokias kaip GNSS, biosensorius, IMU pagrįstas sistemas bei kameras, tačiau dėl privatumo ir tikslumo priežasčių buvo pasirinkta naudoti judėjimo jutiklius. Taip pat buvo tiriamas HAR (angl. *Human Activity Recognition*) metodas, skirtas atpažinti tam tikrus žmogaus atliekamus judesius pagal jutiklio duomenis. Jutikliais grįstos sistemos gali būti suskirstytos į 4 dalis:

1. duomenų surinkimo sistemos pasirinkimas ir kūrimas;

2. duomenų surinkimas;
3. filtravimas ir duomenų paruošimas;
4. HAR atpažinimas pasitelkiant norimą algoritmą ar mašininį mokymąsi.

Šiame straipsnyje naudojama ESP32 mikrovaldiklio sistema, kuri kritimo duomenis siunčia Wi-Fi ryšiu, tada bus registruotas kritimo įvykis ir kviečiama greitoji pagalba. Pirminiam apdorojimui pasitelkiamas „MobiAct“ duomenų paketas. Šis duomenų paketas yra laisvai prieinamas ir turi mobiliojo telefono surinktus akcelerometro duomenis, kurie atvaizduoja tam tikrus vartotojo judesius ir veiklas (stovėjimą, sėdėjimą, šokimą, bėgimą ir t.t.) [14]. Šis paketas yra tinkamas tirti HAR veiklai ir tinkamas tiriamo straipsnio pirminei duomenų bazei. Straipsnyje naudojamas algoritmas atpažinti kritimui yra pagrįstas giliuoju mokymusi (angl. *Deep learning*), pasitelkiama „TensorFlow“ biblioteką. Tai atvirojo kodo biblioteka, skirta apdoroti ir paruošti duomenis, naudoti esamus ar kurti naujus ML modelius, pritaikyti sukurtą modulį įterptinėje sistemoje, asmeniniame kompiuteryje, internetinėje svetainėje ar sudėtinguose tinkluose [15]. Ši biblioteka leidžia greitai ir nesunkiai sukurti prototipą ar galutinį produktą su integruotu ML algoritmu. Straipsnyje taip pat aprašoma naudojama sistema, sukuriama apyrankės prototipas su MPU6050 akcelerometro ir giroskopo moduliu, ESP32 mikrovaldikliu, taip pat SOS mygtuku ir paprasta vartotojo sąsaja. Rezultate buvo gautas 96,55% sistemos tikslumas. Buvo nustatyta, kad naudojant didesnę kiekį neuronų tinkle lemia didesnę tikslumą, tačiau dėl to išauga atminties bei energijos suvartojimas [16].

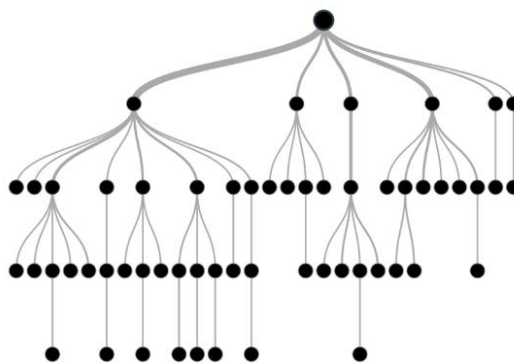
1.5. Žemės drebėjimo aptikimo algoritmų palyginimas naudojant mobilaus telefono jutiklius

Šiame straipsnyje nagrinėjami keli algoritmai, skirti signalų ar įvykių atpažinime. Straipsnyje tirama problema yra susijusi su žemės drebėjimo Indonezijos ir gretimose salose. Žemės drebėjimai kelia labai didelį pavojų tose vietose gyvenantiems žmonėms, taigi buvo tirama sistema, kuri padėtų aptikti žemės drebėjimą ir apie tai įspėtų vartotoją. Idėja naudoti išmanųjį telefoną kilo, dėl jų paplitimo ir galimybės naudoti vidinius jutiklius. Vienintelė problema, su kuria susiduria tokia apsaugos sistema yra ta, kad dėl žmogaus veiklos ir vidinių triukšmų, telefonui sunku išskirti žemės drebėjimo požymį nuo paprastų ir kasdieninių veiklų žmogaus gyvenime. Tyrime buvo palyginami kelių tipų atpažinimo algoritmai:

1. sprendinių medis (angl. *Decision Tree*);
2. tiesinė diskriminacinė analizė (angl. *Linear Discriminant Analysis*);
3. atraminių vektorių klasifikatorius (angl. *Support Vector Machine*);
4. K-arčiausių kaimynų algoritmas (angl. *K-nearest neighbor*).

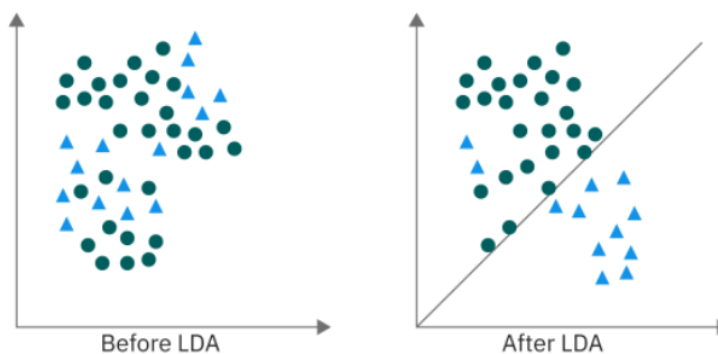
Be šių algoritmų buvo atliktas pradinis signalų apdorojimas. Kadangi naudojamas akcelerometras buvo registravo ir žmogaus judesius, turėjo vidinį triukšmą bei buvo matuojama pastovus Žemės gravitacinis pagreitis ($G = 9,8 \text{ m/s}^2$), surinkus duomenis juos reikėjo paruošti algoritmams. Straipsnyje siūlomas aukštų dažnių filtras signalų paruošimui. Kadangi G dydis yra pastovus ir visada matomas akcelerometro duomenyse, jis yra nuolatinė dedamoji, taigi pasitelkiant AD filtrą ir pašalinus šią dedamąją, būtų gaunami duomenys, be nuolatinės dedamosios. Po apdorojimo signalas skaldomas į kelis segmentus ir sistema išgaus požymius iš šių segmentų laiko ir dažnio srityse. Duomenų atpažinimui buvo sukurta įprastų žmogaus manevrų duombazė, skirtingiems eksperimento dalyviams buvo liepta atlikti specifinius veiksmus, telefonui esant jų kelnų bei marškinėlių kišenėse.

Žemės drebėjimo duomenys buvo paimti iš viešai prieinamų duomenų. Signalams atpažinti ir klasifikuoti buvo naudojami aukščiau išvardinti ML algoritmai. Pirmasis algoritmas, Sprendinių medis, tinkamas regresijos ir klasifikavimo užduotims, tai paprastas modelis naudojantis medžio topologiją. Pradinis kriterijus šakojasi į konkretesnius požymius, o šie, į dar konkretesnius. Sprendinių medžio veikimą galima suprasti, kaip paprastą *if-else* ciklą, jei kažkoks kriterijus atitinka požymį, analizuojama atitinkama šaka ir signalas nagrinėjamas iki tol, kol bus sumažinama entropija ir nežinomybė, kada priimtą signalą galima priskirti vienai iš esamų grupių.



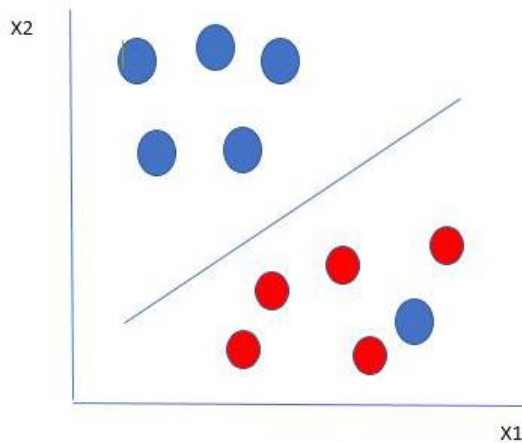
6 pav. Sprendinių medžio struktūra [17]

Tiesinė Diskriminacinė analizės metodas yra naudojamas spręsti klasifikavimo problemas, kai yra naudojama daug požymių klasių. Šis algoritmas tai pasiekia pasitelkdamas duomenų dimensijų prastinimą, jis veikia parinkdamas tiesinę priklausomybę tarp požymių, kurie identifikuoja dvi ar daugiau klasių. Taip pat šis algoritmas dar vadinamas duomenų dimensijų redukcijos metodu, nes jis daugiadimensinius duomenis suprastina iki vienos dimensijos.



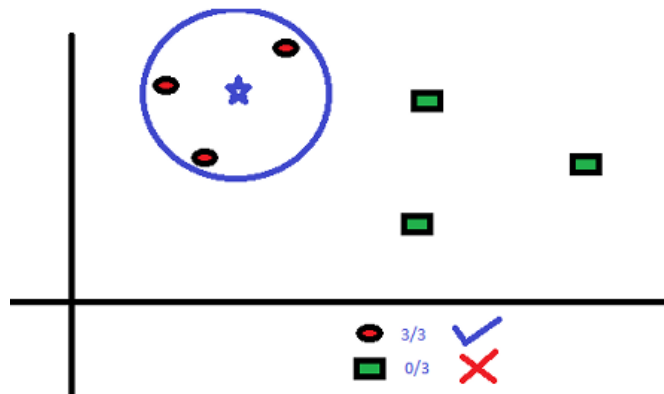
7 pav. LDA pavyzdys [18]

Atraminų Vektorių Klasifikatoriaus algoritmas yra skirtas klasifikavimui ir regresijai. Šio algoritmo pagrindinis uždavinys yra surasti optimaliausią ir geriausią ribą, kuri atskirtų du požymius į skirtingas klases. Riba, kada požymis yra vienos ar kitos klasės, turėtų būti kuo labiau nutolusi nuo požymių, pagal tai algoritmas ir ieško optimaliausio požymių klasifikavimo kriterijaus. Taip pat jis sugeba atrasti šią ribą net esant išskirtims (angl. *Outliers*).



8 pav. Atraminių vektorių klasifikatoriaus veikimo pavyzdys [19]

Paskutinis, K-arčiausių kaimynų algoritmas, kaip ir aukščiau išvardinti, geriausiai tinka klasifikavimo ir regresijos uždutims spręsti. Šis metodas veikia principu, kad panašūs duomenų taškai turi panašias reikšmes ir gali būti klasifikuojami vienai klasei. K yra kaimynų skaičius, kuris bus ieškomas klasifikuojant turimą požymį. Tada apie tiriama požymį sudaromas žiedas, kuriame turi būti bent trys jau žinomi duomenų taškai ir pagal jų klasę, tiriamaam požymiui priskiriama atitinkama klasė. Šis metodas naudoja daugumos balsų sistemą nustatyti tiriama objekto klasei, kadangi daugumos kaimyninių duomenų taškų klasių nuspręš, kuriai klasei reikia priskirti tiriama požymį.

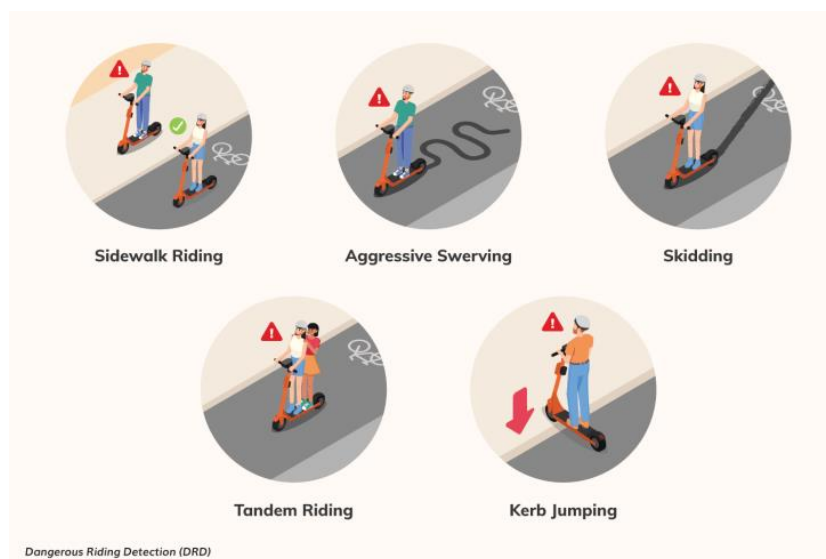


9 pav. K-arčiausių kaimynų algoritmo pavyzdys [20]

Remiantis šiais algoritmais ir jų klasifikavimo galimybėmis buvo atlikta turimų žmogaus elgsenos ir žemės drebėjimo duomenų priskyrimas vienai ar kitai klasei. Charakterizavus įvairius požymius, kaip visų ašių vidurkį, medianą, *RMS* vertę ir t.t. buvo gauta, kad žemės drebėjimo požymiai ganėtinai aiškiai skiriasi, nuo žmogaus veiklos požymių, žemės drebėjimo amplitudės ir vertės buvo kur kas mažesnės nei žmogaus veiklos duomenų. Atlikus visų algoritmų panaudojimą, autorius pabrėžia, kad algoritmo pasirinkimas priklauso ne tik nuo jo tikslumo, bet ir resursų naudojimo, algoritmo greičio, atminties kaštų ir panašiai. Sprendimų medis pateikė geriausius rezultatus greičio ir atminties suvartojimo kriterijuje, tačiau jis turėjo prasčiausią tikslumą, atraminių vektorių klasifikatoriaus algoritmas turėjo didžiausią šių resursų sueikvojimą, bet jo tikslumas buvo 26% didesnis, už sprendimų medžio. Algoritmo pasirinkimą lemia turimų resursų kiekis bei užduotyje reikalaujamas tikslumas, šio straipsnio autorius pasirinko K-arčiausių kaimynų metodą, kaip geriausią jo specifinei užduočiai dėl aukšto tikslumo ir ganėtinai mažos atminties suvartojimo [21].

1.6. „Neuron Mobility“ siūlomas produktas

Atliekant esamos, mikromobilumo priemonių, rinkos apžvalga buvo rasta Singapūre įsikūrusi įmonė, kuri, kaip pagrindinį savo gaminių privalumą, išskiria saugumą. „Neuron mobility“ įmonė užsiima elektrinių paspirtukų produkcija bei nuomos paslaugomis, taip pat teikia mobiliąją aplikaciją, su kuria galima prisijungti prie paspirtuko. Įmonė susikūrusi Singapūre, savo veiklą taip pat pradėjo vykdyti Australijoje, tačiau ateityje numatoma veiklą plėsti ir Europoje bei Šiaurės Amerikoje. Ši įmonė savo gaminamuose paspirtukuose naudoja pozicijos nustatymo sistemas, kas leidžia realizuoti geozonas įvairiose miesto vietose. Geozonų pagalba galima automatiškai reguliuoti ir apriboti paspirtuko greitį, ši sistema jau pradeda naudoti ir kitose nuomos paslaugomis užsiimančiose įmonėse, tokiose kaip, Lietuvoje veikiančioje „Bolt“. Geozonos taip pat padeda stebėti paspirtuko parkavimo vietą, kad ji netrukdytų eismui ar pėstiesiems. Taip pat „Neuron Mobility“ paspirtukuose yra naudojami judesio jutikliai, kurių pagalba galima aptikti nesaugius ir pavojingus manevrus. Šie manevrai yra: agresyvus vinguriavimas, staigus stabdymas slystant, važiavimas daugiau nei vienam, šuoliai nuo ir ant šaligatvio bortelio, važiavimas šaligatviu.



10 pav. "Neuron Mobility" įmonės identifikuoti pavojingi manevrai [22]

Taip pat šios įmonės paspirtukai turi tvirtinamus šalmus, kuriuos privalu dėvėti kelionės metu, dėl to prieš pradėdamas važiavimą, būtina atlikti asmenukės nuotrauką dėvint šalmą, programėlė atpažįsta ir patvirtina tai bei leidžia pradėti važiavimą. Dar viena priemonė skirta drausminti vairuotojus tai įmontuotas garsiakalbis, kuris praneša vairuotojui apie jo važiavimą, perspėja dėl per didelio greičio ar pavojingų manevrų atlikimą.

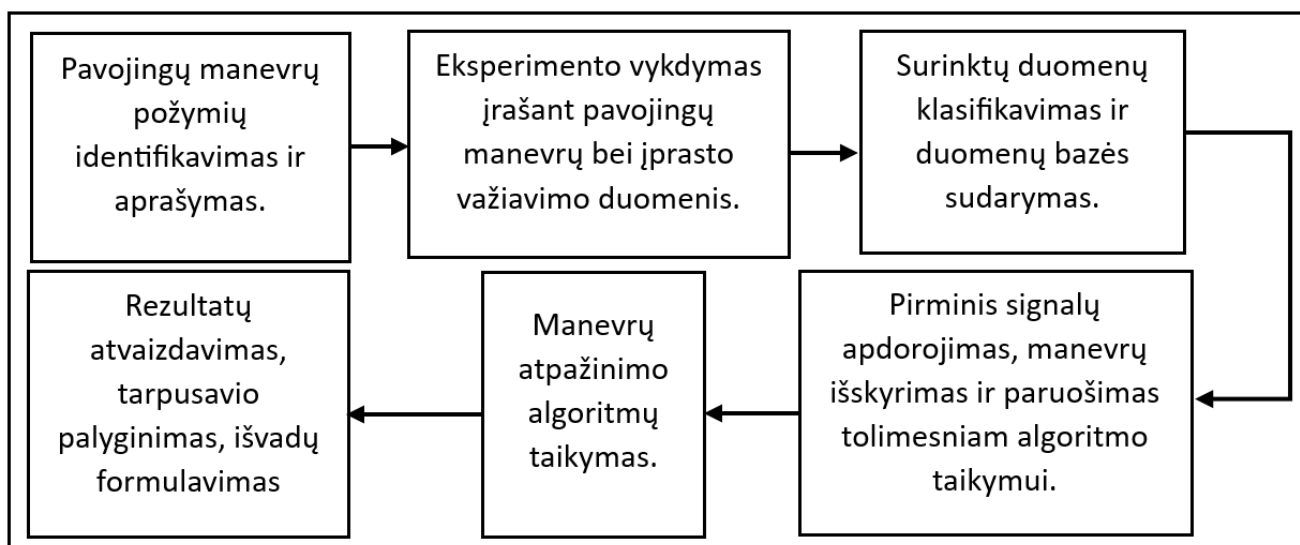
Nors įmonės puslapyje ir nebuvo gilinamasi į naudojamus algoritmus ar jutiklius, skirtus aptikti manevrus, tačiau aprašoma, ką galima laikyti pavojingais manevrais, kas padės tolimesniame tyrime identifikuoti pavojingus manevrus.

2. Metodinė tyrimo dalis

Šioje dalyje bus aptariama visa tyrimo eiga. Pagrindinės šio skyriaus dalys yra pavojingų manevrų išskyrimas ir detalus aprašymas, eksperimento modeliavimas ir vykdymas, duomenų bazės sudarymas, duomenų paruošimas tolimesniam apdorojimui, naudojamų manevrų identifikavimo algoritmų aprašymai ir testavimai.

2.1. Tyrimo struktūrinė schema

Žemiau pateikta visa tyrimo struktūrinė schema:



11 pav. Tyrimo struktūrinė schema

Vadovaujantis šia schema bus vykdomas visas tolimesnis tyrimas.

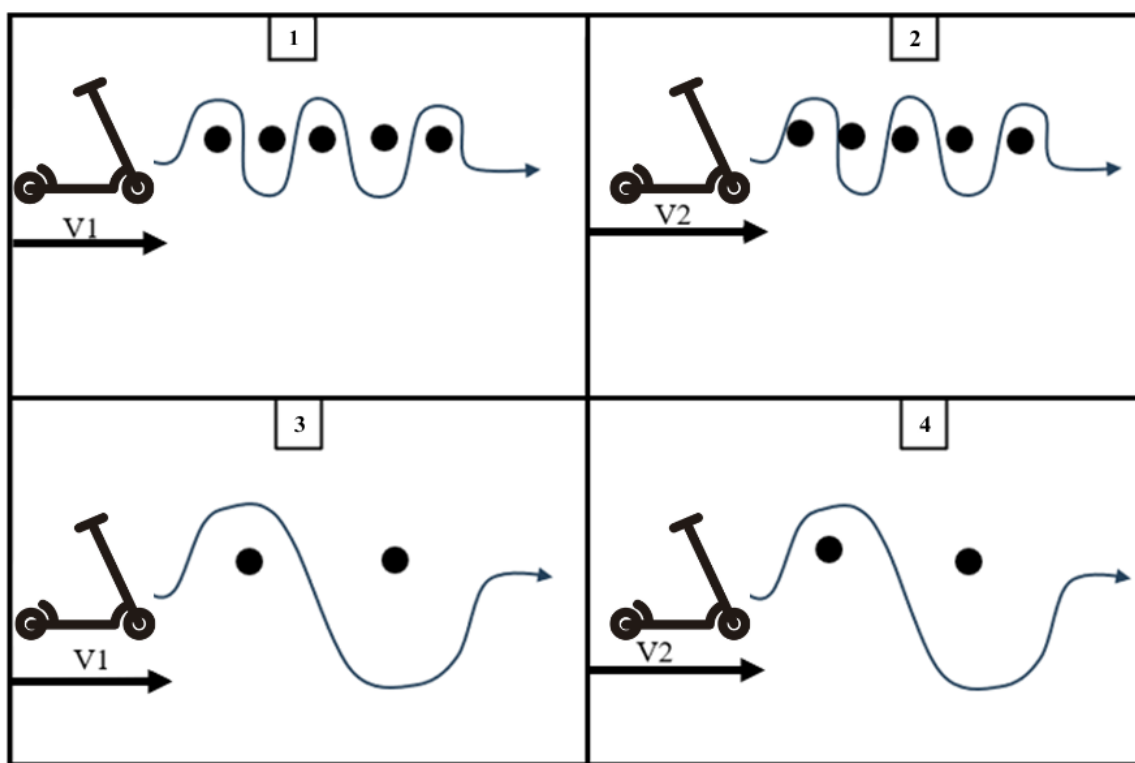
2.2. Pavojingų manevrų identifikavimas ir aprašymas

Kiekviena mikromobilumo priemonė yra skirtinga tiek savo konstrukcija, tiek veikimo principu, kadangi, dažniausiai sutinkama priemonė yra elektrinis paspirtukas, toliau darbe bus orientuojamasi į jo manevrus ir atliekami eksperimentai naudojantis šia priemone. Elektrinis paspirtukas turi du ratus, yra naudojamas stovimoje arba sėdimoje padėtyje. Abiem atvejais, mobilumo priemonės svorio centras yra ganėtinai aukštai nuo paviršiaus, o balanso išlaikymo dėsnis sako, kad didelės masės, didelio pagrindo paviršiaus ploto ir žemo masės centro objektas gali geriau išlaikyti balansą, kai yra veikiamas išorinių jėgų (pvz.: stūmimas, traukimas) [23]. Dėl šio dėsnio paspirtukas yra labiau nestabilus ir juo naudojantis yra lengviau išeiti iš balanso būsenos, vadinasi naudojantis paspirtuku yra svarbu stengtis išlaikyti balansą. Dėl to staigūs manevrai, tokie kaip staigūs posūkiai gali išbalansuoti priemonę ir vartotojui nesugebėjus išvairuoti, priemonė gali tapti nevaldoma ir yra galimybė pargriūti. Važiuojant vienodu greičiu ir įvertinant prieš save esančias kliūtis, aplinką, bei pėsčiuosius. Greitų posūkių ar manevrų atlikti neprireikia, atsakingo vartotojo važiavimas turėtų būti nuoseklus, pereinamieji procesai lėti. Jei vyksta stabdymas, jis neturėtų būti staigus, jei atliekamas posūkis, greitis turėtų sumažėti, posūkis atliekamas kuo didesniu lanku, jei priekyje esamą pėsčiujų, greitis turėtų būti parenkamas lėtesnis, važiavimas atliekamas atsitraukus nuo pėsčiujų. Taigi vieną iš pavojingų manevrų galima identifikuoti, kaip vinguriavimą tarp kliūčių, dažniausiu atveju,

pėsčiųjų, nes tai sukelia galimybę nesuvaldyti priemonės ir vartotojui kristi taip sužalojant pėsčiuosius, kitų turtą ar save patį. Galimi 4 šio manevro variantai:

1. važiuojama dideliu greičiu atliekant mažos amplitudės posūkius;
2. važiuojant dideliu greičiu atliekant didelės amplitudės posūkiu;
3. važiuojant mažu greičiu atliekant mažos amplitudės posūkius;
4. važiuojant mažu greičiu atliekant didelės amplitudės posūkius.

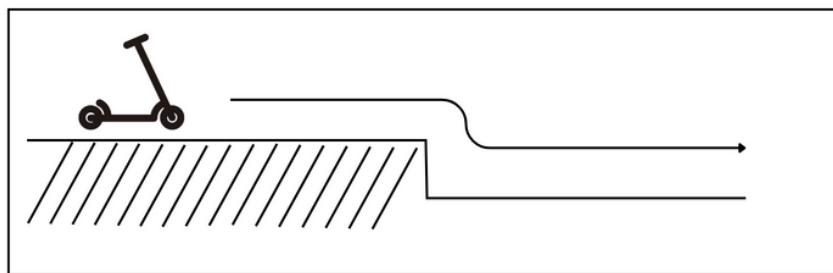
Visi manevro variantai išvardinti pagal pavojingumą, didelis greitis visais atvejais sukelia didesnę riziką atsitikti susidūrimui ar kritimui. Kitas veiksnys yra posūkių staigumas arba vinguriavimo atveju, amplitudė, staigūs posūkliai ar greitai keičiama posūkio pusė gali lengviau priversti prarasti balansą ir tapti nevaldomu. Atliekant lėtesnius ar didesnius posūkius patiriama mažiau jėgų, kurios kenkia vartotojui ir kraštutiniais atvejais gali priversti griūti. Visais atvejais atliekant posūkį lėtesniu greičiu yra saugiausias variantas [24]. Žemiau pateikta manevro atlikimo principinė schema, du svarbūs parametrai tai greitis V ir atstumas tarp kliūčių, kurias reikia apvažiuoti:



12 pav. Pavojingo manevro "Vinguriavimas" principinė schema

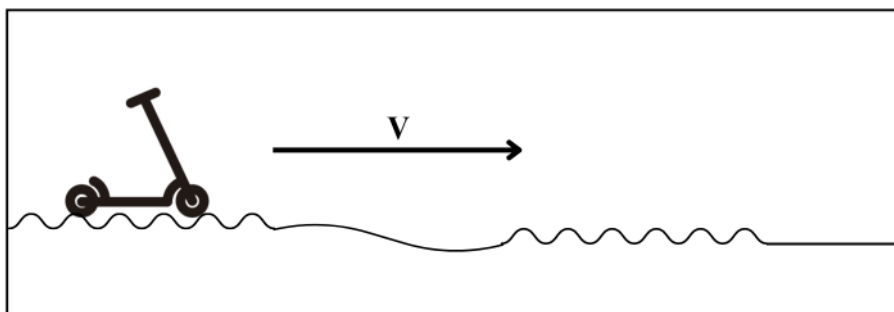
Nuo kliūčių atstumo priklausys manevro dažnis bei amplitudė, o nuo paspirtuko greičio - pavojingumas, taigi šiam manevrui užfiksuoti yra svarbu akcelerometro, giroskopo ir momentinio greičio duomenys.

Taip pat prie pavojingų manevrų galima priskirti ir šuolį nuo šaligatvio bortelio ant važiuojamosios dalies. Tai ne tik pažeidžia KET taisyklės, kurios draudžia kirsti kelią neįsitikinus, kad yra saugu, bet taip pat sukelia pavojų vairuotojams, kurie gali išsigąsti ar nepastebėti elektrine priemone važiuojančio ir neteisėtai kelią kertančio asmens.



13 pav. Pavojingio manevro "Šuolis nuo bortelio" principinė schema

Dar vienas nesaugus mikromobilumo priemonės naudojimo scenarijus galėtų būti važiavimas dideliu greičiu nelygiu keliu. Tai gali būti žvyrkelis, duobėtas kelias, kažkoks takelis ar tiesiog važiavimui nepritaikytas ruožas. Važiuojant tokiu keliu nėra būtina atlikti jokių papildomų manevrų, užtenka didelio greičio, kadangi paspirtukas patiria labai dideles vibracijas, ne vientisas kelias gali mėtyti paspirtuko ratus ir priversti vairuotoją griūti. Važiuojant prasta kelio danga visada svarbu įvertinti tinkamą greitį, o jei sąlygos neleidžia važiuoti, derėtų priemonę prasivesti ar prasinešti arba tiesiog vengti tokių kelio ruožų.



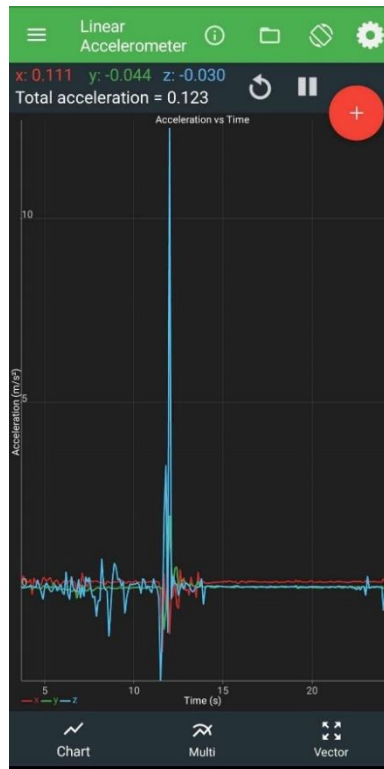
14 pav. Pavojingio manevro "Greitas važiavimas prasta danga" principinė schema

2.3. Eksperimento maketas

Sekančiame poskyryje bus aptariami naudoti maketai, taikyti duomenų surinkimo algoritmai ir eksperimento seka, skirta surinkti mikromobilumo priemonės duomenims, kurie bus naudojami tolimesniame tyrime.

2.3.1. Duomenų surinkimo sistema pasitelkiant išmanųjį telefoną

Pirminis duomenų surinkimas buvo atliekamas su išmaniuoju telefonu ir jame esančiais jutikliais. Buvo naudojama „*Physics Toolbox Suite*“ programėlė. Ši programėlė leidžia sekti ir įrašinėti vidinių telefono jutiklių duomenis, programėlėje galima sekti akcelerometro, giroskopo, inklinometro duomenis realiu laiku. Buvo naudojama „Xiaomi“ mi 10 išmanusis telefonas, kuris savyje turi įmontuotą „STMicroelectronics“ gamintojo LSM6DSO inercinį matavimo bloką, kuriuo buvo įrašinėjami pagreičio duomenys. Kaip mikromobilumo priemonė, buvo naudojama „Xiaomi“ mi scooter 1s elektrinis paspirtukas. Telefonas buvo montuojamas dvejose vietose: ant vairo ir ant kojos. Dvi vietos buvo parinktos tam, kadangi nuo jutiklio pozicijos priklauso ir surenkami duomenys, esant telefonui ant vairo akceleracijos bus matomos mažesnės amplitudės nei esant ant pagrindo ar ant kojos.



15 pav. „Physics Toolbox Suite“ programėlės vaizdas

Duomenys buvo rašomi .csv failo pavidale. Šis eksperimento stendas leido greitai ir nesunkiai prisirinkti važiavimo paspirtuku duomenų, kuriuos galima pradėti nagrinėti. Su šiuo maketu buvo atlikta keletą važiavimų su įvairiais manevrais: šuolis nuo šaligatvio bortelio, greitas stabdymas, vinguriavimas, paprastas ir ramus važiavimas. Visi šie manevrai buvo įrašomi telefono pozicijai esant ant vairo, tiek pritvirtinam ant kojos. Duomenys buvo įrašomi m/s^2 formatu:

	A	B	C	D	E
1	time	ax (m/s ²)	ay (m/s ²)	az (m/s ²)	aT (m/s ²)
2	0.002761	0.1622	0.1733	0.2686	0.358
3	0.003402	-0.2415	0.1148	0.3353	0.429
4	0.006543	-0.3941	0.0111	0.4899	0.629
5	0.015485	-0.3557	-0.0256	0.7424	0.824
6	0.017449	-0.3628	-0.3226	0.8156	0.949
7	0.021884	-0.5158	-0.423	0.5306	0.852
8	0.026783	-0.3521	-0.1165	0.2661	0.456
9	0.031126	-0.3618	-0.1071	0.5427	0.661
10	0.03674	-0.5287	-0.3945	0.5105	0.834
11	0.04155	-0.527	-0.0242	0.4469	0.691
12	0.046342	-0.2163	0.2812	0.7136	0.797
13	0.051072	-0.2814	-0.0807	0.1512	0.329
14	0.056881	-0.0962	0.2609	-0.4616	0.539
15	0.061037	-0.5291	-1.4366	2.5112	2.941
16	0.066663	-1.316	-2.237	6.1009	6.63
17	0.071647	-0.3235	1.082	4.1007	4.253
18	0.082412	0.2173	3.0771	-0.4277	3.114
19	0.083426	0.2066	2.9013	-5.7675	6.459
20	0.086291	0.3361	2.005	-8.1234	8.374
21	0.092155	0.898	0.7312	-4.5104	4.657
22	0.096945	0.1078	0.8502	-3.2924	3.402
23	0.10118	0.8936	-0.3139	1.5091	1.782
24	0.105807	1.6285	-1.8285	2.6648	3.619
25	0.110785	1.088	-2.317	-0.4864	2.605
26	0.11651	0.8598	-1.7984	-2.2131	2.979
27	0.121939	0.9796	-1.0946	-1.3284	1.98

16 pav. Įrašytų duomenų pavyzdys

Iš gautų laiko atžymų buvo nustatyta, kad diskretizavimo dažnis nėra pastovus, jis kinta atitinkamai nuo akceleracijos. Kuo didesnis akceleracijos pokytis, tuo labiau diskretizavimo dažnis didėjo, esant lėtam kitimui, diskretizavimo dažnis sumažėdavo kelis kartus, taigi gauti duomenys buvo diskretizuoti nestabiliu dažniu, tai didelė problema norint toliau apdoroti duomenis, taigi būtina perdiskretizuoti visą duomenų failą, taip pat jutiklio jautrumas buvo nežinomos skalės, programėlėje jo pasirinkti negalima, taigi nebuvo aišku ar jis išlieka stabilus viso įrašo metu.

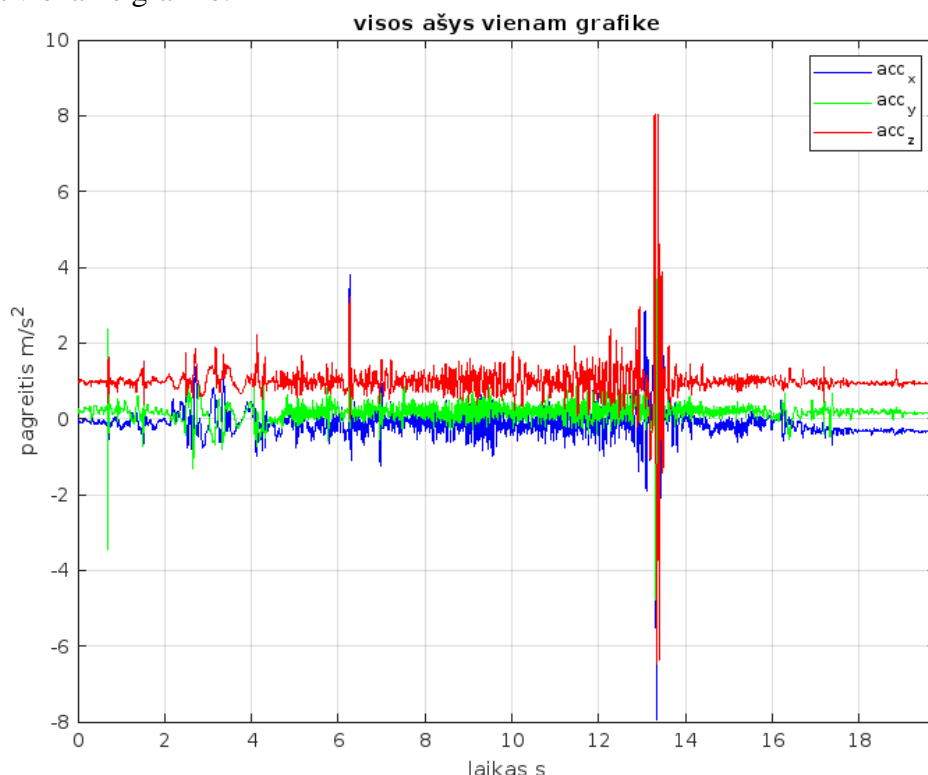
Turimi duomenys buvo perkelti į „Matlab“ programinį paketą, kuriame bus vystomas tyrimo algoritmas. Pirmas žingsnis buvo duomenų perdiskretizavimas. Tam buvo paskaičiuotas vidutinis diskretizavimo dažnis, skaičiuojant laiko atžymų diferencialą - iš N+1 atskaitos atimant N atskaitą. Taip buvo suformuotas skirtumų masyvas, tada iš šio masyvo paskaičiuotas aritmetinis vidurkis pagal šią formulę [25]:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^{n-1} x_i \quad (2)$$

Galiausiai turimas vidutinis periodas buvo paverstas į dažnį pagal šią formulę:

$$f = \frac{1}{T} \quad (3)$$

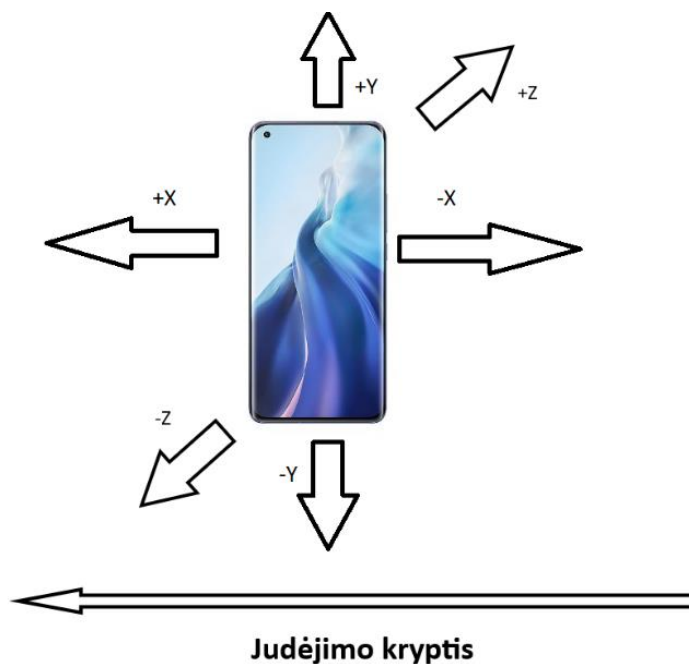
Taip gautas vidutinis viso įrašo dažnis. Žinant tai, galima atlikti perdiskretizavimą arčiausiam sveikam skaičiui, tai buvo 500 Hz. Pasitelkiant *resample* funkciją, buvo nustatytas visų trijų ašių (X, Y ir Z) diskretizavimo dažnis lygus 500. Toliau atvaizduotas šuolio nuo bortelio manevras, visas tris ašis sudedant viename grafike:



17 pav. Gautas manevras laike

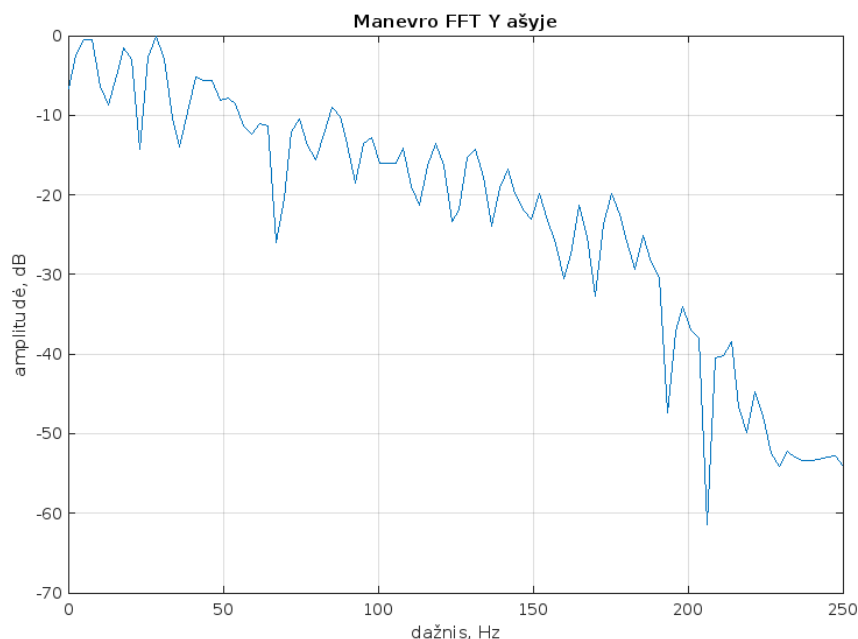
Kiekvieną ašį paanalizavus iš arčiau buvo nustatyta, kad manevras prasideda 13,2 s ir baigiasi 13,4 s, taigi buvo iškirpta ši signalo dalis. Kadangi naudojamas įrašas buvo sudaromas, kai telefonas buvo pritvirtintas vartotojo prie kojos, reikėjo nustatyti, kuri ašis yra lygiagreti judėjimo pusei ir kurios statmenos. Tai buvo atlikta stebint telefono akcelerometro parodymus, jam esant paguldytam ant

stalo. Kadangi akcelerometras matuoja nuolatinę Žemės traukos jėga, taigi ta ašis, kuri bus nustatyta į žemės centrą rodys artimą 1 G dydžiui. Taip buvo nustatytos ašių vektorių kryptys pagal judėjimo kryptį:



18 pav. Telefono pozicija eksperimento metu

Šuolio nuo bortelio manevro metu tikslinga stebėti Y ašį, kadangi ji patirs didžiausias akceleracijas, taigi Y ašies manevro iškarpai buvo paskaičiuota FFT. Greitoji Furjė Transformacija (arba FFT) yra efektyvus ir optimizuotas metodas skaičiuoti Diskretinę Furjė Transformaciją (arba DFT), DFT yra Furjė transformacija skirta diskretiems signalams, kuri leidžia signalą išskaidyti į dažnines komponentes [26]. Su šiuo algoritmu išskaidžius signalą į dažnines dedamąsias galima nustatyti, kuriame dažnių ruože yra manevras. Taigi pasinaudojus *fft* funkcija buvo gautas toks manevro spektras:



19 pav. Šuolio nuo bortelio Y ašies signalo FFT rezultatas

Iš šio grafiko galima teigti, kad manevras patenka į labai žemų dažnių ruožą, didžiausios dedamosios yra nuo 0 iki 32 Hz. Ir tai neprieštarauja logikai, nes visi mechaniniai manevrai ar virpesiai, kuriuos fiksuoja jutiklis, turėtų būti žemadažniai, kadangi važiavimas paspirtuku, šuolis ar greitas posūkis yra ganėtinai lėtas procesas, paprastai bent koks žmogaus atliekamas manevras truks sekundžių eilėje, dėl fizinių galimybių. Taigi iš šio pirminio eksperimento buvo nustatyta toki manevrų parametrai, kaip dažnių juosta, parinkti pakankamą diskretizavimo dažnį bei pradėtas algoritmo kūrimą, tačiau tolimesnei tyrimo eigai reikėjo sukurti stabilią ir laisvai valdomą sistemą, kuri leistų įrašyti ne tik akcelerometro duomenis, bet ir giroskopo, taip pat būtų aktualu žinoti momentinį greitį. Tuo tikslu buvo pradėta kurti lanksti įterptinė sistema skirta duomenų surinkimui.

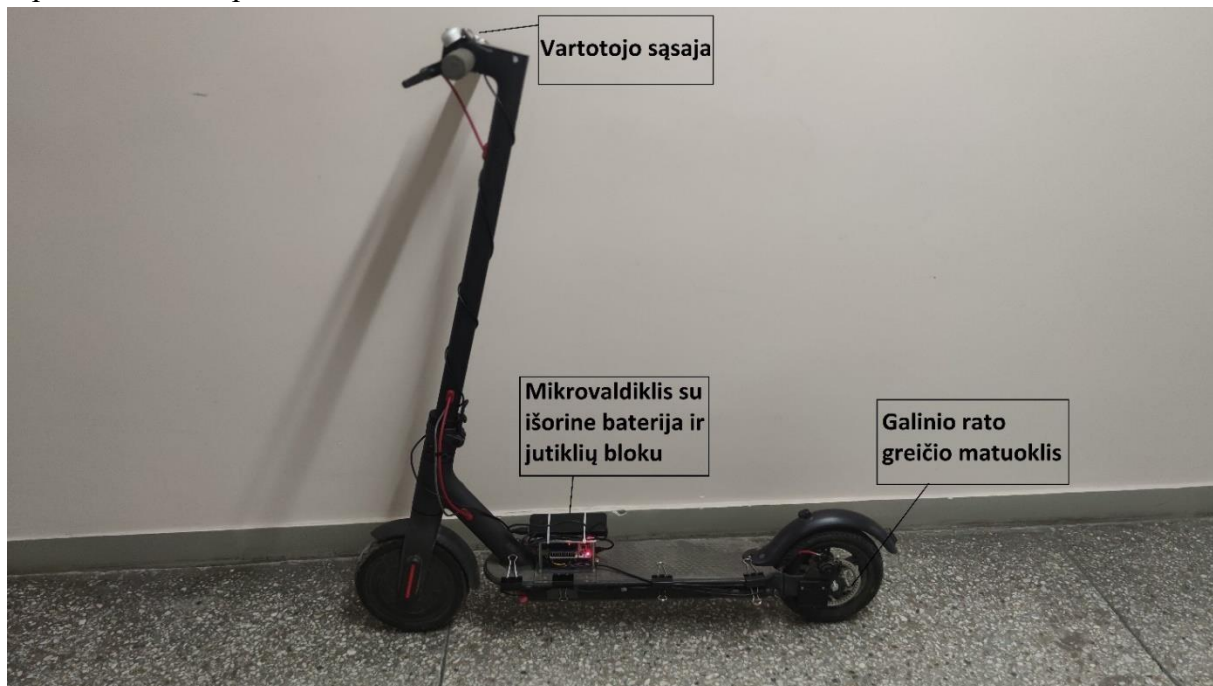
2.3.2. Duomenų surinkimo sistema pasitelkiant įterptinę sistemą

Sekantis eksperimento prototipas buvo kurtas pasitelkiant mikrovaldiklį ir reikalingus jutiklius. Iš jau žinomų signalo požymių ir charakteristikų buvo išskelti sistemos reikalavimai:

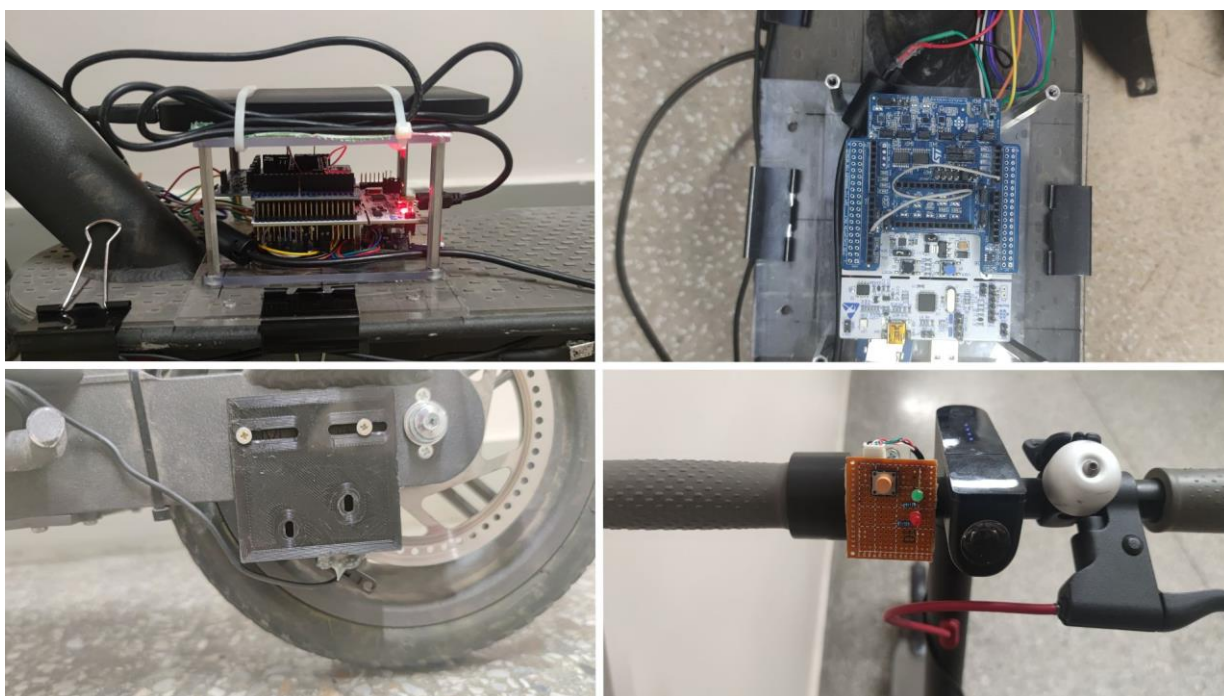
- sistema turi gebėti surinkti akcelerometro, giroskopo ir momentinio greičio duomenis;
- duomenys turėtų būti saugomi patogioje ir lengvai perkeliamoje į asmeninį kompiuterį, formoje;
- sistemos diskretizavimo dažnis turėtų būti bent 100 Hz, kad būtų užfiksuoti manevro dažniai;
- sistema turėtų turėti minimalią vartotojo sąsają, kuri leistų pradėti duomenų įrašymą bei indikuotų apie vykstantį įrašymą;
- sistema turi būti lengvai tvirtinama.

Taigi turint šiuos reikalavimus, buvo pradėta ieškoti komponentų, kurie padėtų tai įgyvendinti. Akcelerometro ir giroskopo jutikliams buvo nuspręsta naudoti tą patį LSM6DSO jutiklį, kadangi tai ir akcelerometras ir giroskopas viename, taip pat jo diskretizavimo dažniai yra pasirenkami plačiame ruože: 1,6 Hz, 12,5 Hz, 26 Hz, 52 Hz, 104 Hz, 208 Hz, 416 Hz, 833 Hz, 1.66 KHz, 3.33 KHz, 6.66 KHz. Toks platus pasirinkimo ruožas leis lanksčiau surinkti duomenis, bei parinkti optimaliausią variantą. Taip pat šis jutiklis turi keletą skalės nustatymų tiek giroskopui, tiek akcelerometrai, kas leis paderinti gaunamus duomenis ir pasirinkti tiksliausią skalę neapribojant duomenų amplitudės. Šį jutiklį galima rasti „STMicroelectronics“ gamintojo vystymo plokštėje X-NUCLEO-IKS01A3, kuri leidžia be papildomų sunkumų ją naudoti ant daugelio egzistuojančių mikrovaldiklių mokomųjų plokščių. Taip pat ši plokštė leis nesunkiai derinti kodą, jeigu kuriant sistemos algoritmą atsirastų klaidų. Ši plokštė yra suderinama su bent kokia „Arduino“ tipo plokšte arba su ST gamintojo „Nucleo“ plokštėmis, kurios yra specialios mokomosios plokštės, skirtos tyrinėti tam tikro mikrovaldiklio galimybes ir periferiją. Buvo nuspręsta naudoti NUCLEO-F401RE mokomąją plokštę, kuri turi integruotą STM32F401RE mikrovaldiklį. Šis mikrovaldiklis yra F4 serijos, todėl jis yra priskiriamas našioms ir greitoms mikrovaldikliams, gebantiems atlikti skaitmeninio signalų apdorojimo užduotis. Kadangi specialių reikalavimų mikrovaldikliui nebuvo iškelta, išskyrus tai, kad jis turėtų turėti GPIO periferiją ir standartines nuoseklias sąsajas (pvz: I2C, SPI, UART), buvo nuspręsta naudoti būtent šią plokštę, su kuria yra puikiai suderinta jutiklių plokštė. Vartotojo sąsajai bus naudojami 2 spalvų šviestukai, raudonas - signalizuoti apie gedimą ir žalias - signalizuoti, kad vyksta duomenų nuskaitymas ir įrašymas. Taip pat bus naudojamas vartotojo mygtukas, kurį paspaudus prasidės jutiklio duomenų nuskaitymas ir įrašymas. Greičio matuokliui bus naudojamas lazerinis impulsų jutiklis, kuris bus montuojamas ant galinio rato stabdymo disko, jame yra vienodais atstumais išmuštos skylės, kurias galima naudoti impulsams aptikti ir sekti rato pasisukimo kampą ir greitį. Visa sistema bus maitinama išorine baterija, taip pat bus naudojami ekranuoti kabeliai išnešti mygtukui ir šviestukams šalia vairo, taip bus galima važiuojant ir prieš atliekant manevrą pradėti įrašymą ir matyti ar įrašymas vyksta. Taip pat ekranuoto kabelio pagalba impulsai kelias nuo galinio

rato iki plokštės. Paskutinė dalis tai bus duomenų įrašymas. Kadangi juos perduoti terminalo pagalba į asmeninį kompiuterį nebūtų patogu ir patikima, nes duomenų srautas bus didelis, o ir asmeninio kompiuterio nebūtų galima tvirtinti ant paspirtuko, buvo nuspręsta nuskaitytus duomenis įrašinėti į SD atminties kortelę, kadangi tai nesudėtingas metodas, užtenka paprastos SPI sąsajos komunikuoti su kortele ir tai leis lengvai duomenis perkelti į asmeninį kompiuterį norimu pavidalu. Mikrovaldiklis su jutiklių plokšte buvo patalpinti į specialiai išpjautą korpusą, kuris pritvirtintas metaliniais spaustukais prie paspirtuko pagrindo, vartotojo sąsają ir greičio jutiklį tvirtintas sumodeliuotomis ir 3D spausdintuvu atspausdintomis detalėmis.



20 pav. Galutinai surinkta eksperimento sistema



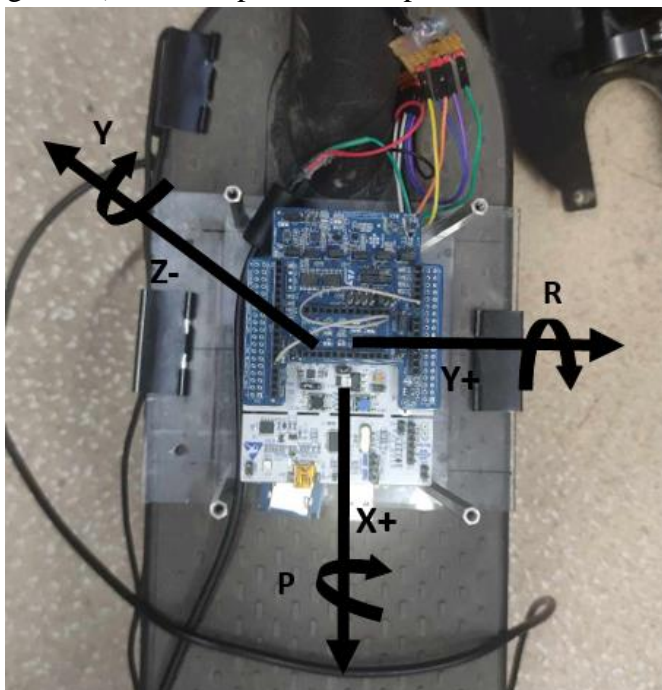
21 pav. Eksperimento stendo komponentai

Pagal jutiklio poziciją ir jo aprašą buvo nustatyta, kad akcelerometro vektorių vertės bus tokios:

- X ašis turės neigiamą vertę judėjimo kryptimi ir teigiamą vertę prieš judėjimo kryptį;
- Y ašis turės neigiamą vertę kairėje paspirtuko pusėje ir teigiamą dešinėje paspirtuko pusėje;
- Z ašis turės neigiamą vertę statmenai aukštyn paspirtukui ir teigiamą vertę statmenai žemyn.

Giroskopas fiksuos šiuos orientacijos pokyčius:

- P orientacija (angl. *Pitch*) fiksuos apsisukimus aplink X akcelerometro ašį;
- R orientacija (angl. *Roll*) fiksuos apsisukimus aplink Y akcelerometro ašį;
- Y orientacija (angl. *Yaw*) fiksuos apsisukimus aplink Z akcelerometro ašį.



22 pav. Jutiklio vektorių vizualizacija

Surinkus visą aparatinę dalį buvo pereita prie algoritmo kūrimo.

2.3.3. Eksperimento stendo algoritmas duomenų surinkimui

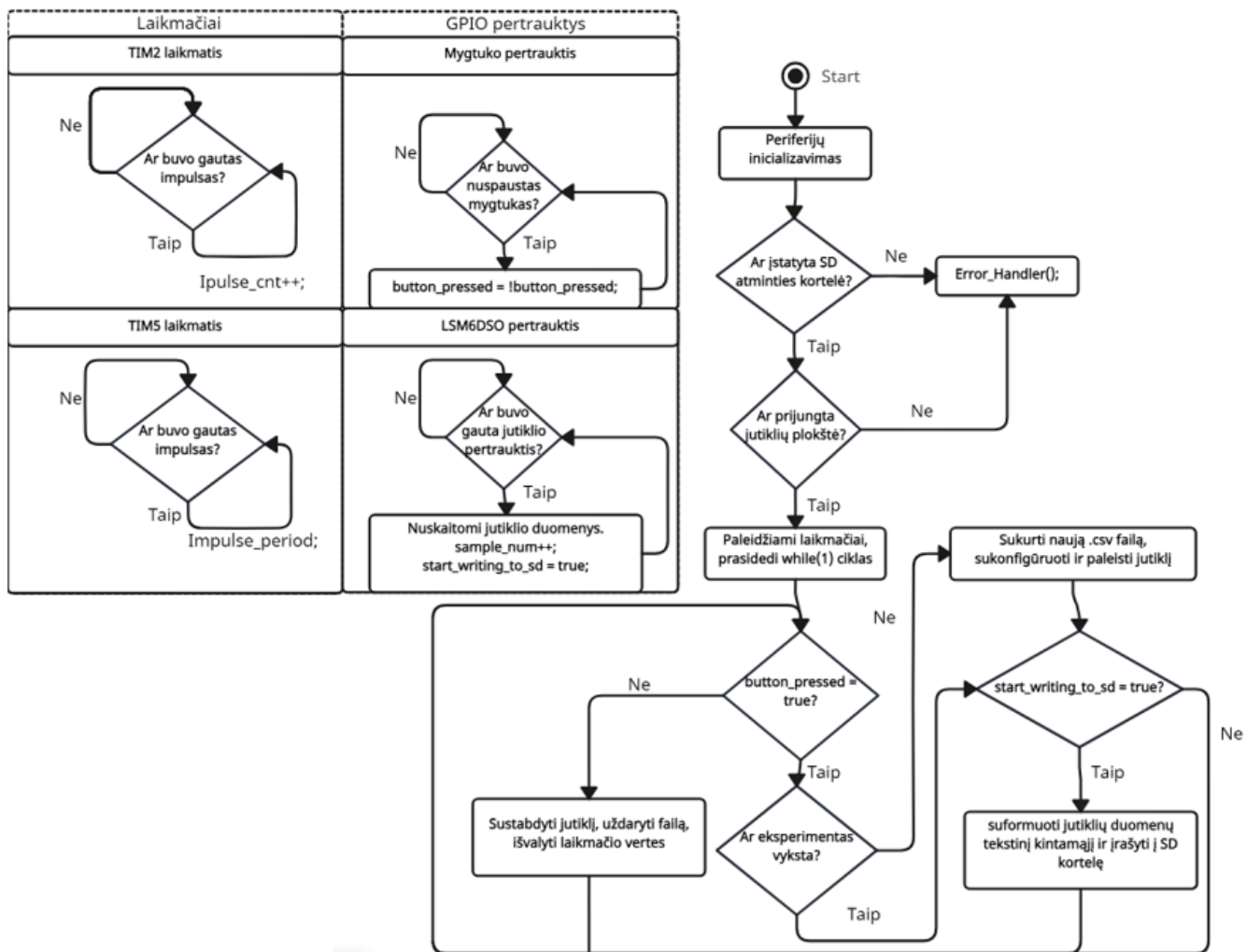
Šiame etape buvo realizuotas ir sukurtas algoritmas, kuris geba nuskaityti jutiklių duomenis ir juos įrašyti į SD kortelę pasirinktu diskretizavimo dažniu. Pradžioje buvo įgyvendinta sąsaja ir komunikacija su IMU bloku, kadangi tai pagrindinis jutiklis. Jutiklis veikia dviem pagrindiniais režimais: vientiso duomenų skaitymo arba FIFO buferio principo, kuomet duomenys rašomi į atmintį ir prikaupus tam tikrą kiekį, duodamas pranešimas, kad duomenys yra paruošti. Kadangi yra svarbus duomenų eiliškumas ir naujumas, o taipogi, apart duomenų skaitymo ir rašymo, algoritmas daugiau nieko nevykdys, buvo nuspręsta naudoti paprastą nuskaitymo metodą. Jutikliui kaskart diskretizavus naują vertę bus išstatomas pertraukties signalas INT2 išvade ir mikrovaldiklis gavęs pertrauktį nuskaitys duomenis. Kadangi diskretizavimo dažnis yra nustatomas jutikliui, o ne diktuojamas mikrovaldiklio, jutiklis ir sinchronizuos visą sistemą, jis praneš, kada galima skaityti duomenis ir juos įrašyti. Taip pat dėl šios priežasties duomenų skaitymas tampa svarbiausia visos sistemos dalimi, kadangi pražiopsojus pertrauktį ar nenuskaičius laiku signalo, duomenys bus prarasti ir gaunama jau nauja vertė. Taigi akceleracijos ir giroskopo duomenys bus nuskaityti nustatytu fiksuotu jutiklyje esančiu diskretizavimo dažniu ir nuskaityti duomenys įrašyti į SD kortelę. Pasitelkiant I2C sąsaja buvo sukonfigūruotas jutiklis ir sukurtos nesudėtingos funkcijos jo valdymui, paleidimui ir sustabdymui.

Toliau reikėjo sukurti momentinio greičio matuoklio valdymą. Kadangi iš jutiklio bus gaunami impulsai, aukštas ir žemas lygis, tai impulsus galima fiksuoti, kaip pertrauktis ir skaičiuoti jų kiekį, tačiau tam prireiktų papildomo laikmačio, apskaičiuoti jų dažnį, kas apkrautų visą algoritmą ir apribotų ir taip siaurą langą, per kurį privalu įrašyti duomenis, taip pat, kad išvengtų pertraukčių prioritetų dėliojimo ir dėl didesnės greitaveikos, buvo nuspręsta pasinaudoti aparatiniu laikmačiu momentinio greičio realizavimui. Bus naudojami du laikmačiai: TIM2 ir TIM5. TIM2 laikmatis bus taktuojamas ateinančiais impulsais ir tiesiog skaičiuos jų kiekį, tokiu būdu bus galima pasiimti surinktų impulsų kiekį ir nustatyti nuvažiuotą atstumą. TIM5 laikmatis naudos specialią funkciją PWM_in, kuri skirta nustatyti PWM signalo dažnį ir skverbtį, taigi bus tik dažnio nustatymo funkcija, kuri įrašys skaitiklio vertę, ties kuria bus priimtas impulsas ir ją išvalys. Skaitiklis didės tam tikru, nustatytu dažniu, taigi žinant skaitiklio inkrementavimo dažnį ir jo vertę, galima apskaičiuoti impulsų periodą, ką taip pat galima paversti į momentinį greitį. Abu šie procesai vyks aparatiškai ir nereikalaus procesoriaus ciklų juos įgyvendinti, tereikės nuskaityti registrų vertes prieš pat rašymą į kortelę.

Dar vienas svarbus etapas yra rašymas į kortelę. Tam bus naudojama SPI sąsaja komunikuoti su kortele, taip pat reikės metodo, kaip sukurti ir valdyti failus, kadangi eksperimentai bus vykdomi po kelis kartus, taigi būtų patogų turėti skirtingus failus skirtingiems eksperimentams. Dėl šių priežasčių buvo nuspręsta naudoti FATFS failų tvarkymo biblioteką, kuri implementuoja funkcijas skirtas nuskaityti SD kortelės turinį, parametrus, sukurti failus, juos rašyti ir atlikti įvairias valdymo funkcijas, šiam darbui reikalingos yra failo sukūrimo, atidarymo, rašymo bei uždarymo funkcijos.

Paskutinis etapas - tai vartotojo sąsaja, jai reikalingi 3 GPIO išvadai: 2 LED valymui ir 1 mygtuko lygio nuskaitymui. Mygtuko lygis bus priimamas pertrauktyje ir bus aukščiausio prioriteto, kadangi nuo jo paspaudimo priklausys, kada prasidės ir pasibaigs testas.

Turint visas reikiamas algoritmo dalis teliko viską apjungti. Taigi pradžioje bus inicializuojamos visos reikiamos periferijos, tada nuskaitoma SD kortelė, jos dydis. Jei SD kortelė nebus patalpinta, raudonos spalvos LED diodas pradės mirgėti ir kodas pereis į klaidos funkciją iš kurios neišeis, tai apsaugos priemonė, kuri įsitikins, kad kortelė teisingai suformatuota ir patalpinta. Toliau vyks jutiklio *Who_Am_I* registro nuskaitymas, tai registras, kurį nuskaičius bus gaunama ta pati vertė kiekvieną kartą, tai dar viena apsaugos priemonė skirta įsitikinti, kad jutiklis prijungtas ir komunikacija veikia. Toliau vyks laikmačių paleidimas, kurie dirbs viso algoritmo metu. *While(1)* cikle bus tikrinama mygtuko vėliava *button_pressed*, jeigu jos būseną bus *true*, bus tikrinama - eksperimentas tik prasidėjo ar jis jau vyksta, jei tik prasidėjo, bus sukuriamas failas su eksperimento numeriu, kuris padidės kaskart pradėjus eksperimentą, failas bus kuriamas .csv formatu, tai patogus formatas duomenų rašymui, kadangi jis automatiškai duomenis atskiria į stulpelius. Galiausiai bus išvalomos laikmačių registrų vertės, kad jos pradėtų kisti nuo 0 ir paleidžiamas LSM6DSO jutiklis. Jei eksperimentas jau vyksta, bus laukiama vėliavos *start_writing_to_sd* pasikeitimo, tada bus nuskaitomos laikmačių vertės, paimamos jutiklio vertės ir suformuojamas kintamasis su visais duomenų laukais viename tekstiniame pavidale, tada šis kintamasis įrašomas į SD kortelę ir gražinama *start_writing_to_sd* kintamojo vertė į *false*. Jei mygtuko kintamojo būseną pasikeistų į *false*, būtų sustabdomas jutiklis ir išvalomi visi duomenų kintamieji. Pertrauktyse bus laukiama mygtuko nuspaudimo, kas pakeis *button_pressed* kintamojo reikšmę arba jutiklio LSM6DSO duomenų paketo impulso pertraukties, kas reikš apie paruoštus duomenis, kuriuos nuskaitys ir pakeis *start_writing_to_sd* kintamojo reikšmę. Algoritmo struktūrinė diagrama pateikta žemiau:



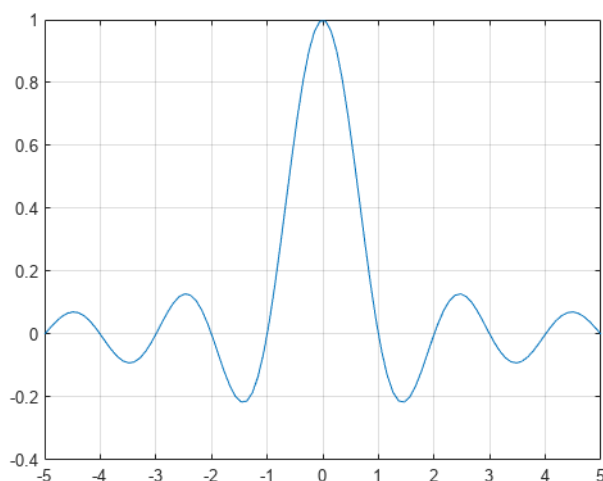
23 pav. Eksperimento stendo algoritmas

2.4. Duomenų bazės sudarymas

Sekantis žingsnis buvo realių duomenų rinkimas. Šiame etape svarbu iš anksto numatyti, kokie eksperimentai bus vykdomi ir kokie manevrai, kiekvieno važiavimo metu, bus atliekami, kadangi iš neapdorotų duomenų yra sunku pasakyti, koks manevras yra daromas. Kiekvienas eksperimentas bus atliekamas keletą kartų, kad būtų surinktas pakankamai didelis duomenų rinkinys ir iš jo būtų galima ištraukti požymius ar paskaičiuoti statistinius dydžius, taip pat kiekvienas eksperimentas bus klasifikuojamas, atlikus pakankamą kiekį manevrų, įrašai bus talpinami asmeniniame kompiuteryje, aplankuose, kurie turės prasmingus pavadinimus. Pirmi eksperimentai bus pastovaus greičio važiavimai įvairių kelio dangų erdvėse. Iš šių duomenų bus galima nustatyti ar sistema tvarkingai veikia, ar fiksuojamas greitis yra panašus į paspirtuko rodomą momentinį greitį. Toliau buvo pereita prie tyrimui aktualių manevrų fiksavimo.

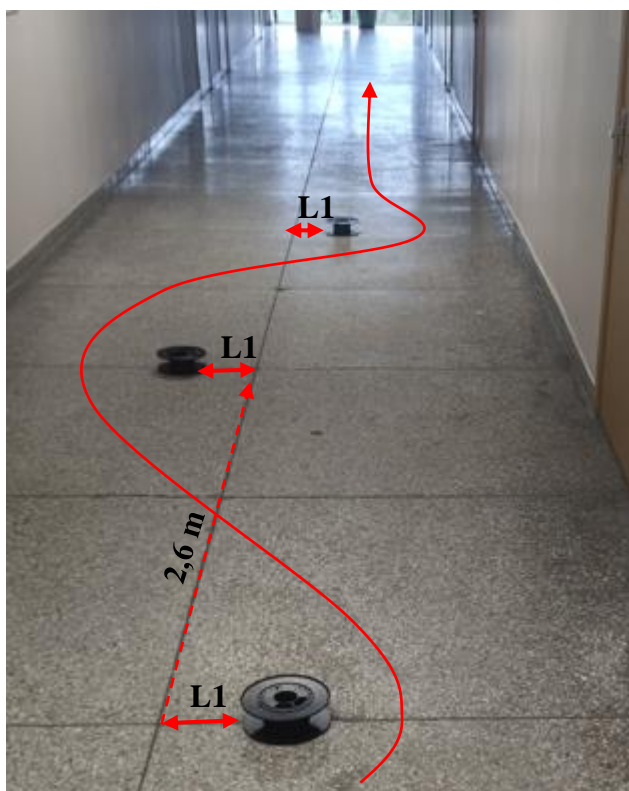
Iš pradžių reikėjo nustatyti ribą, kada manevras yra pavojingas, kada ne. Staigus stabdymas gali reikšti neatidų vairuotoją, o tolygus stabdymas visada turėtų būti traktuojamas, kaip saugus manevras, staigus kliūtis apvažiavimas taip pat gali reikšti nesaugų ar chuliganišką vairuotoją, tačiau lėtas posūkis galbūt yra posūkis ar kliūtis apvažiavimas. Taigi kiekvienas iš identifikuotų pavojingų manevrų turės būti eksperimentiškai išbandytas ir subjektyviai nustatyta jo pavojingumo riba. Šuoliui nuo bortelio ar važiavimo netinkama danga eksperimentams pavojingumo ar ne-pavojingumo ribos nustatyti nereikės, kadangi šie manevrai iškart yra netinkami ir keliantys pavojų, nesvarbu ar šuolis bus nuo mažo bortelio ar didelio, aktualu bus tik duomenys ir jų vertės atliekant šiuos manevrus.

Tačiau tokie manevrai, kaip apvažiavimas ar vinguriavimas gali būti ir nepavojingi, jei tai nedideli ir lėti manevrai, jie pavojaus nekelia nei vairuotojui, nei kitiems eismo dalyviams. Taip pat aktualu žinoti ir fizines galimybes atlikti manevrą, neverta ieškoti manevrų, jei jų atlikti neįmanoma, yra per daug rizikinga ar gali būti destruktivu. Norint nustatyti pavojingumo ribą vinguriavimo ir apvažiavimo manevrui buvo atliekami du eksperimentai, kurių parametrai buvo gerai žinomi ir sekami, siekiant kaskart išgauti atsikartojamumą ir pašalinti, kuo daugiau neapibrėžties iš eksperimento. Pirmas eksperimentas buvo 3 kliūčių apvažiavimas, taip sudarant panašų į *Sinc* formos signalą.



24 pav. Sinc formos signalas [27]

Eksperimentas vyks važiuojant tiesiai, tada apvažiuojant pirma kliūtį per apačią, apvažiuojama antra kliūtis per viršų, ir galiausiai, trečią kliūtį per apačią ir grįžtama į pradinę trajektoriją. Eksperimentas buvo atliekamas patalpoje, ilgame koridoriuje, taigi kelio danga lygus grindinys, tarp grindinio plokščių yra 1,3 m atstumas, taigi šis dydis naudojamas, kaip atžymos dydis, atidėti kliūtims. Pirminių važiavimų metu buvo nustatyta, kad 12 km/h greičiu apvažiuoti kliūtis pavyksta, kai jos atidėtos 2,6 m atstumu viena nuo kitos, šis dydis parametrizuoja manevro trukmę, norint parametrizuoti amplitudę, reikės keisti 1 ir 3 kliūties atstumą iki 2 kliūties. Derinant šį dydį buvo nustatomas ribinis manevras, kai atlikti tokį apvažiavimą dar realu, tačiau yra pavojingą ir yra nesaugu. Per didelės amplitudės posūkis gali išvesti iš balanso vairuotoją, padaryti mikromobilumo priemonę nestabilią ar nevaldomą, taigi suradus manevrą, kai apvažiavimo sėkmingai padaryti nepavyks arba pavyks retai, galima teigti, kad ribinė vertė buvo prieš tai naudotas atstumas tarp kliūčių ir ją galima naudoti, kaip ribinį pavojingą ir dar realų manevrą. Palengvinti manevro radimą duomenyse, taip pat buvo naudojama atžymos fizinis taškas važiuojamoje trajektorijoje, ant grindinio pritvirtinta ilga organinio stiklo žyma, kuri buvo pervažiujama prieš pat atliekant manevrą, tai sukėlė akcelerometro Z ašies duomenų šuolį ir jį suradus galima nustatyti, kurioje tiksliai vietoje prasideda manevras. Eksperimento schema pateikta žemiau:



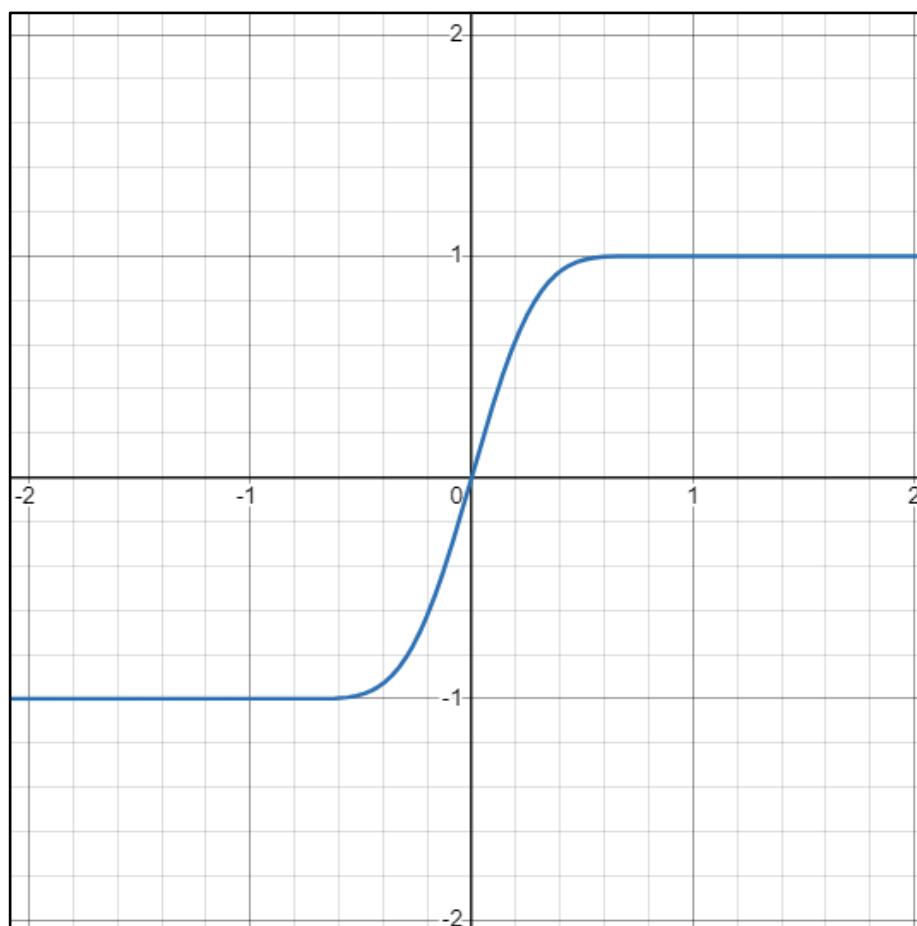
25 pav. Manevro schema

Keičiant L dydį, kartu keičiasi ir manevro amplitudė. Buvo atlikti matavimai prie šių manevro amplitudės dydžių (L):

- 10 cm ;
- 15 cm;
- 20 cm;
- 25 cm;
- 30 cm;
- 35 cm.

Atliekant 10, 15, 20 cm amplitudės eksperimentus, nebuvo susidurta su sunkumais, tačiau ties 25 cm amplitudė pavykdavo ne visi manevrai, tai reiškia, dėl ne idealiai pastovaus greičio, kartais manevras buvo atliekamas su šiek tiek per dideliu greičiu, ko pasėkoje, jis nesigaudavo gražus ir nuoseklus. Tai buvo pirmas požymis, kad buvo artėjama link ribinės amplitudės. Ties 30 cm amplitudė nepavykę manevrai padažnėjo, taip pat, kai kurie manevrai pasibaigdavo kojos atsispyrimu nuo grindinio, buvo keletas eksperimentų, kai paskutinė kliūtis buvo kliudoma. Taip pat buvo pastebėta, kad atliekant posūkį buvo intuityviai ir pasąmoniškai atleidžiama greičio valdymo rankenėlė, kas taip pat signalizuoja apie manevro pavojingumą. Ties 35 cm riba buvo pasiekta per sudėtingo manevro riba, kadangi dažniausiai manevrai užsibaigdavo 3 kliūtis kliudymu arba nepilnai pabaigtu manevru, tik labai maža dalis manevrų pavyko, taigi iš šio eksperimento buvo nustatyta, kad ties 30 cm amplitudė gaunamas ribinis manevras, kurio dar galima ieškoti, aukštesnės amplitudės manevro, turimomis sąlygomis, atlikti nepavyko taigi teoriškai jo sutikti signale nepavyks. Buvo atlikta dar keletą važiavimų ties 30 cm riba ir duomenys išsaugoti.

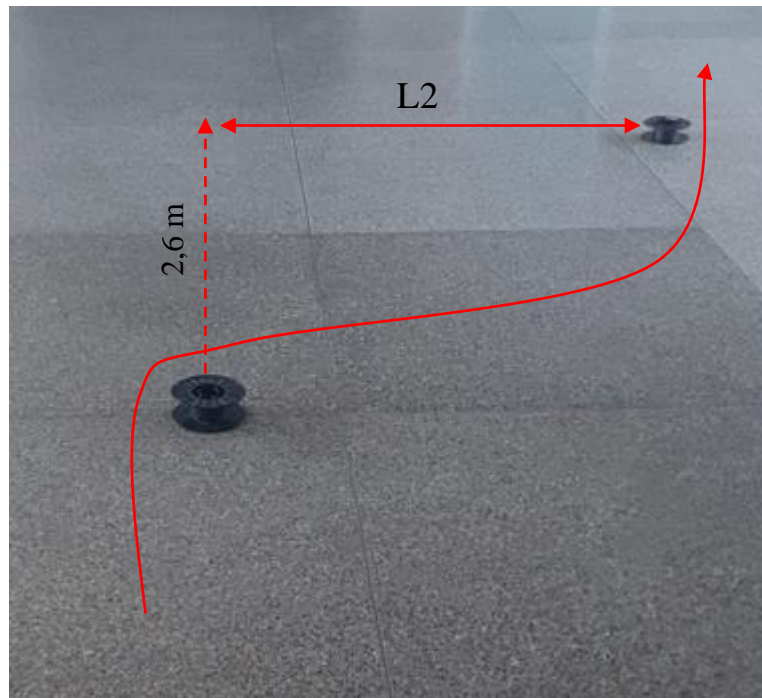
Antras eksperimentas buvo atliekamas apvažiuojant 2 kliūtis ir jo forma panaši į vienetinio šuolio signalą:



26 pav. Vienetinio impulso formos signalas

Eksperimentas vyksta važiuojant tiesiai pastoviu greičiu, tada apvažiuojama pirmą kliūtį ir daromas manevras, kad būtų apvažiuojama antroji kliūtis, ko pasėkoje, gaunamas impulso formos manevras, pakeičiant važiavimo trajektoriją. Šis manevras imituoja vienos kliūties apvažiavimą ir juo tiriama, galimybes apvažiuoti kliūtį bei grįžti į važiavimo kryptį, neišlaikant tos pačios pozicijos. Šis eksperimentas vykdomas tomis pačiomis sąlygomis, patalpoje ant lygaus grindinio, atsidėjus kliūtis fiksuotu ir pamatuotu atstumu, tačiau viena kliūtis atitraukta tam tikrais atstumais, taip keičiant impulso amplitudės dydį. Pradžioje reikėjo nustatyti, kokių atstumų reikia atidėti kliūtis, tai parametrizuoja manevro fronto statumą. Kadangi atlikti labai staigų posūkį su paspirtuku, jam važiuojant, yra labai sudėtinga arba neįmanoma, buvo pradėta nuo, ganėtinai didelio, 1 m atstumo tarp kliūčių, antrą kliūtį atitraukta 50 cm atstumu. Tačiau toks fronto statumas buvo žymiai per didelis ir atlikti manevro vis tiek nepavyko, taigi kliūtis buvo nutolinta per vieną grindinio plytos ilgį, t.y. 1,3 m tačiau tai nepadarė įtakos, taigi kliūtis buvo atitraukta tol, kol manevrą atlikti pavyko ir tai buvo ~2,6 m. Žinant ribinį fronto statumą, buvo pradėta nustatinėti impulso amplitudė, kuri bus ribinė ir įgyvendinama realiame važiavime. Šis dydis (L) keičiamas kas 50 cm ir atliekant važiavimus ties šiomis vertėmis:

- 50 cm;
- 100 cm;
- 150 cm;
- 200 cm.



27 pav. Manevro schema

Ties 2 m buvo pasiekta riba, nes atlikti tokį posūkį neprarandant greičio buvo ganėtinai sunku, tačiau didesnės eilės amplitudė jau buvo neįgyvendinama, taigi pasirinkta, kad 2 m atstumas yra ribinė vertė, nuo kurios manevras tampa neįmanomas ir gali pasibaigti destruktiviai. Kaip ir ankstesniame eksperimente, šiame taip pat buvo naudojamas sinchronizavimo impulsas, kad lengviau būtų nustatoma, kada prasideda manevras. Buvo naudojama ta pati žyma, kurią pervaziavus, buvo pradedamas manevras. Kadangi buvo aktualu turėti požymius, tiek apvažiuojant kliūtis per dešinę, tiek per kairę puses, tas pats eksperimentas buvo kartojamas ir 2 kliūtį atidėjus kitoje pusėje. Prikauptus pakankamai duomenų galima pereiti prie kitų eksperimentų.

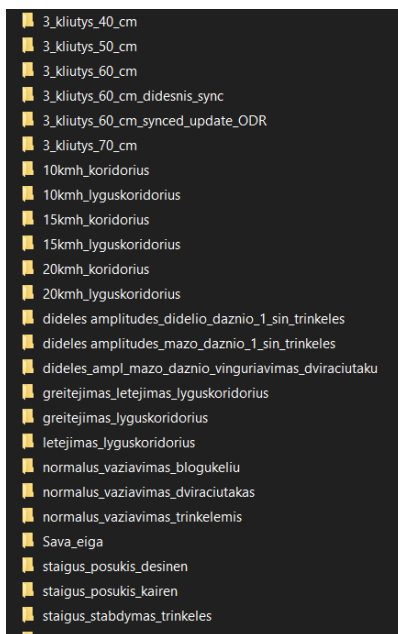
Sekantis žingsnis tai šuolio nuo bortelio eksperimento atlikimas. Tam specialaus pasirengimo ar testo parametrų sekti nereikia, kadangi svarbus, bent kokio šuolio nuo bortelio požymis. Šis manevrai nėra klasifikuojamas į saugų ar pavojingą, bei įmanoma ar neįmanoma atlikti realiai, kadangi šuolis gali būti atliekamas tik vienu ir tuo pačiu manevru, nesvarbu kokiu greičiu bus važiuojama ar kiek manevras truks, jei tai šuolis nuo labai aukšto ar labai žemo bortelio, visais atvejais tai turėtų būti klasifikuojama, kaip pavojingas manevras. Taigi eksperimentas buvo atliekami realiomis sąlygomis. Visi manevrai buvo atliekami iškart po sinchronizavimo atžymos. Šuoliui nuo bortelio, pasirinktas vidutinio aukščio bortelis stovėjimo aikštelėje, kad nebūtų trukdomas eismas.

Galiausiai buvo atliekami įprasto važiavimo eksperimentai. Važiuojant tiek patalpoje, tiek lauke, ant įvairių dangų buvo atliekami šie manevrai ir scenarijai:

- ramus važiavimas su įprastais posūkiiais;
- važiavimas su staigiais 2 posūkiiais;
- ramus pagreitėjimas ir lėtas sustojimas;
- ramus pagreitėjimas ir staigus stabdymas;
- man įprastas važiavimas lauke;

- greitas ir neatsakingas važiavimas kieme atliekant įvairius pavojingus ir sudėtingus manevrus (staigus sukimas, staigus stabdymas, šokinėjimas važiuojant tiesiu keliu ir panašiai);
- važiavimas įvairiomis kelio dangomis (prasta, trinkelėmis, dviračių taku, automobilių keliu).

Buvo atlikta labai įvairių manevrų ir važiavimų po kelis kartus ko pasėkoje, visi manevrai tvarkingai suklasifikuoti, pridėti aprašymai su konkrečių eksperimentų trumpais aprašymais ir galutinė duomenų bazė buvo sudaryta.

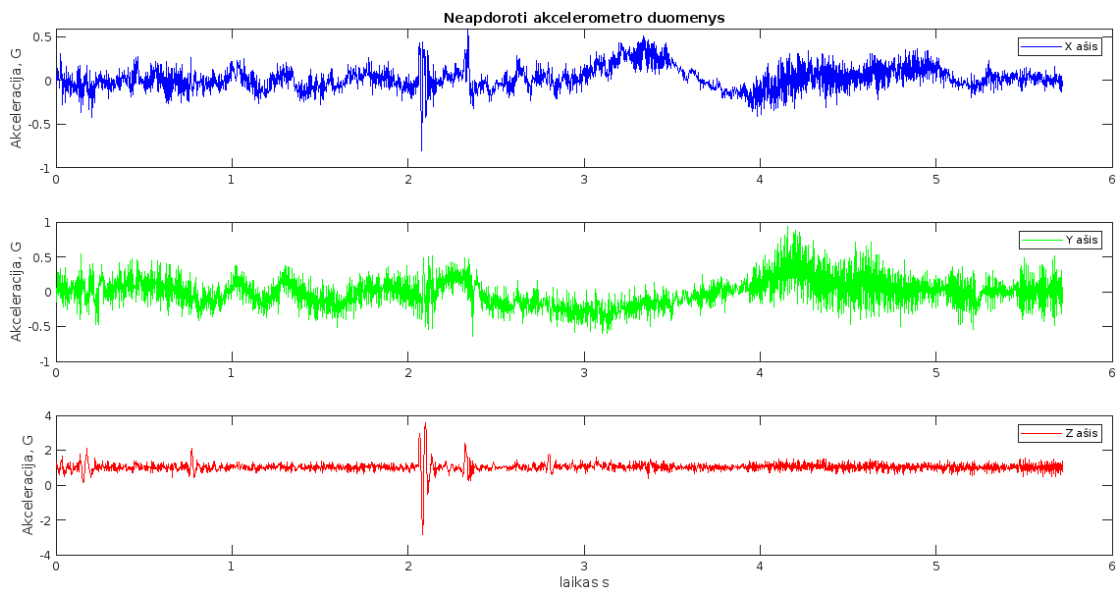


28 pav. Sudarytos duomenų bazės iškarpa

2.5. Duomenų paruošimas ir požymių charakterizavimas

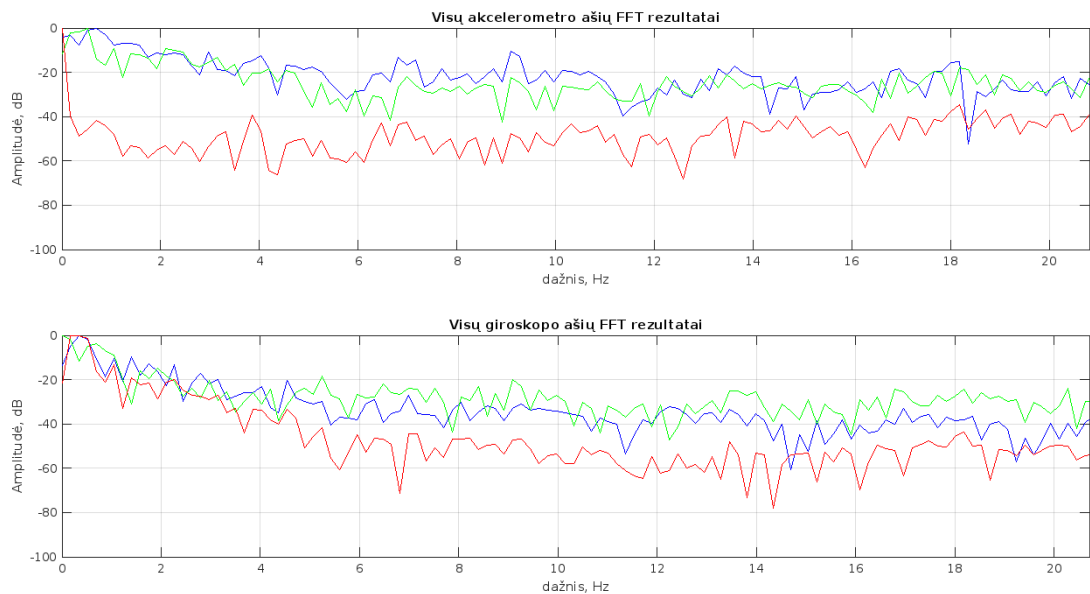
Sudarius manevrų duomenų bazę, visi įrašai buvo suklasifikuoti į dvi grupes: įrašai, kurie bus naudojami manevrui gauti ir įrašai, kuriuose bus ieškoma tam tikrų manevrų. Abi šios klasės šiek tiek skirsis jų duomenų paruošimu ir panaudojimu, kadangi įrašai, kuriuose ieškoma manevro, atitiks realias sistemos veikimo sąlygas su ribotais resursais. Algoritmas, skirtas manevro charakterizavimui, nebus naudojamas realioje sistemoje, taigi manevrui išgauti yra skirta, sąlyginai, neriboti resursai, svarbus tik rezultatas - kuo tikslesnis ir idealesnis manevro požymis. Požymiai - tai tam tikri signalo bruožai, toki kaip forma, amplitudė, trukmė, kuriuos pasitelkiant, galima apibendrinti ir apipavidalinti manevrą. Turint manevrų požymius, galima ieškoti jų realiuose važiavimo duomenyse ir lyginti naudotų algoritmų tikslumą.

Pirmas žingsnis - tai eksperimento duomenų įkėlimas. Kadangi įrašų skaičius, kiekvieno eksperimento, buvo bent 10 įrašų, buvo sukurtas algoritmas, kurio pagalba galima pasiimti bent kurio failo, bent kurio stulpelio duomenis, nes visi duomenys bus apdoroti vienu metu ir rezultate gaunamas vienas rafinuotas manevro požymis. Atvaizdavirus duomenis ir įsitikinus, kad pavyko tvarkingai atlikti failo nuskaitymą, buvo atliktas filtravimas, kadangi duomenys buvo užteršti akcelerometro vidinėmis paklaidomis.



29 pav. 2 kliūčių apvažiavimo eksperimento neapdoroti akcelerometro X, Y, Z ašių duomenys

Tam pasitelkta FFT funkcija ir atvaizdavus rezultatus, pavyko identifikuoti signalų dažnių juostas. Kadangi kiekviena akcelerometro ir giroskopo ašis fiksuoja skirtingus duomenis ir įvykius, jų dažnių juostos skiriasi, tačiau vizualiai apžiūrėjus signalų spektrus, buvo nustatyta, kad signalų dažnių juostos yra susitelkusios labai žemuose, $<10\text{ Hz}$ dažniuose:



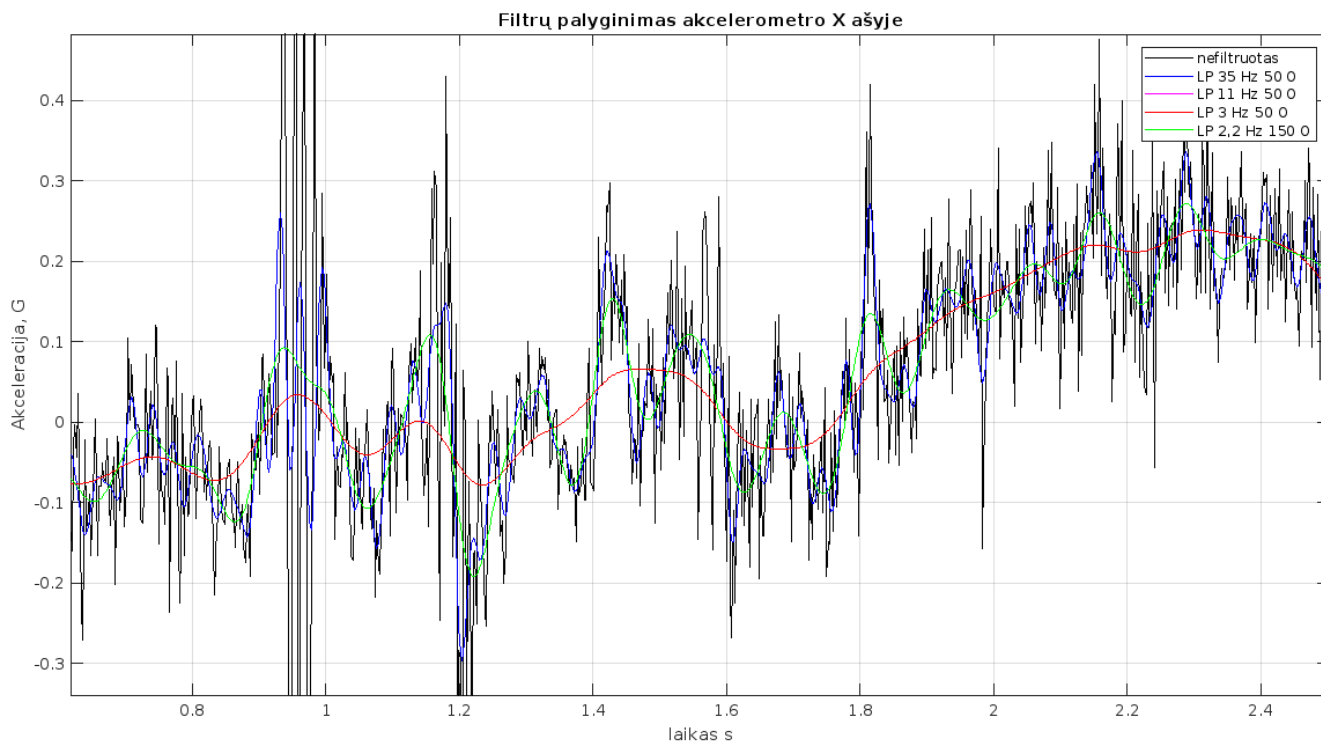
30 pav. 2 kliūčių apvažiavimo eksperimento akcelerometro ir giroskopo ašių FFT rezultatai

Toliau pereita prie filtro modeliavimo. Filtrui buvo išskirti šie kriterijai:

- filtras turėtų būti kuo statesniais frontais pralaidos juostoje;
- filtras turėtų būti žemų dažnių (LP), pageidautina FIR tipo dėl jo tiesiškumo fazės atžvilgiu;
- filtras turėtų kuo labiau slopinti visus kitus dažnius;
- filtras neturėtų iškreipti signalo fazės.

Pasitelkiant „Matlab“ aplinkos įrankį „FilterDesigner“ sumodeliuoti ŽD 50 ir 150 eilės filtrai su šiais atkirtos dažniais: 35 Hz, 11 Hz, 3 Hz ir 2 Hz. Taip pat, požymiui išgauti, buvo atliekamas specialus filtravimas vadinamas nuliniu-fazės poslinkio filtravimu (angl. *Zero-phase filtering*). Šis filtras

pasižymi tuo, kad yra tiesinės fazės filtras, kuris neįneš jokių fazinių iškreipimų į signalą po filtravimo. Tai galima pasiekti atliekant dvigubą filtravimą, signalą filtruoti normaliai ir tada gautą rezultatą siųsti per tą patį filtrą atbuline seka (paskutinę imtį filtruoti pirma), taip išsilygina fazinei iškreipimai ir rezultate gauname faziškai nepakitusių signalą. Vienintelis šio filtravimo minusas yra tas, kad tai nekauzalinė sistema, jos negalime naudoti realiu laiku filtruoti duomenis, nes reikia turėti visą įrašą, taip pat tai užtrunka ilgiau, nei taikyti paprastą filtravimą [28].

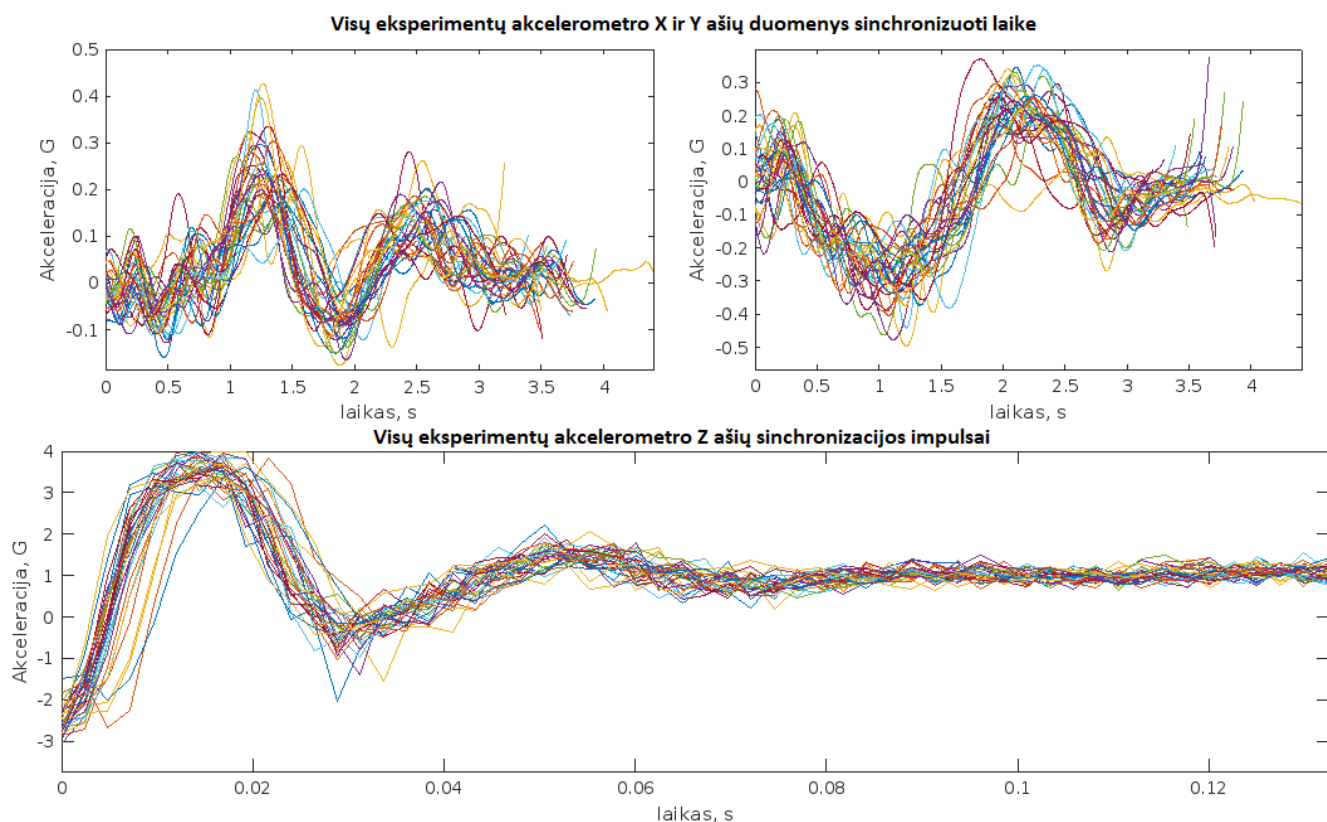


31 pav. 2 kliūčių apvažiavimo eksperimento akcelerometro X ašies filtravimo palyginimai

Atlikus filtravimą ir palyginus rezultatus, iškart pavyko nustatyti, kad 35 Hz filtras nesugeba pašalinti visų triukšmų. 3 ir 11 Hz filtrai grąžino tą patį rezultatą, tačiau 2,2 Hz filtro rezultatuose atsirado triukšmo dedamųjų, taigi galima teigti, kad 2,2 Hz filtras pradėjo slopinti signalo dažnių ruožą. Iš šių rezultatų pasirinkta naudoti FIR 3 Hz LP filtrą, taigi buvo atliktas visų duomenų filtravimas šiuo filtru ir „Matlab“ aplinkos nulinės-fazės filtravimo funkcija *filtfilt*. Vienintelių duomenų, kurių nereikėjo filtruoti buvo akcelerometro Z ašies duomenys, kadangi ši ašis bus naudojama sinchronizuoti visų eksperimentų duomenis laike, atlikus filtravimą, svarbus duomenų pikas gali pranykti arba jo amplitudė stipriai sumažėti, taigi šią ašį buvo nuspręsta palikti visiškai neapdorotą.

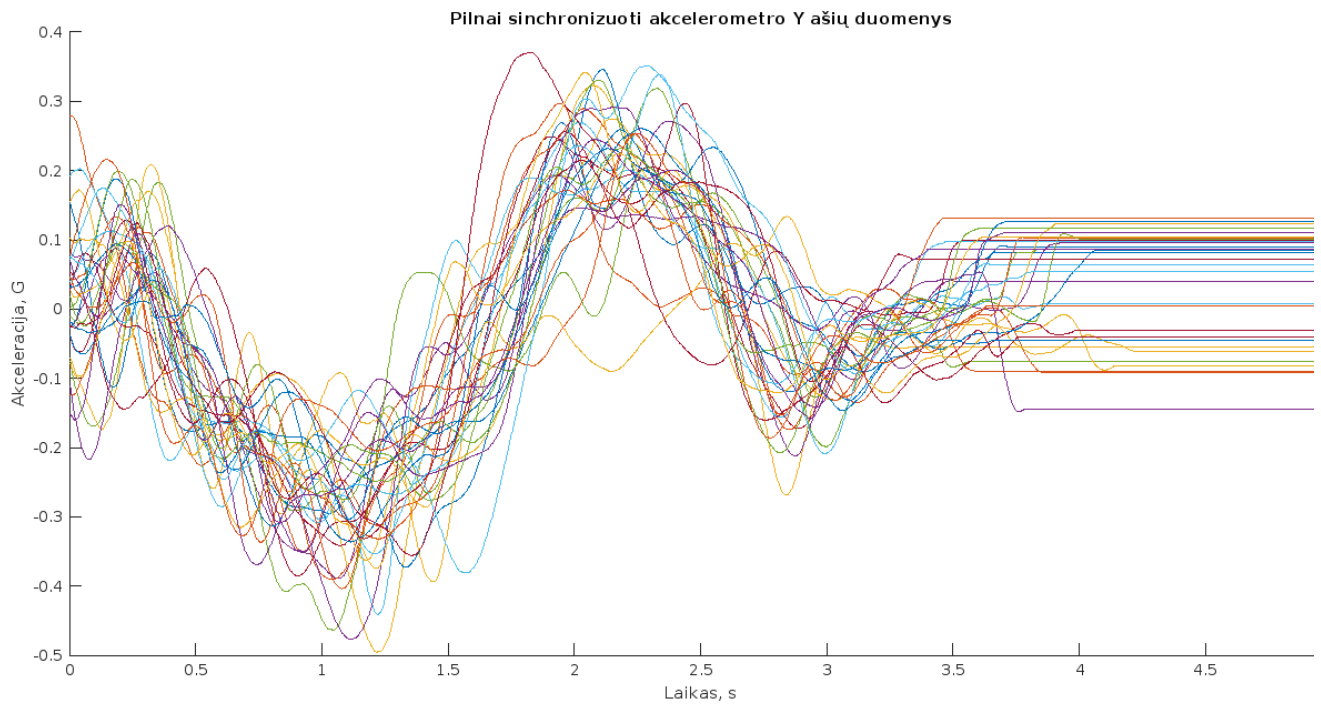
Sekantis žingsnis buvo visų eksperimento duomenų sinchronizavimas laike, tai buvo reikalinga atlikti dėl to, nes visų įrašų ilgai skiriasi, taip pat skiriasi ir manevro pradžios laikai įrašė, bandant aklaui spėti ar vizualiai nustatyti, kada prasideda manevras, yra labai netikslu ir užtruktų ilgai, tai atlikti dideliame duomenų kiekiui, tačiau atliekant eksperimentus, duomenyse buvo paliekami ganėtinai išsiskiriantys pikai prieš pat manevro pradžią ir šie pikai yra fiksuoti erdvėje, taigi jie puikiai žymi manevro pradžios atskaitą. Duomenyse šią žymą galima rasti akcelerometro Z ašyje, joje pirmose sekundėse yra matomi du aukšti pikai. Buvo nuspręsta visus duomenis sinchronizuoti pagal apatinį piką, kadangi jis yra tik vienas ir dažniausiai didesnės amplitudės, nei du viršutiniai. Peržvelgus kelių failų grafikus, pasirinkta vidutinė apatinio piko vertė, kurią, testuojant algoritmą, pavyko paderinti ir išoptimizuoti, kad ji nebūtų per maža, kitaip galima susinchronizuoti duomenis nuo įvairių šiukšlių

atsiradusių duomenyse, ir ne per didelė, kad pavyktų aptikti apatinį piką. Suradus apatinio piko elemento vietą, visus duomenis iki šios vertės galima iškirpti.



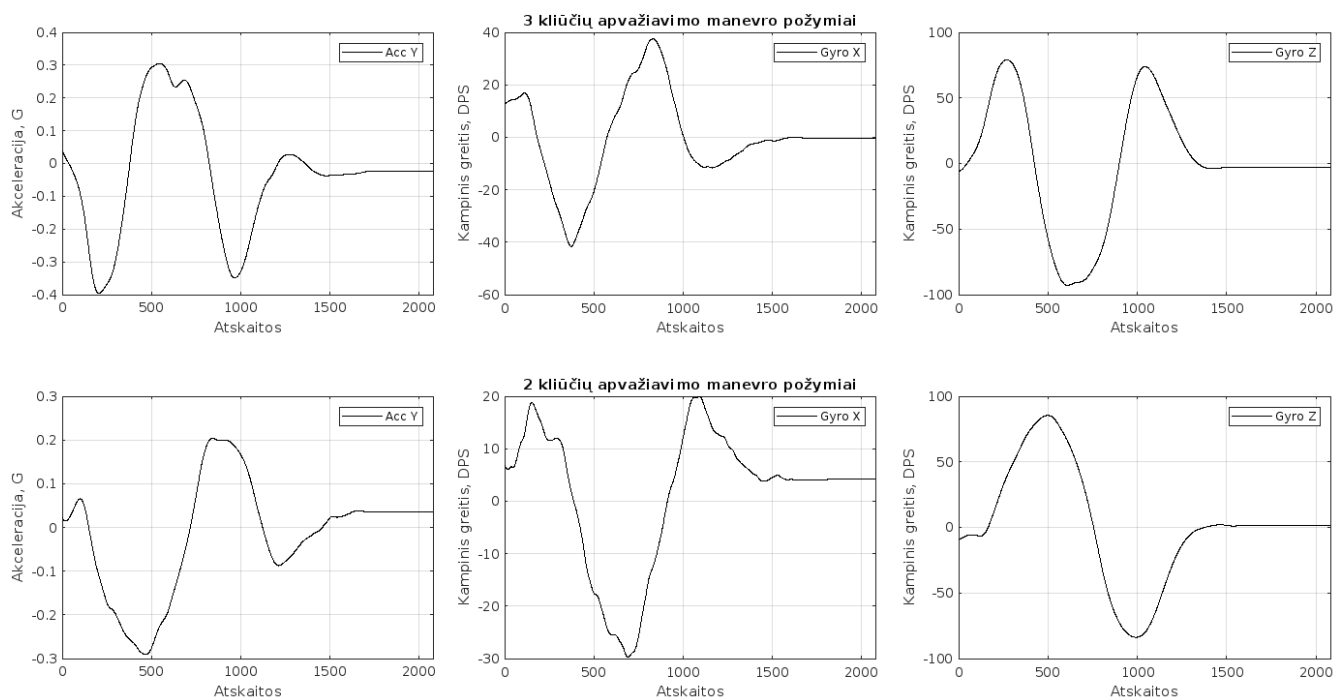
32 pav. 2 kliūčių apvažiavimo eksperimentų akcelerometro duomenys sinchronizuoti laike

Suvienodinus visų eksperimentų failų duomenis pradžias, reikėjo tai atlikti ir gale, kadangi visi įrašai skiriasi jų trukmėmis, taip pat dažnai nepavyko atkartoti identišką pabaigos eksperimento užbaigime, idealiu atveju, eksperimentas turėtų pasibaigti nuvažiuojant tiesiai be jokių papildomų manevrų, tačiau realiose sąlygose tai padaryti ne visada pavyko, dėl ko signaluose pabaigoje matomi verčių pokyčiai ar užkilimai prieš pat pabaigą. Kad būtų išspręstos pabaigos sinchronizavimo ir duomenų užbaigimo problemos, toliau atliktas dirbtinio signalo prailginimo veiksmas, tačiau to padaryti negalima atsitiktine verte, taigi buvo sukurtas algoritmas, kuris tarp paskutinių, pasirinkto kiekio, atskaitų (pvz: 100, 200, 500) suras artimiausią 0 atskaitą (pvz: 0,05, 0,02, 0,1), tada nukirps visas likusias atskaitas ir *wextend* funkcijos pagalba sukurs naujas atskaitas, kurios užbaigs signalo frontą ir pratęs jį tam tikra paskutine verte. Kadangi signalo pabaiga manevro požymiui nėra aktuali, nes manevras jau yra pasibaigęs, taigi nėra labai svarbu išlaikyti duomenų autentiškumą, šis veiksmas yra skirtas tam, kad būtų gaunama vienodi signalų ilgiai, kuriuos vėliau galima karpyti ir koreguoti.



33 pav. 2 kliūčių apvažiavimo eksperimentų akcelerometro Y ašių sinchronizuoti duomenys

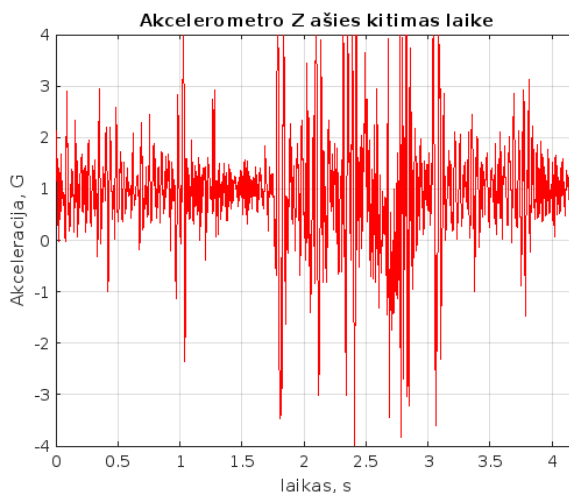
Paskutinis etapas - tai požymio charakterizavimas. Manevro požymis iš turimų signalų buvo ištraukiamas laiko srityje, kadangi duomenys laike yra dinamiškesni ir neša daugiau informacijos apie manevrą, duomenų amplitudės kinta aiškiau, taip pat tam tikrais laiko momentais galima matyti, kada atliekamas posūkis, šuolis ar stabdymas, dažnių ašyje šie manevrai pasireiškia, kaip tam tikros dažninės dedamosios, tačiau dėl įvairių eksperimento važiavimo sąlygų, gali atsirasti ir dabar ne užfiksuotų dedamųjų, kurios apsunkintų požymio atpažinimą. Turint visų eksperimentų duomenis sinchronizuotus laike ir juos atvaizdavus, galima išvelgti manevrų tam tikras signalų formas, būdingas tik tiems manevras, tačiau kiekvieno eksperimento signalų amplitudės ir trukmės bei formos šiek tiek skyrėsi, taigi norint išgauti konkretizuotą ir apibendrintą signalą, privalu atlikti visų signalų vidurkinimą, paskaičiuoti vidutinę, kiekvienos atskaitos konkrečiu laiku, vertę. Tai buvo atlikta pasitelkiant aritmetinio vidurkio (2) formulės pagalba. Gautas rezultatas - tam tikro eksperimento ir laike izoliuoto manevro požymis, etaloninis signalas, į kurį yra panašus kiekvienas eksperimento manevras. Darant kuo daugiau eksperimentų šis signalas artės link idealaus. Panaudojus šį algoritmą 3 ir 2 kliūčių eksperimentams, iš jų charakterizuoti šie požymiai:



34 pav. Charakterizuoti manevrų požymiai

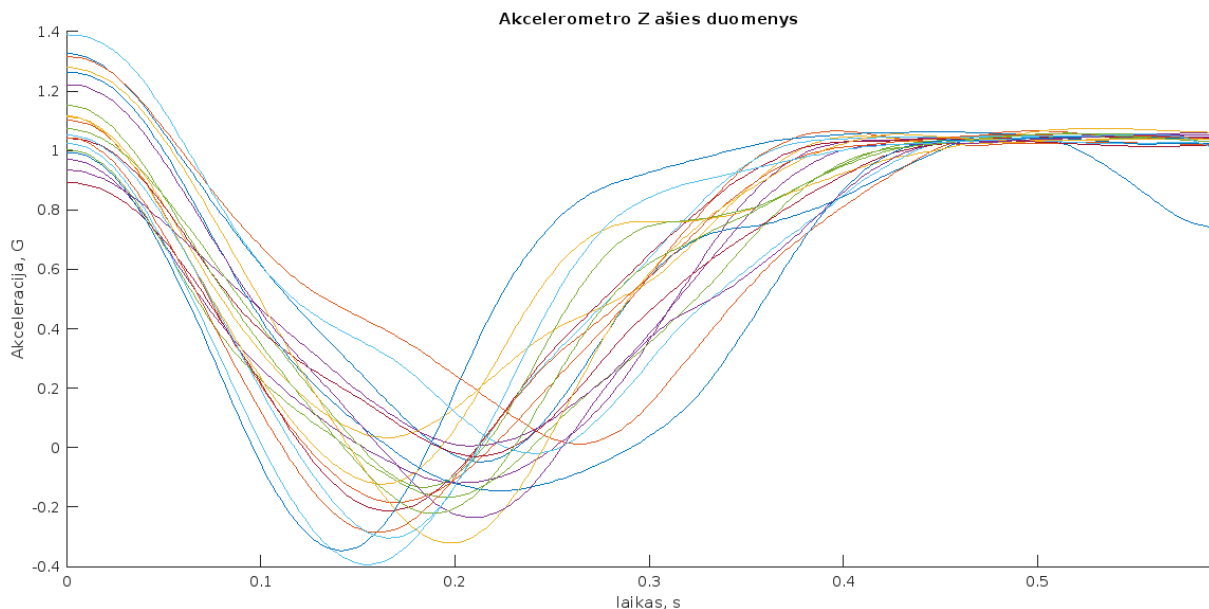
Buvo pasirinktos 3 duomenų ašys, kadangi ne visi duomenys yra aktualūs konkrečiam manevro metu, pavyzdžiui, važiuojant tiesiai ir darant staigius posūkius, akcelerometro X ir Z ašyse nebus matoma labai aiškių ar aktualių formų, nes šios ašys nefiksuoja šoninio pagreičio, tačiau giroskopo duomenyse šios ašys gali būti naudingos. 2 ir 3 kliūčių apvažiavimui daugiausiai informacijos neša akcelerometro Y ašis ir giroskopo X ir Z ašys. Taip pat, turint daugiau nei vieną manevro požymį, padidina šansą jį aptikti, nes tikrinama daugiau ašių ir ieškoma panašių formų, jei vienoje iš ašių manevras nebus aptinkamas, kitose dvejose jis gali būti fiksuojamas.

Dar vienas manevras, kurio požymis taip pat svarbus tyrime, tai šuolis nuo šaligatvio bortelio. Šuolis nuo bortelio yra ne kas kita, kaip laisvasis kritimas labai trumpą laiką, nes nuvažiuojant nuo šaligatvio, mikromobilumo priemonė kartu su vairuotoju patiria 1 G jėgą, kurios vektoriaus kryptis yra link žemės centro. Idealiomis sąlygomis, važiuojant tiesiai akcelerometro ašis visada rodytų 1 G dydį, tačiau patiriant laisvąjį kritimą, ji nukris iki 0, nes jėgos išsilygina. Apdorojant duomenis, visa veiksmų seka buvo atliekama ta pati ir pasitelkiant tą patį filtravimo metodą, kaip ankstesniuose manevruose.



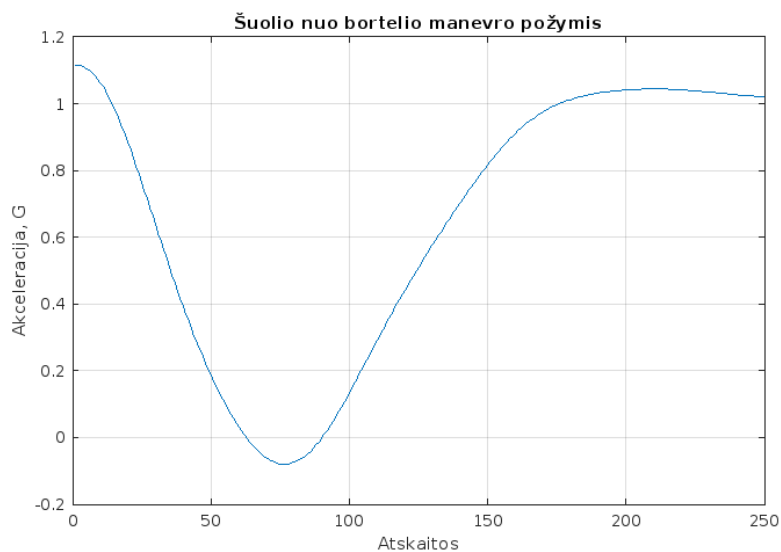
35 pav. Šuolio nuo bortelio neapdoroti akcelerometro Z ašies duomenys

Sinchronizacija buvo vykdoma pagal atsikartojantį teigiamą duomenų piką, prieš pat laisvąjį kritimą, nes manevras atliekamas šokant nuo bortelio, visu kūnu kartu su paspirtuku fiziškai šokant į viršų, kad nuvažiuojant nebūtų kliudomas paspirtuko dugnas, šis šuolis ir buvo sinchronizacijos impulsas. Signalų pabaigos buvo nukertamos iškart kai tik amplitudė pakilo iki $\sim 1 G$ dydžio. Duomenų filtravimui buvo pasitelktas jau naudotas $3 Hz$ LP filtras. Rezultate gauta filtruotų manevro duomenų atvaizdas viename grafike.



36 pav. Šuolio nuo bortelio eksperimentų akcelerometro Z ašių sinchronizuoti duomenys

Šis manevras išsiskiria iš prieš tai tirtų tuo, kad yra lengviausiai atpažįstamas, aptikus akcelerometro Z ašies kritimą, galima teigti, kad manevras buvo atliktas. Tačiau bent kokios vibracijos Z ašyje sukelia duomenų švytavimą aplink 0, itin ekstremaliais atvejais, kai pervažiuojama nedidelė duobė ar šaka galima pasiekti ir artimą 0 dydį. Kad atskirti manevrą nuo triukšmo ar grindinio dangos sukeltamų duomenų kritimo, svarbu įvertinti trukmę, kaip greitai neigiamas pikas leidosi ir vėl kilo. Iš gautų duomenų galima teigti, kad šuolis nuo bortelio yra ganėtinai lėtas procesas ir amplitudė nekinta staigiai lyginant su neapdorotuose duomenyse matomais staigiai amplitudės švytavimas. Pasitelkiant (2) formule buvo gautas šuolio nuo bortelio manevro požymis.



37 pav. Šuolio nuo bortelio manevro požymis

2.6. Standartinės deviacijos skaičiavimas

Standartinė deviacija – tai statistinis dydis, parodantis, kokiuose režiuose imties vertė gali varijuoti nuo vidutinės visų imčių vertės. Šis dydis žymimas σ (graikiškai *sigma*) ir yra apskaičiuojamas pagal šią formulę:

$$\sigma = \sqrt{\frac{1}{N-1} \sum_{i=1}^N |x_i - \mu|^2} \quad (4)$$

čia:

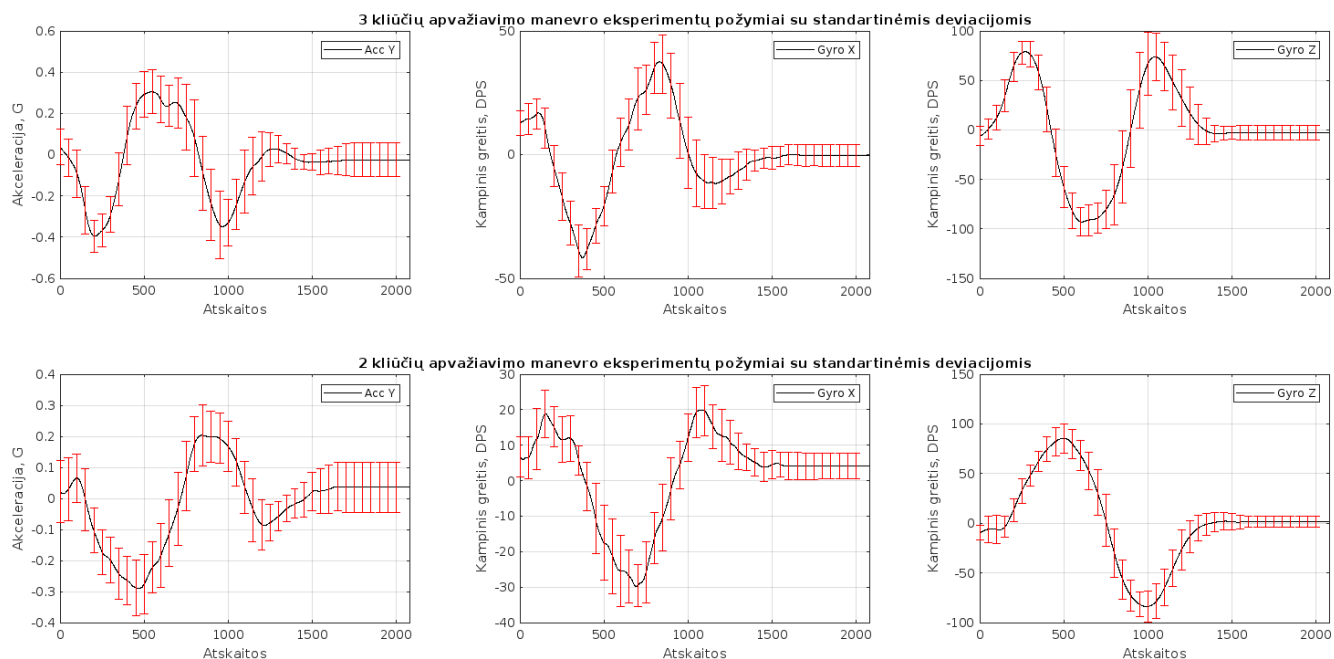
σ – standartinė deviacija;

N – imčių kiekis;

μ – vidutinė imčių vertė;

x_i – imties vertė.

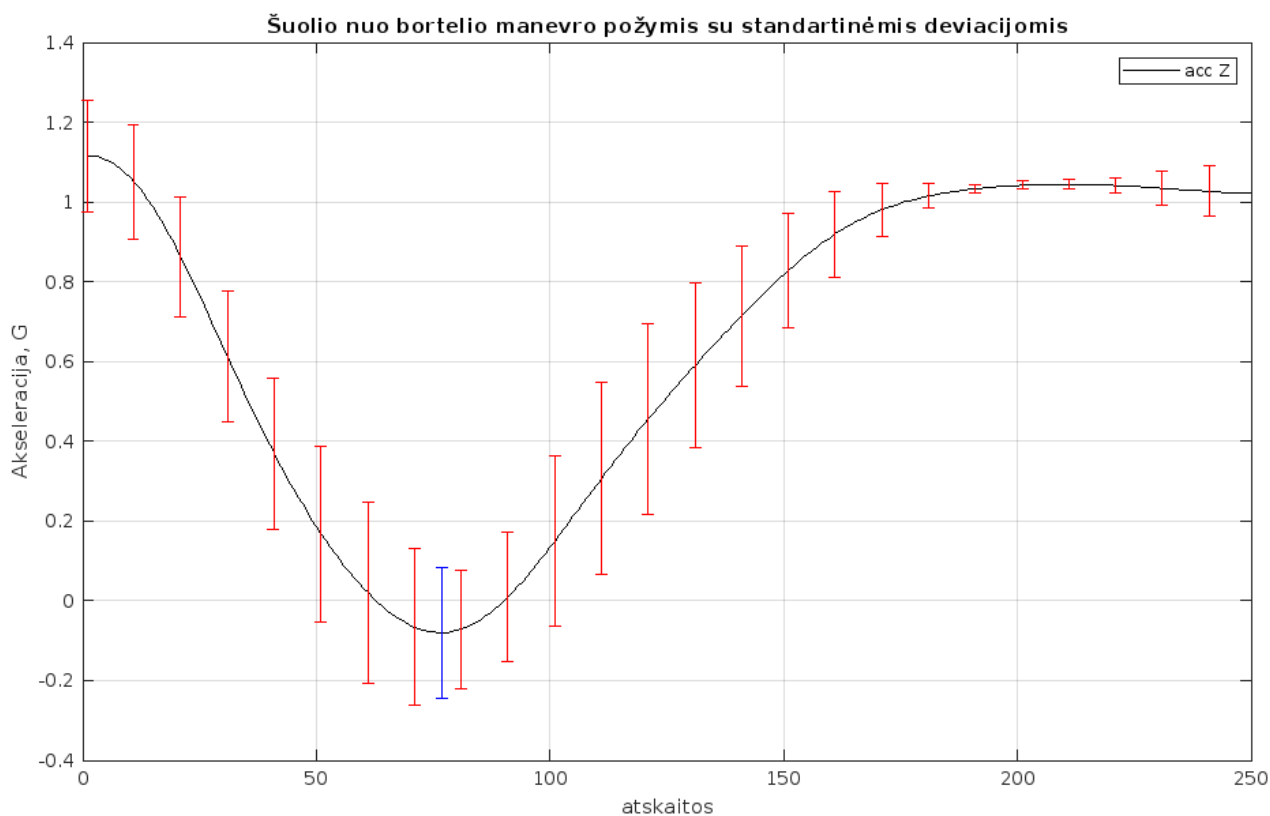
Standartinė deviacija nurodo, kokiuose režiuose atsitiktinai išsibarstę duomenys. Ji žymi Gauso arba normaliojo skirstinio pakopas. 68,2 % viso matavimo imčių gali būti nutolusios nuo imčių vidurkio per $\pm \sigma$, 95,4 % imčių – per $\pm 2\sigma$ ir t.t. [29]. Šis dydis parodo matavimo tikslumą, duomenų atsikartojamumą bei triukšmo ir atsitiktinių veiksnių įtaką, kuo standartinė deviacija mažesnė, tuo daugiau matavimo imčių yra artimos vidurkiui. Tai labai svarbu ieškant manevro požymių, nes esant didelei deviacijai, manevro požymis gali mažiau atspindėti realiose sąlygose sutiktam manevrui. Kadangi manevro požymis yra visų eksperimentų duomenų vidurkis, buvo paskaičiuota standartinė deviacija tarp visų eksperimentų duomenų. Eksperimento duomenų imtys buvo imamos kas 50 atskaitų. Apskaičiuotos ir pavaizduotos 2 manevrų, 3 požymių standartinės deviacijos:



38 pav. Apskaičiuotos standartinės deviacijos

Iš gautų rezultatų matyti, kad standartinė deviacija įvairiose atskaitose didžiausia akcelerometro duomenyse, kadangi šis jutiklis labai jautrus įvairioms vibracijoms ir Žemės traukai. Giroskopo duomenys abėjuose manevruose rodo, kad išsibarstymas stipriai sumažėja pabaigus arba pradėjus manevrą, kai nekinta paspirtuko orientacija ir neatliekamas joks sukimasis aplink ašį. Giroskopo Z ašies manevras, iš visų trijų manevrų, rodo geriausias deviacijos rezultatus, šioje ašyje visi eksperimentai su panašiu atsikartojamumu ir mažu nuokrypiu nuo vidurkio užfiksavo duomenis net ir manevro metu. Šios ašies požymis tiksliausiai atspindi visų eksperimentų duomenis, todėl ja pasikliauti būtų tinkamiausia, akcelerometro Y ašies požymis mažiausiai atspindi kiekvieną eksperimentą, dėl ganėtinai nemažos deviacijos viso manevro metu. Tačiau kiekvienas manevro požymis, net su didžiausia deviacija, išlaiko savo formą.

Taip pat buvo paskaičiuota šuolio nuo bortelio manevro standartinė deviacija. Šiuo atveju svarbu ne tik viso manevro deviacija ir galimi nuokrypiai, bet ir neigiamo piko (žemiausios gautos vertės) deviacija. Žinoti galimas vertes, kiek daug ar mažai akcelerometro duomenys sumažės, yra naudinga ir taikant tolimesnius identifikavimo metodus. Standartinė deviacija šiam manevrui buvo skaičiuojama kas 10 atskaitų, nes tai trumpesnis manevras. Papildomai buvo paskaičiuota ir išskirta žemiausio taško (neigiamo piko) standartinė deviacija.



39 pav. Atvaizduotos standartinės deviacijos su papildoma piko deviacija

Iš rezultatų matyti, kad duomenys labiausiai kinta leidimosi metu, tai gali lemti ir priemonės greitį ir vartotojo svorį. Ilgesnių manevrų frontas bus lėtesnis todėl ir vertės bus didesnės konkrečiais momentais, lyginant su greitesnių manevrų matavimais. Taip pat manevro pabaigą žymi amplitudės šuolis atgal į 1 G ir tuo momentu, kad manevras pasibaigia, matomas labai minimalus deviacijos dydis, visų eksperimentų metu gautas itin panašus dydis. Tai mažiausia standartinė deviacija užfiksuota visų skaičiavimų metu, kas gali reikšti, kad ši vertė yra mažai veikiama triukšmo ir dažniausiai atsikartoja tokio pačio ar labai artimo dydžio.

2.7. Slenkančio vidurkio filtro modeliavimas

Naudotas 50 eilės 3 Hz atkirtos dažnio LP ir nulinio fazės poslinkio filtras itin stipriai slopina triukšmus ir išryškina signalo formą, kas ir svarbu ieškant manevro. Tačiau tokio tipo filtravimą atlikti realioje sistemoje būtų labai sudėtinga, nes reikalingas nemažas atminties kiekis laikyti 51 koeficientą. Taip pat norint išgauti nulinio fazės poslinkio filtravimą, reikėtų signalą kaupti buferyje, kas taip pat reikalautų daug atminties, sukaupus buferį, jį filtruoti du kartus, nuo ko kentėtų greitaveika. Reikalingas kitas metodas, kuris tiktų realiu laiku apdoroti signalą ir neekvotų atminties resursų. Dažniausiai naudojamas ir tenkinantis šiuos kriterijus filtras yra slenkančio vidurkio (angl. *Moving Average*) filtras. Šis filtras yra optimalus paprastoms užduotims: sumažinti atsitiktinį triukšmą ir filtravimą atlikti laiko ašyje. MA filtras yra paprastas vidurkio skaičiavimas naujai atskaitai, jis veikia paimdamas atskaitą ir kartu su šalimais esančiomis, paskaičiuoja naują vertę, sekanti atskaita išmes seniausią vertę iš sumos skaičiavimo ir paims naują, taip vyks slinkimas per atskaitas. Ma filtro formulė pateikta apačioje:

$$y[i] = \frac{1}{M} \sum_{j=0}^{M-1} x[i + j] \quad (5)$$

čia:

y – išėjimo signalas;

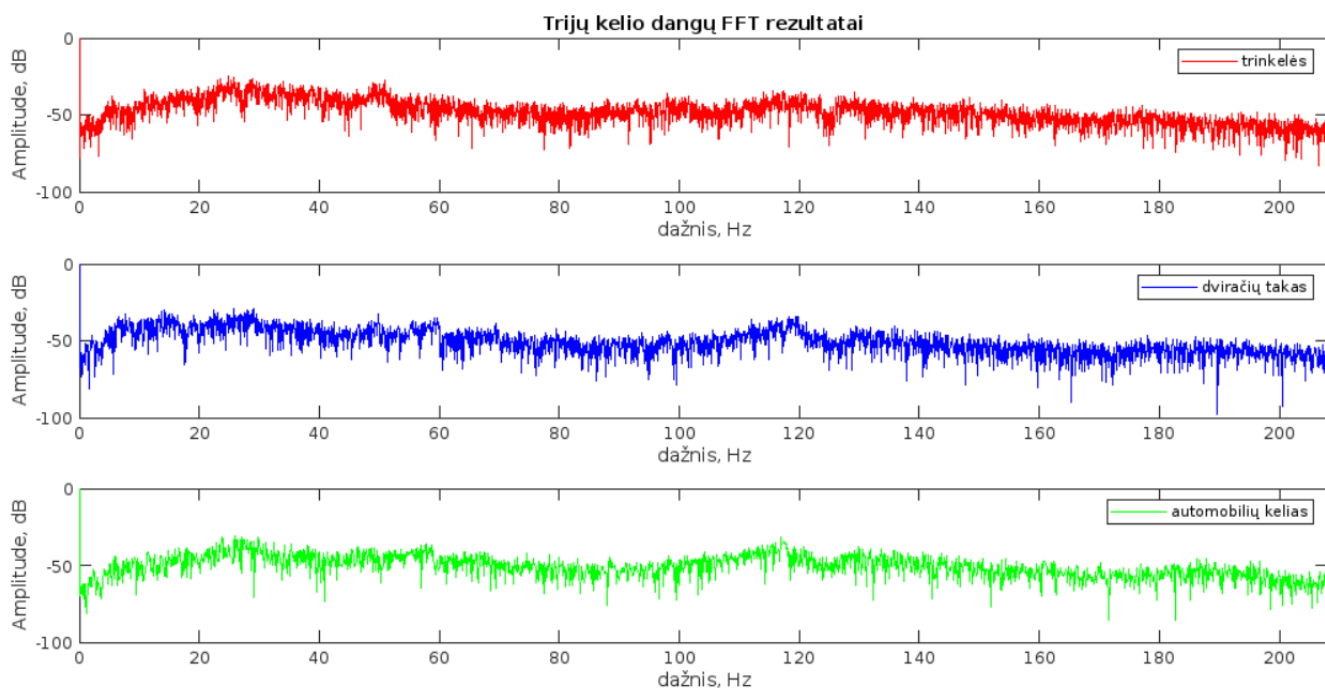
M – slenkamojo filtro buferio imčių skaičius;

x – įėjimo signalas;

Šis filtras yra žemų dažnių tipo, taigi jis puikiai tiks tiriamiems signalams filtruoti. Taip pat jis labai lengvai realizuojamas, tereikia skaičiuoti atskaitų sumą ir padalinti iš norimo kiekio, taip pat įmanoma optimizuoti šį algoritmą, kad nereiktų kaskart skaičiuoti naujos sumos, galima naudoti prieš tai buvusią sumos vertę, atimti išstumtą atskaitą ir pridėti naują [30]. Šio filtro pagalba bus šalinamas atsitiktinis triukšmas bei aukštadažniai virpesiai įrašuose, kuriuose bus ieškoma manevro. Tokiu būdu algoritmai bus testuojami realiomis sistemos veikimo sąlygomis. Pradžioje buvo ištirtos kelios standartinės kelio dangos, kuriomis važiuojama mikromobilumo priemone. Tai yra dviračių takas, pėsčiųjų takas grįstas trinkelėmis, važiuojamoji dalis. Įvairūs virpesiai iš trinkelėlių tarpų, dviračio tako nelygumai ar kiti triukšmai, ateinantys iš paviršiaus gali labai stipriai užteršti signalą ir manevras gali pranykti triukšme. Žinant standartinių dangų keliamus triukšmus ir dažnių juostas bus galima nustatyti kriterijus, modeliuojamam filtrui. Atlikus tiesaus važiavo be papildomų manevrų visomis dangomis eksperimentus ir pasitelkiant *FFT* operaciją, buvo gautos šios triukšmų, ateinančių iš kelio dangos, vertės akcelerometro Z ašyje:

- važiuojant trinkelėmis – 50 Hz;
- važiuojant dviračių taku – 120 Hz;
- važiuojant automobilių keliu – 116 Hz;

Taip pat buvo pastebėta, kad visose kelio dangose egzistuoja 25 Hz plačiajuostis triukšmas.

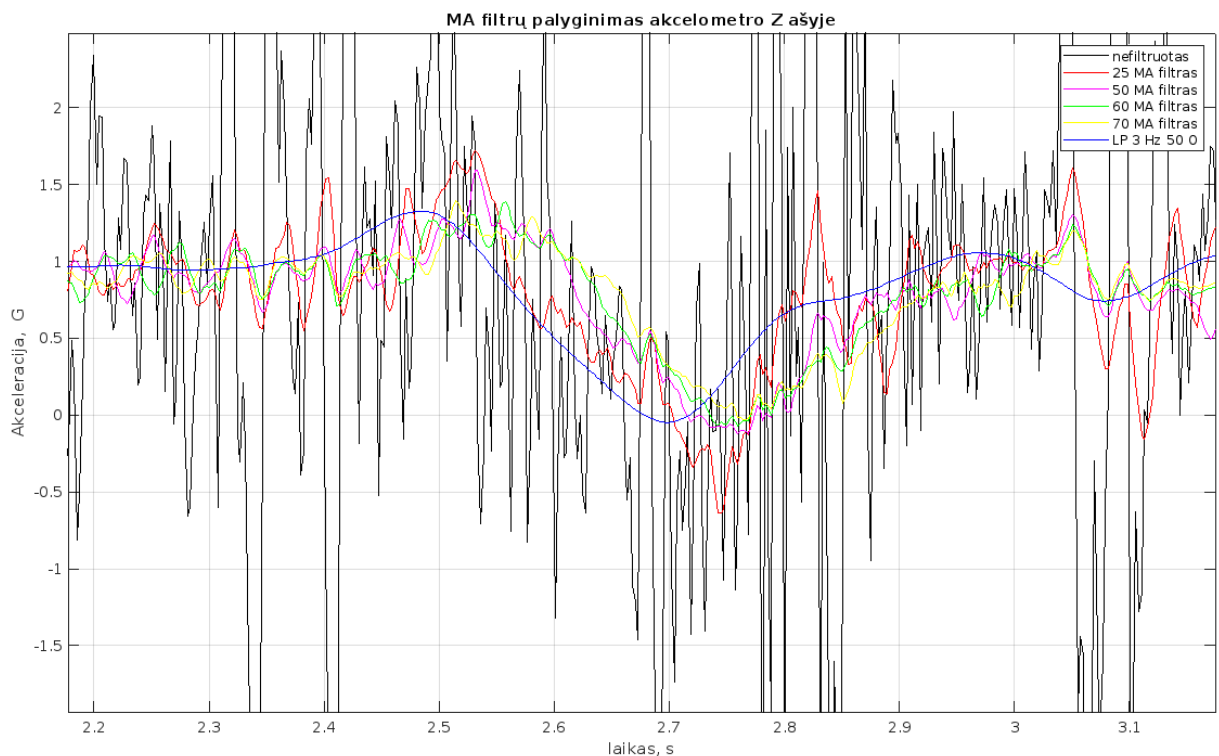


40 pav. Trijų kelio dangų važiavimo *FFT* palyginimas

Remiantis šiais rezultatais buvo nustatyta, kad didžiausia, galima atkirtos dažnio riba – 20 Hz . Toliau eksperimentiškai ir stebint amplitudės dažninę charakteristiką buvo modeliuojami kelių tipų MA filtrai. Buvo nustatyta, kad geriausius slopinimus ties aptiktais triukšmų ir aukštesniais dažniais atlieka šių buferių dydžio filtrai:

- 25 atskaitų buferio dydis;
- 50 atskaitų buferio dydis;
- 60 atskaitų buferio dydis;
- 70 atskaitų buferio dydis;

Filtruojant rezultatai buvo lyginami su naudotu LP 3 Hz nulinio fazės poslinkio filtru. Šis filtras atitinka etaloninį filtrą, kurį, idealiu atveju, ir buvo bandoma atkurti MA filtru. Rezultate matyti, kad 25 eilės filtras nesugeba prislopinti triukšmų ir signalas dar yra ganėtinai užterštas. 60 ir 70 eilės filtrai ganėtinai identišškai prislopino triukšmo lygi, tačiau juose matomas ir didžiausi fazės iškreipimai. Tarpiniai rezultatai, tarp gero slopinimo ir kuo mažesnio fazės kraipymo buvo gautami ties 50 filtro eile.



41 pav. Slenkamojo vidurkio filtrų palyginimas

Dėl šių priežasčių buvo pasirinkta naudoti būtent šį, 50 eilės MA filtrą tolimesniam tyrimui.

2.8. Manevrų identifikavimo metodai

Manevrui identifikuoti iš turimo įrašo reikia dviejų dalykų, tai manevro požymio, kad žinotume ko ieškoti signale ir algoritmo, kuris sugebėtų aptikti tą požymį signale. Vizualiai tai padaryti nėra sudėtinga, nes aukščiau išskirti manevrai yra aiškiai apibrėžti trukme, amplitudės kitimais ir forma, turint važiavimo įrašą galima identifikuoti atliktus maneversus ir jų kiekį. Tačiau dirbant su įterptinėmis sistemos tai turi gebėti atlikti procesorius, taigi toliau bus aptariami naudoti algoritmai.

2.8.1. Kros-koreliacijos metodas

Pirmas ir bene plačiausiai naudojamas algoritmas, yra koreliacija. Tai matematinis algoritmas, kuris naudoja du signalus paskaičiuoti trečiam signalui. Jei koreliacijai naudojami du skirtingi signalai, jų rezultatas yra vadinamas kros-koreliacija, jeigu koreliuojamas tas pats signalas, tačiau prastumtas laike, rezultatas vadinamas autokoreliacija [31]. Koreliacijos algoritmas veikia paimdamas vieną signalą ir jį slinkdamas į kairę arba dešinę po vieną atskaitą per kitą signalą ir skaičiuoja jų atskaitų sandaugų sumą. Rezultate gauta reikšmė parodo, kaip stipriai signalų atskaitos tarpusavy koreliuoja, kuo vertė didesnė, tuo signalų atskaitos panašesnės viena į kitą. Kros-koreliacijos rezultate gauta maksimali vertė yra vadinama koreliacijos piku ir ji žymi laiko momentą, kada signalai sutampa labiausiai ir yra vienas į kitą panašiausi. Koreliacijos formulė pateikta apačioje [32]:

$$CC[m] = \sum_{n=1}^N x[n] * y[n - m] \quad (6)$$

čia:

CC – kros-koreliacijos rezultatas;

N – ilgiausio signalo ilgis;

m – koreliacijos rezultato ilgis: nuo $-N + 1$ iki $N - 1$;

x – stacionarus signalas;

y – signalas, kurį per 1 atskaitą stumiami į kairę ar dešinę pusę.

Šis algoritmas dažniausiai naudojamas radarų sistemose ir skaičiuojama autokoreliacinė funkcija, tačiau jo pagalba galima nesunkiai aptikti manevrus signale. Siunčiant požymio signalą per akcelerometro filtruotų duomenų įrašą ir ieškant kros-koreliacijos piko galima aptikti tikslų laiko momentą, kada manevras buvo atliktas. Esminiai šio metodo dalykai yra kros-koreliacinio piko vertė ir atstumas tarp šių pikų. Kadangi algoritmas skaičiuoja sandaugų sumą, net ir atsitiktinis triukšmas turės įtakos į koreliacijos rezultatą. Norint atskirti manevrą nuo atsitiktinio triukšmo ar kitų manevrų signale, reikia nustatyti piko vertę, nuo kurios galima teigti, jog manevras koreliuoja su požymiu ir daryti prielaidą, kad manevras buvo aptiktas. Taip pat tarp pikų turėtų būti bent vieno požymio dydžio tarpas, kadangi neįmanoma atlikti daugiau manevrų greičiau. Visi kiti pikai neturėtų būti fiksuoti kaip manevras, iki praeis požymio signalo atskaitų kiekis. Realus koreliacijos rezultatas pateiktas apačioje:



42 pav. 3 kliūčių apvažiavimo akcelerometro Y ašies kros-koreliacijos rezultatas

Taip pat koreliacinio piko derėtų ieškoti tik vertėse, kurios yra tarp 0 ir signalo ilgio režiuose. Taip yra dėl to, nes koreliacija atliekama prailginant trumpesnę signalą nuliais iki ilgesniojo ilgio, taip pat slenkamas signalas pradedamas stumti nuo momento, kai jo paskutinė atskaita dauginama su stacionaraus signalo pirmąją. Iki 0 rezultato atskaitos, požymio signalas pilnai neužlipa ant testuojamo signalo ir ieškoti tik dalies požymio manevro būtų klaidinga. Pasitelkiant visus šiuos kriterijus buvo atliekama 10 kiekvieno turimo manevro važiavimų, kai atliekamas pavojingas manevras. Įrašyti testų duomenys buvo filtruojami MA filtru, šuolio nuo bortelio testui papildomai

reikėjo pašalinti nuolatinę dedamąją, taigi buvo atimta reikšmė 1 (apytikslė pastovi Žemės laisvojo kritimo pagreičio vertė). Iš kros-koreliacijos pikų rezultatų ieškoma mažiausia vertė, kuri bus naudojama kaip slenkstinė vertė, nuo kurios galima laikyti, kad manevras buvo atliktas.

1 lentelė. Paskaičiuotos kros-koreliacijos pikų vertės prie 3 manevrų

Testo nr.	Koreliacinio piko vertė						Šuolis nuo bortelio Acc z
	3 kliūčių apvažiavimas			2 kliūčių apvažiavimas			
	Acc y	Gyro x	Gyro z	Acc y	Gyro x	Gyro z	
1	67.851	577450	4451400	31.465	369080	3905100	67.104
2	61.499	649119	4409910	29.853	380770	3839100	80.050
3	57.224	645440	4190700	29.517	336820	3841200	66.579
4	67.908	582780	4542400	36.828	340990	3912600	81.971
5	66.141	639040	4475100	33.589	444030	4103200	83.841
6	62.115	606450	4543200	44.973	350270	4310200	87.790
7	58.184	652610	4487400	36.224	392640	4110200	78.369
8	66.713	612640	4583500	41.831	377400	4269600	74.834
9	57.068	610620	4351300	36.170	376270	4014900	67.442
10	60.634	618960	4247300	38.987	357800	3808700	69.030

Taip pat, naudojant 3 ašis vietoj 1 identifikuoti manevrą padidina teisingo aptikimo tikimybę. Tikrinant visas 3 ašis ir momentus, kada aptinkama aukšta koreliacija, galima priimti daugumos atsakymą. Jei akcelerometro ašis fiksuoja tik 2 maneversus, o abi giroskopo ašys 3, tai galima teigti, kad akcelerometras nesugebėjo užfiksuoti dar vieno manevro arba jis buvo arti slenkstinės vertės.

Šis metodas yra tinkamiausias aptikti signalo vėlinimus turint žinomą signalą, taip pat jam neturi įtakos triukšmas, kadangi vertės yra vidurkinamos. Tačiau šis metodas atlieka daug sumavimo ir dauginimo operacijų, taip pat jam reikalingas jau sukauptas buferis su duomenimis, bei privalu laikyti ieškomą signalą, kas didina atminties resursų eikvojimą.

2.8.2. Signalo energijos metodas

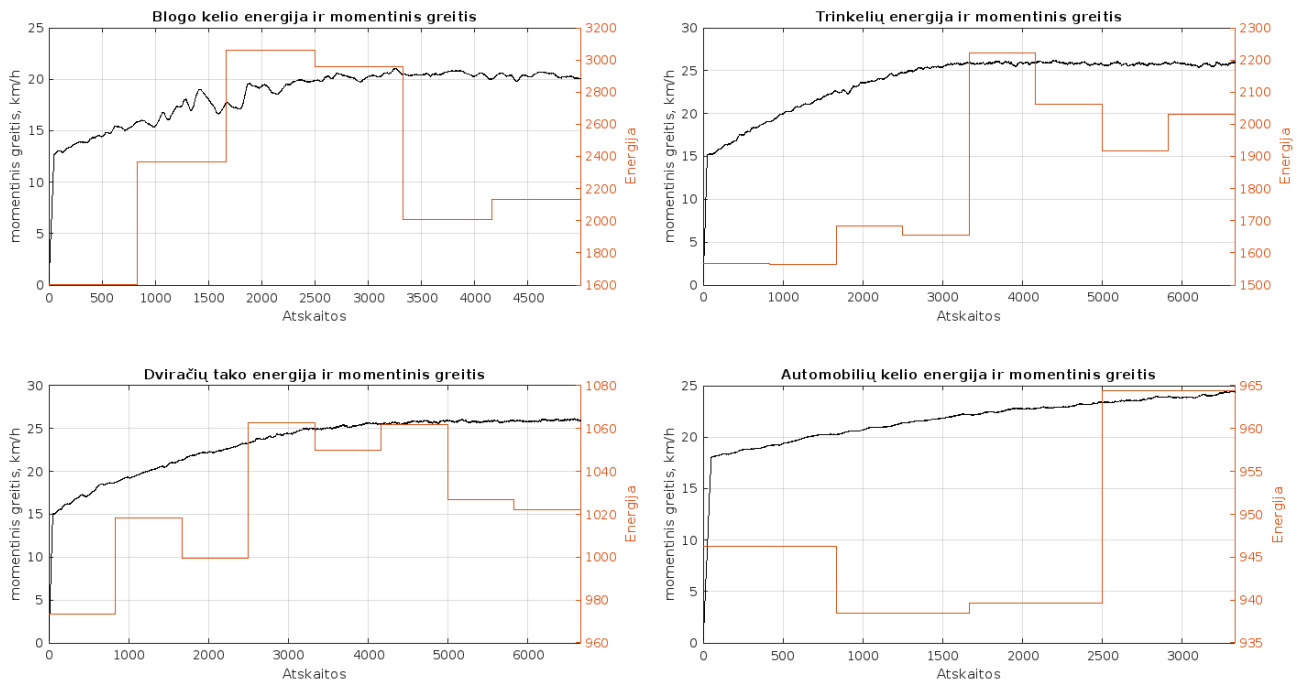
Šis algoritmas tinkamas nustatyti blogos kelio dangos manevrą, kai vartotojas važiuoja tam nepritaikytu keliu, taip sukeldamas sau pavojų. Metodas naudoja signalo energijos lygį, kuris paskaičiuojamas pagal žemiau pateiktą formulę [33]:

$$E_x = \sum_n |x[n]|^2 \quad (7)$$

Signalų energija priklausoma nuo amplitudės vertės, vadinasi, kuo didesnė amplitudė, tuo didesnę energiją signalas neša. Važiuojant nelygiu ar duobėtu keliu, geriausiai tai perteikia akcelerometro Z ašis, kadangi ji yra statmena važiuojamam pagrindui ir reaguoja į sukiamas vibracijas. Kuo kelias bus nelygesnis, tuo daugiau vibracijų fiksuos akcelerometras ir signalo amplitudė šioje ašyje šokinės labiau. Kadangi signalo energija turėtų būti matuojama viso važiavimo metu, skaičiuojant bendrą energiją būtų sunku išskirti, jei kelionės metu buvo važiuota prastu keliu. Sprendimas yra matuoti energiją langų pagalba. Iš signalo išskirti numatyto dydžio langą ir energiją matuoti tame lange. Tai leis detalčiau ir tiksliau sekti energiją bei konkrečiais momentais nustatyti, ar buvo važiuojama tinkama kelio danga ar ne. Buvo tiriami nefiltruoti akcelerometro Z ašies duomenys važiuojant 4 kelio dangomis kartu su momentiniu greičiu:

- blogo kelio (labai nelygus ir duobėtas asfaltas ir miško keliukas su daug šaknų ir akmenų);
- pėsčiųjų takas grįstas trinkelėmis;
- dviračių takas;
- automobilių važiuojamoji dalis.

Pradžioje buvo nustatytas optimalus lango ilgis. Kadangi važiuojant paspirtuku kelio danga nesikeičia greitai, o visi procesai vyksta lėtai, lango ilgis turėtų būti ganėtinai didelis. Didinant lango dydį, didėja ir paskaičiuotos energijos vertė, taip pat ji labiau atspindi kelio dangos energiją, o ne pasitaikiusių objektų ar manevrų, tačiau ilgesnis langas lemia ilgesnį buferio dydį ir papildomus atminties resursus. Pradžioje buvo eksperimentuojama su 0,5Fs lango dydžiu, tai atitinka 500 ms. Tačiau toks lango dydis lėmė tai, kad paskaičiuotai energijai didelę įtaką darė įvairūs objektai, kaip šulinių dangčiai ar menki dangos defektai. Didinant lango dydį iki Fs, 2Fs ir 3Fs buvo nustatyta, kad ties 2Fs gaunami geriausi rezultatai. Visų eksperimentų metu buvo greitėjama, taigi stebint rezultatus matyti, kad ir nuo greičio priklauso signalo energija. Kuo didesnis greitis, tuo didesnės vibracijos ir signalo energija. Gauti greičio ir energijos lygių palyginimai pateikti žemiau:



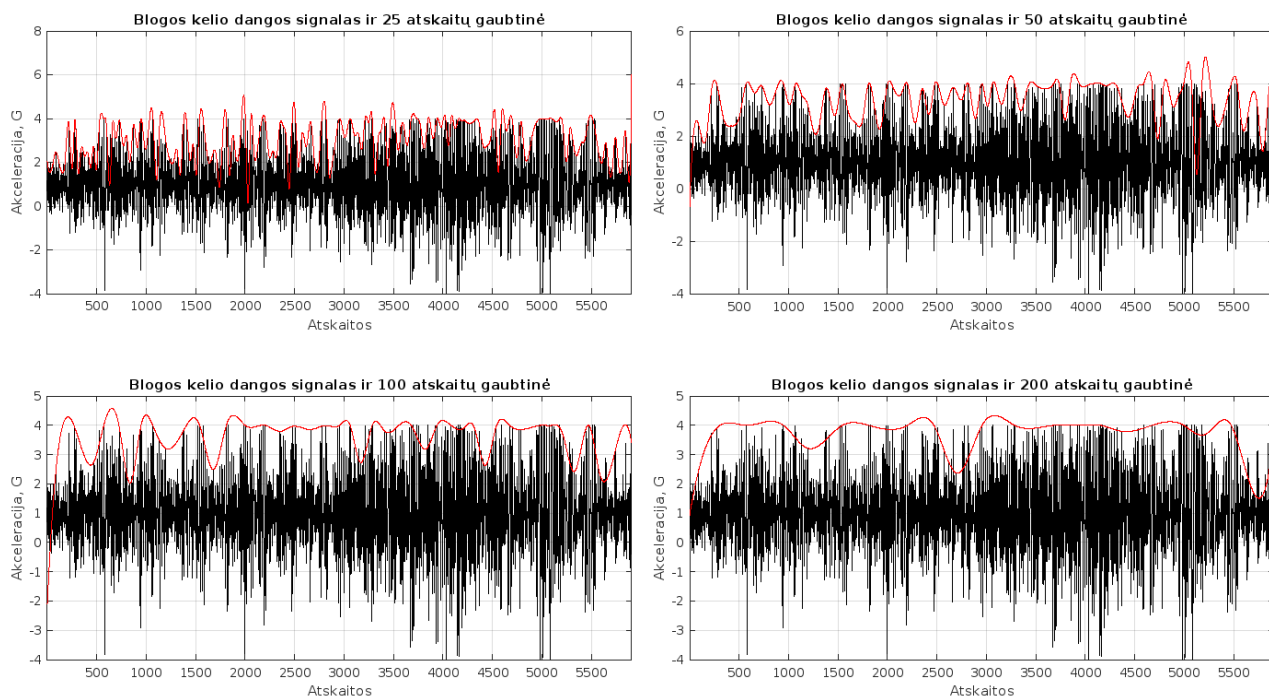
43 pav. Energijos vidutinės vertės priklausomybė nuo kelio dangos

Iš rezultatų matyti, kad važiuojant dviračių taku ir automobilių keliu energija signale yra 1000 skaitmeninių vienetų eilės. Ji nežymiai kyla didėjant greičiu. Blogo kelio energija yra žymiai didesnė lyginant su šiomis dviem kelio dangomis, tačiau panaši energija gaunama ir trinkelė kelio dangos eksperimente. Tai yra problema, kadangi didėjant greičiu trinkelė energijos lygis panašėja į prasto kelio ir juos gali būti sunku atskirti. Taip pat svarbu atkreipti dėmesį į momentinį greitį, važiuojant mažu greičiu net ir prastu keliu gali būti saugu ir nesukelti pavojaus vairuotojui. Pavojinga tampa greičiui didėjant, atliekant eksperimentus, ties 15 km/h greičiu ir daugiau, blogame kelyje buvo juntamos stiprios vibracijos ir vairuoti buvo sudėtinga. Greitis dažnai buvo primažinamas dėl saugumo, todėl rezultate matomas greičio šokinėjimas ir dėl to nepasiekiamas maksimalus paspirtuko greitis (maksimalus gamyklinis greitis: 25 km/h). Iš šių testų ir subjektyvios nuomonės buvo nustatyta

15 km/h riba, ties kuria bus tikrinama signalo energija. Važiuojant mažesniu greičiu net ir prasta danga galima traktuoti, kaip saugų ir nepavojingą manevrą. Toliau buvo pamatuota vidutinė mažiausia energijos vertė ties šiuo greičiu, tai buvo 2332. Ji yra didesnė už pamatuotus didžiausius trinkelį dangos energijos lygius, tačiau ganėtinai artima, kas gali lemti neteisingus manevro aptikimus važiuojant trinkelėmis ir atliekant kitus manevrus. Šis metodas yra nesudėtingas ir greitai, skaičiuojamas, jo algoritmas yra lengvas, reikalinga tik kėlimo kvadratu ir sumavimo operacija, duomenų specialiai apdoroti nereikia, tačiau reikia stebėti tiek momentinį greitį, tiek energijos lygį.

2.8.3. Gaubtinės analizės metodas

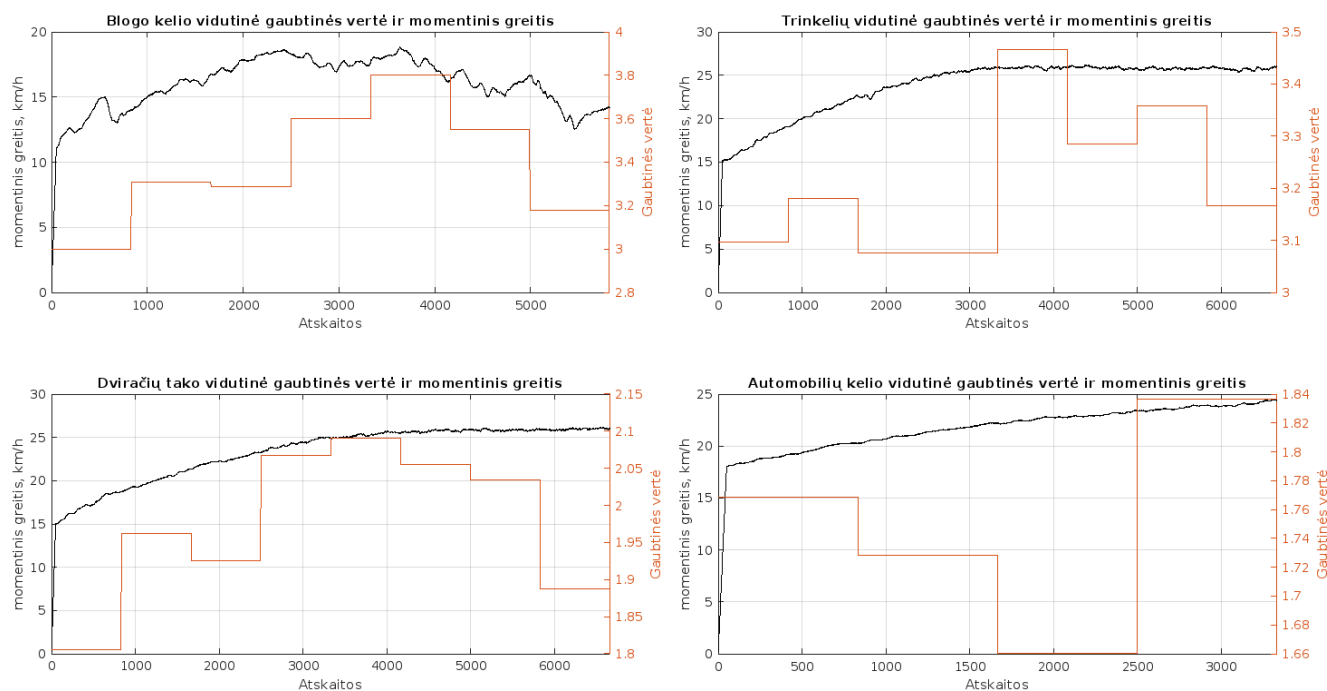
Šis algoritmas naudoja signalo gaubtinę ir jos vidutinę reikšmę aptikti greito važiavimo bloga kelio danga manevrą. Gaubtinė yra kreivė, kuri apibrėžia maksimalius signalo amplitudės pokyčius. Ši gaubtinė išlygina įvairių vibracijų duomenis ir geriau perteikia amplitudės pokyčius laike. Gaubtinės pagalba dažniausiai išgaunami moduluoti signalai, tai signalai, kurie sudaryti iš dviejų dažnių - nešlio ir moduluojančio aukštadažnio signalo [34]. Dažniausias metodas išgauti gaubtinei yra Hilberto moduliacija. Gaubtinė yra lygi absoliutinei analitinio signalo vertei, o ji gaunama tiriamo signalo ir jo Hilberto transformacijos rezultato sumai. Tačiau šis metodas tinka tik tada, kai tiriamas signalas yra sudarytas iš dviejų, skirtingų dažnių signalų. Tyrimo duomenis galima laikyti, kaip atsitiktinį triukšmą, nes jie neturi jokių aiškių dažnio verčių ar harmonikų, taigi šis metodas netinkamas. Dar vienas metodas yra naudoti maksimalias vertes ir interpoliaciją. Pradžioje surandami maksimalūs taškai, kurie yra didesni už kaimyninius taškus. Šie taškai atrenkami tam tikru užduotu atskaitų skaičiumi. Didinant šį skaičių, gaubtinė bus mažiau detali ir neapglėbs staigių amplitudės fliuktuacijų, mažinant šį skaičių, gaubtinė panašės į originalų signalą. Gaubtinės priklausomybė nuo atskaitų skaičiaus pateikta žemiau:



44 pav. Blogos kelio dangos signalo gaubtinės atskaitų tarpų palyginimai

Didinant atskaitų skaičių gaubtinė praranda detalumą ir neperteikia signalo formos, per mažas atskaitų skaičius interpoliuos gaubtinę, kuri yra labai artima originaliam signalui. Iš rezultatų buvo pasirinkta 50 atskaitų vertė. Turint gaubtinę sekantis žingsnis jos vidutinės reikšmės skaičiavimas.

Kaip jau minėta, paspirtuko manevrai yra lėti procesai, taigi vidutinė vertė turėtų būti skaičiuojama ilgame periode. Eksperimentų metu buvo nustatyta, kad 2Fs ilgio gaubtinės vidurkis duoda geriausius rezultatus. Vidutinės gaubtinės vertės su momentinio greičio grafiku pateikti žemiau:



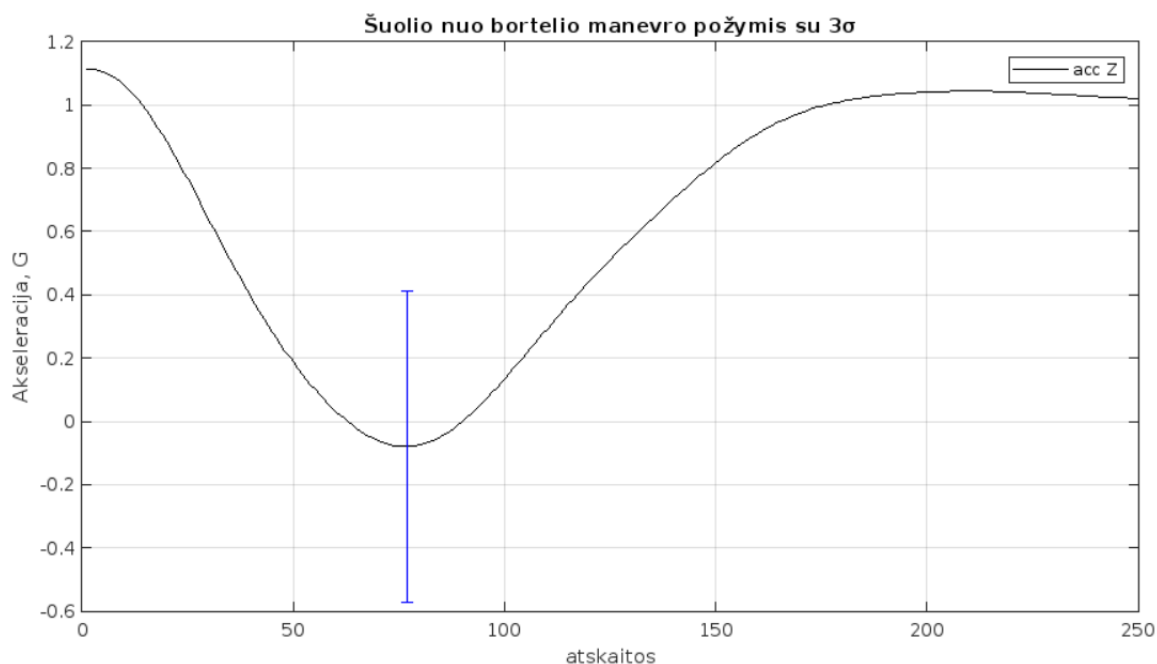
45 pav. Gaubtinės vidutinės vertės priklausomybė nuo kelio dangos

Iš rezultatų matyti, kad gaubtinės vidutinės vertės kitimas yra panašus į energijos lygio kitimą. Trinkelių didžiausia gaubtinės vertė yra ganėtinai arti blogo kelio gaubtinės vertėms, kas gali sukelti klaidingus manevro aptikimus. Šis metodas labai panašus į energijos lygio metodą, tačiau šiuo atveju yra daugiau parametrų, kurios galima reguliuoti ir derinti rezultatus. Tačiau jo minusas yra žymiai didesnis operacijų kiekis, taip pat reikalingos ganėtinai sudėtingos operacijos, tokios kaip interpoliacija.

2.8.4. Slenkstinės vertės metodas

Šis algoritmas pagrįstas slenkstinės vertės nustatymu. Slenkstinė vertė – tai tam tikrais metodais gauta vertė, kuri žymi kažkokio proceso tam tikrą ribą (angl. *Threshold*). Slenkstinės vertė gali būti labai aiški, po kurios labai staigiai įvyksta kažkoks procesas, arba gali būti labai abstrakti ir kintanti, kuriai nustatyti reikalingas vidurkis ar kitoks metodas. Stebint kažkokio proceso vertes galime pasinaudoti slenkstinėmis vertėmis nustatyti, kokiame etape yra procesas. Panašiu principu galima pasinaudoti ir manevrų identifikavimo taikyme. Akcelerometro vertės yra dažnai kintančios ir iš slenkstinių verčių nustatyti įvairius sudėtingus manevrus būtų netikslingą, kadangi jie apibūdinami ne tik jutiklio duomenų vertėmis, bet ir signalo forma bei trukme. Tačiau, toks manevras, kaip šuolis paspirtuku nuo bortelio, gali labai nesunkiai būti aptiktas būtent šiuo metodu. Dėl akcelerometro veikimo principo, jis visada fiksuoja žemės traukos jėga bent vienoje iš ašių arba ašių kelių sumoje, jei akcelerometro ašys nėra tobulai statmenos žemės paviršiui. Ši trauka yra lygi $\sim 1 G$, tyrimo stendo atveju, Z ašis visada jus šią jėgą ir naudojantis paspirtuku visada bus matoma nuolatinė dedamoji Z ašyje. Egzistuoja tik 2 atvejai, kai nuolatinės dedamosios nebus duomenyse: kai paspirtukas bus ne važiuojamoje padėtyje arba kai paspirtukas patirs laisvąjį kritimą. Laisvasis kritimas – tai kūno kritimas, kai jį veikia tik laisvasis pagreitis, kuris yra lygus $1 G$. Vadinasi laisvai krentantis

paspirtukas Z ašyje fiksuos laisvojo kritimo jėga, kuri kompensuos gravitacijos traukos jėga ir rezultate bus matomas 0. Būtent tai šiuo metodu ir galima fiksuoti. Šokant nuo šaligatvio bortelio paspirtukas labai trumpą laiko tarpą patirs laisvąjį kritimą ir jo akcelerometro Z ašies duomenys bus mažesni nei 1 G, dėl įvairių paklaidų, greičio bei papildomų pašalinių jėgų šis dydis nebus lygus 0, tačiau bus gerokai mažesnis už 1. Ribinė vertė padės nustatyti, ar paspirtukas atlieka šuolį ar ne. Jeigu vertė bus mažesnė nei ribinė vertė, galima traktuoti, kad šuolis buvo atliktas, visais kitais atvejais tai reiktų laikyti kaip įprastą ir nepavojingą važiavimą. Slenkstinei vertei nustatyti buvo pasinaudota šuolio nuo bortelio požymio signalas. Kadangi tai vidutinis manevras, jo žemiausia vertė yra vidutinė visų važiavimų vertė, tačiau parinkus ją, kaip slenkstinę ribą, nebūtų tikslu, kadangi tai nėra mažiausia galima vertė. Čia į pagalbą gali ateiti standartinės deviacijos skaičiavimai, kadangi per 1σ nuo vidutinės vertės gali būti nutolusios 68,2 % visų verčių dėl įvairių atsitiktinių veiksnių ir paklaidų, tai panaudojus šį dydį, kaip slenkstinę vertę algoritmas teoriškai gebės aptikti 68,2 % manevrų. Didinant σ atstumą nuo vidutinės vertės sudaromi didesni rėžiai į kuriuos patenka vis daugiau matavimų su atsitiktinai išsibarsčiusiomis vertėmis. 3σ atveju būtų sudaromi rėžiai, kuriuos patektų 99,6 % visų verčių, kas yra labai aukštas tikslumas. Taigi buvo pasirinkta $x_{min} + 3\sigma = -0,0797 + 0,49 = 0,4103 G$. Pasiėkus šią ribą galima laikyti, kad patiriamas laisvasis kritimas ir buvo atliktas šuolis.



46 pav. Ribinės vertės vizualizacija

Kadangi akcelerometro Z ašies duomenys šuolio metu labai stipriai kinta, svarbu atlikti duomenų filtravimą prieš vertės tikrinimą. Tuo tikslu yra naudojamas 50 eilės MA filtras. Šis metodas yra labai paprastas ir nereikalauja daug skaičiavimų, tai elementarus kiekvienos vertės lyginimas su slenkstine riba. Algoritmo esmė nėra aptikti žemiausią vertę, bet aptikti Z ašies duomenų kritimą link 0 ribos. Su šiuo metodu nustatoma vertė, nuo kurios galima teigti, kad manevras yra atliekamas. Šiam metodui reikia tik 1 vertės, taigi jis nereiklus atminties resursams. Tačiau jis nėra atsparus triukšmams, taigi būtinas filtravimas, taip pat jis negali atskirti atsitiktinės vertės kritimo nuo laisvojo kritimo manevro. Jei dėl kelio dangos ar nelygumų Z ašies duomenys staiga pakristų žemiau nustatytos ribinės vertės, galėtų būti fiksuotas klaidingas manevras.

2.8.5. Dinaminio laiko deformavimo metodas

Paskutinis metodas, tai yra dinaminis laiko deformavimo metodas (angl. *Dinamic Time Warping*). Šis metodas pradžioje buvo naudojamas balso atpažinime tačiau dabar jis plačiai naudojamas įvairiuose laiko srities signalų palyginimo uždaviniuose. Algoritmo esmė yra surasti optimalų signalų sulyginimą pagal jų formą, net jei signalai skiriasi dažniu ar yra prasistūmę laike, taip pat signalų ilgiai gali skirtis. Algoritmas gali naudoti įvairius taškų panašumo palyginimo metodus, esmė, kad taškų panašumas skaičiuojamas ne tik konkrečiu momentu laike, bet visame signalo ilgyje, su kiekvienu tašku. Tarp metodų, dažniausia naudojami euklidinis ar absoliutinis. Panašumas arba vienodumas tarp taškų yra nustatomas pagal formulę žemiau [35]:

$$D_{i,j} = |x_i - y_j| + \min \begin{cases} D_{i-1,j-1} \\ D_{i-1,j} \\ D_{i,j-1} \end{cases} \quad (7)$$

čia:

D – deformavimo koeficientas;

x_i – pirmo signalo vektoriaus atskaita i laiko momentu;

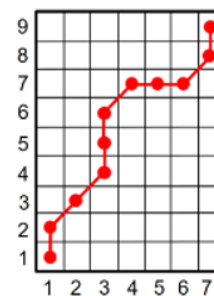
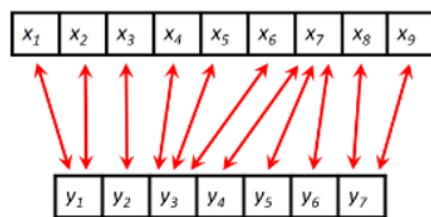
y_j – antro signalo vektoriaus atskaita j laiko momentu;

Tokiu metodu palyginamas atstumas tarp visų taškų. Rezultate gaunama D(i, j) matrica, sudaryta iš deformavimo konstantų. Šios vertės žymi suminį atstumą, reikalingą sulygtinti signalus iki tos vertės taško.

2		10	10	10	14	16	8	6	7
1		9	9	11	12	10	6	7	7
-3		9	9	11	9	5	8	13	14
0		5	5	5	4	8	8	10	10
4		4	4	2	7	13	15	13	14
2		1	1	1	5	11	12	11	12
1		0	0	2	5	10	11	12	12
y(j)									
	x(i)	1	1	3	-2	-4	0	2	1

47 pav. DTW atstumų matricos ir optimalaus kelio pavyzdys

Turint supildytą atstumų matricą, sudaromas optimalus kelias, kuris parodo pigiausią ir paprasčiausią dviejų sekų sulyginimo eigą. Kelias pradedamas viršuje dešinėje esančioje celėje (i, j) ir baigiamas apačioje kairėje (1, 1). Renkantis cele svarbu atkreipti dėmesį, kad ieškoma pigiausio ir mažiausio atstumo turinčio varianto, taigi renkama mažiausia vertė. Taip pat keliavimas per matricą vyksta tuo pačiu principu, kaip ir skaičiuojant atstumus: galima rinktis tarp (i – 1, j), (i – 1, j – 1) ir (i, j – 1) celių. Vadinaisi kelias negali grįžti atgal, užsisukti ar nutrūkti. Sudarytas kelias atspindi, koks vienos sekos taškas yra panašiausias ir suvienodintas, su kitos sekos tašku.



48 pav. DTW algoritmo rezultato pavyzdys [35]

Celėje (i, j) yra visos sekos suminis atstumas. Kuo šis dydis yra didesnis, tuo sekso yra mažiau panašios viena į kitą ir yra reikalinga didesnė kaina, norint jas suvienodinti. Dviejų identiškų signalų DTW grąžins 0, kadangi jų atstumai yra lygūs 0. „Matlab“ programiniame pakete šį algoritmą realizuoja *DTW()* funkcija, ji grąžina sekų suminį atstumą. Būtent šio dydžio pagalba ir buvo ieškoma požymių signale, bent kuriuo laiko momentu. Signalui esant panašiam į požymio signalą arba turint požymio manevro formą buvo gaunama maža DTW vertė, esant kitokiam manevrui ar manevro nebuvimo atveju, DTW vertė buvo kur kas didesnė. Signalai prieš tikrinimą buvo filtruojami MA 50 eilės filtru, taigi išgauti labai tikslų ir precizišką rėžių, kuriame varijavo DTW vertės esant manevrui, nebuvo įmanoma. Gautos vertės surašytos į lentelę ir iš jų pasirinktos didžiausios vertės, kurios bus tam tikro manevro DTW ribinė vertė.

2 lentelė. Paskaičiuotos DTW vertės prie 3 manevrų

Testo nr.	DTW atstumo vertė						
	3 klišių apvažiavimas			2 klišių apvažiavimas			Šuolis nuo bortelio
	Acc y	Gyro x	Gyro z	Acc y	Gyro x	Gyro z	Acc z
1	54.811	8187.291	4716.013	92.914	4459.140	9928.828	64.173
2	39.584	3971.080	5563.667	87.130	6885.540	10331.382	54.680
3	30.765	3971.080	2079.788	109.877	4911.614	6205.503	43.223
4	45.171	5659.669	5983.266	68.718	7717.877	7602.360	51.547
5	55.920	9782.751	7866.094	42.875	8734.647	4857.681	54.187
6	44.932	7607.424	2744.989	149.207	8016.222	12116.296	59.227
7	35.322	5148.408	6201.706	124.683	5769.707	9881.039	39.163
8	55.259	4696.286	6378.510	140.302	6191.159	6348.314	58.054
9	41.201	5602.256	5612.289	136.313	4894.576	8552.668	53.220
10	63.110	7699.704	11192.417	115.769	6915.552	9099.225	93.513

Pavojingio manevro aptikimo principas buvo naudojamas kaip ir koreliacijos metode. Tikrinant 3 ašis ir pagal daugumą nusprendžiant, manevras buvo atliktas ar ne. Šis metodas yra puikiai tinkantis aptikti skirtingo dažnio ar amplitudės signalų panašumą, jo rezultatai lengva interpretuoti. Tačiau šis metodas reikalauja dinaminio programavimo sprendimo norint sudaryti atstumų matricą, taip pat ganėtinai ilgo skaičiavimo, kadangi reikia atlikti $i*j$ sudėties ir atimties operacijų. Tačiau ieškant tik suminio sekų atstumo, galima praleisti optimaliausio kelio paiešką, jei aktuali tik žinoti, kaip labai signalai panašūs.

3. Eksperimentų rezultatai

Paskutinėje dalyje pateikiami rezultatai, gauti iš visų, anksčiau aptartų, algoritmų panaudojimo. Testai atliekami tomis pačiomis sąlygomis, kaip ir požymių manevrų įrašymo metu, važiuojant elektriniu paspirtuku ir įrašant tam tikrus, testavimui skirtus maneversus. Rezultatai sudaryti iš dviejų tipų eksperimentų:

1. metodas bando aptikti manevrą, kai jis signale yra atliktas;
2. metodas bando aptikti manevrą, kai jo signale nėra.

pirmuoju atveju tikrinamas metodo gebėjimas aptikti manevrą, kai jis tikrai yra įrašė. Antruoju atveju tikrinamas metodo gebėjimas neaptikti manevro, kai jo nėra įrašė.

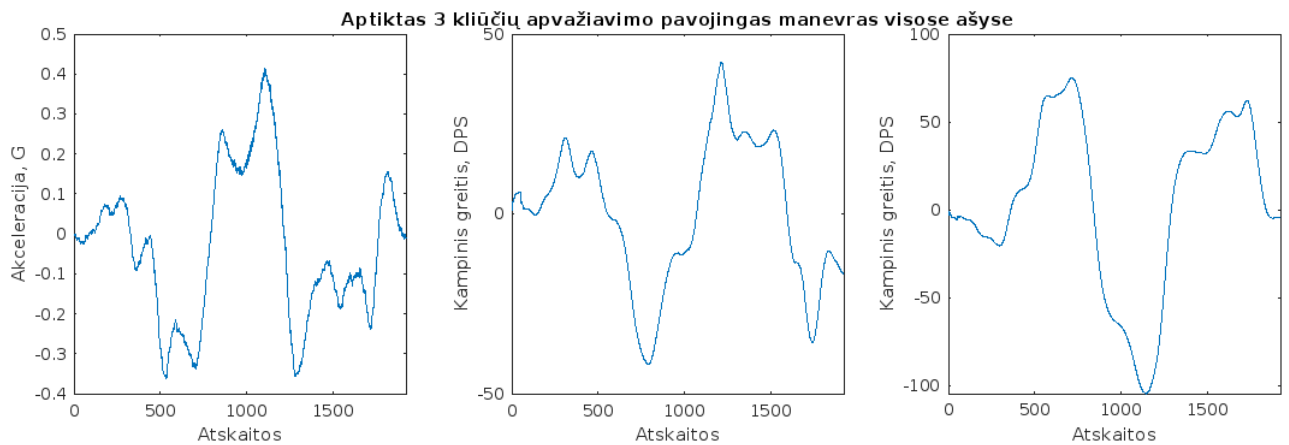
3.1. Metodų gebėjimas teisingai aptikti manevrą

Kiekvienam iš tirtų manevrų buvo atliekama 10 ir daugiau įrašų, kuriuose buvo užfiksuotas tas maneversas. Įrašo metu buvo stengiamasi kiekvieną įrašą atlikti kuo natūralesnį ir unikalesnį, tai leidžia objektyviai įvertinti algoritmo veikimą. Metodo patikimumas išreiškiamas procentais, skaičiuojama, kiek manevrų aptiko iš visų eksperimentų, kuriuose tikrai yra manevro įrašas. Pirmoje lentelėje pateikti duomenys iš 35 eksperimentų.

3 lentelė. 3 kliūčių ir 2 kliūčių apvažiavimo manevro metodų rezultatai

Maneversas	Tikrinta jutiklio ašis	Algoritmas	
		Kros-koreliacija	DTW
3 kliūčių apvažiavimas	acc y	84.78%	52.96%
	gyro x	82.61%	75.10%
	gyro z	71.15%	44.07%
	bendras algoritmo tikslumas	84.78%	52.96%
2 kliūčių apvažiavimas	acc y	95.83%	100%
	gyro x	90.57%	84.87%
	gyro z	73.90%	73%
	bendras algoritmo tikslumas	93.20%	89.04%

Aukščiau pateikti 3 ir 2 kliūčių apvažiavimo manevrų tirtų metodų DTW ir kros-koreliacijos rezultatai. Iš rezultatų galima teigti, kad 3 kliūčių apvažiavimo manevrą geriausiai aptiko kros-koreliacijos metodas, jo bendras tikslumas net 84,78 %. Kadangi abu metodai nustato pavojingą manevrą, jei bent 2 ašys aptinka tą manevrą, iš DTW metodo rezultatų matyti, kad dėl akcelerometro Y ir giroskopo Z ašių šis algoritmas dažnai neužfiksavo manevro, kai jis ten buvo. 2 kliūčių apvažiavimo manevrą aptikti pavyko abejiems manevrams geriau. Bet tendencija išlieka ta pati, dažniausiai maneversus aptiko kros-koreliacijos metodas, tačiau šiuo atveju DTW metodo tikslumas buvo labai artimas, siekė 89,04 % lyginant su kros-koreliacijos 93.20%.



49 pav. Kros-koreliacijos metodu sėkmingai aptiktas manevras visose trijose ašyse

Toliau pateikiami šuolio nuo bortelio manevro rezultatai atlikus 20 eksperimentų.

4 lentelė. Šuolio nuo bortelio manevro metodų rezultatai

Manevras	Tikrinta jutiklio ašis	Algoritmas		
		Kros-koreliacija	DTW	Ribinės vertės metodas
Šuolis nuo bortelio	acc z	90.91%	77.27%	100%

Iš šių rezultatų buvo gautas 100% ribinės vertės metodo tikslumas, vadinasi jis visais atvejais aptiko šį manevrą įrašė. Taip pat iš rezultatų ryškėja, kad DTW metodas prasčiausiai aptinka maneversus ir geba išskirti jų pavojingumą. Kros-koreliacijos metodas išlieka aukšto tikslumo ir šiuo atveju. Paskutiniai manevrai, tai blogo ar netinkamo kelio dangos nustatymas atlikus 20 eksperimentų.

5 lentelė. Blogo kelio dangos nustatymo metodų rezultatai

Manevras	Tikrinta jutiklio ašis	Algoritmas	
		Vidutinės energijos lygio vertė	Gaubtinės vidutinė vertė
Blogo kelio dangos nustatymas	acc z	100.00%	100.00%

Šie rezultatai parodo, kad tiek vidutinės energijos lygio, tiek gaubtinės vidutinės vertės metodai visais atvejais gebėjo aptikti prastą kelio dangą. Iš gautų manevro atpažinimo rezultatų buvo nustatyta, kad tiksliausiai veikė metodai, kurie tikrina tik vieną signalo požymį (energijos lygį, gaubtinės dydis ar signalo vertės dydis). Algoritmai, kurie naudojo charakterizuotus manevro formos signalus, veikė prasčiau ir turėjo mažesnę tikslumą.

3.2. Metodų atsparumas klaidingiems manevrų aptikimams

Labai svarbus metodo bruožas atpažinti manevrą, tačiau ne mažiau svarbus ir gebėjimas nefiksuoti manevrų, kai jų ten nėra. Jei algoritmas turi 100 % tikslumą manevro atpažinime, būtina žinoti ir jo klaidingų aptikimų (angl. *false trigger*) procentą. Šiems rezultatams gauti buvo atliekama įvairių važiavimo eksperimentų su įprastais mikromobilumo priemonės naudojimo metu sutinkamais manevrais:

- važiavimas dviračių taku atliekant posūkius;
- važiavimas įvairiomis trinkelėmis atliekant posūkius;

- važiavimas automobilių keliu atliekant posūkius.

Taip pat buvo tikrinami ir turimi kiekvieno manevro įrašai. Žemiau pateikti visų metodų klaidingai aptikti manevrai išreikšti procentais iš 100 įvairių eksperimentų įrašų:

6 lentelė. Visų manevrų klaidingų aptikimų rezultatai

Manevras	Tikrinta jutiklio ašis	Algoritmas				
		Kros-koreliacija	DTW	Ribinės vertės metodas	Gaubtinės vidutinė vertė	Vidutinės energijos lygio vertė
3 kliūčių apvažiavimas	acc y	12.84%	12.84%	-	-	-
	gyro x	17.43%	24.77%	-	-	-
	gyro z	4.58%	8.25%	-	-	-
2 kliūčių apvažiavimas	acc y	24.1%	27.77%	-	-	-
	gyro x	19.44%	16.67%	-	-	-
	gyro z	7.4%	6.48%	-	-	-
Šuolis nuo bortelio	acc z	0%	91.34%	5.77%	-	-
Blogo kelio dangos nustatymas	acc z	-	-	-	9.73%	52.21%

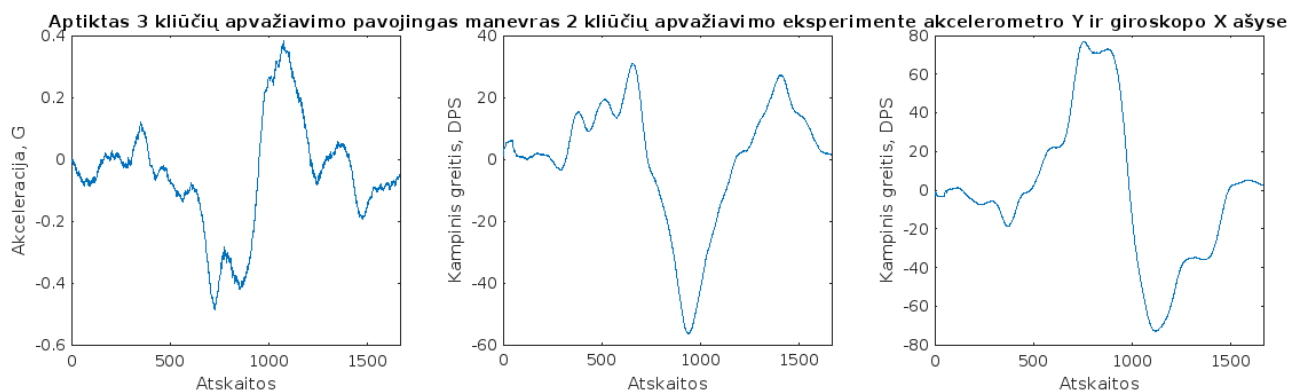
Iš aukščiau pateiktų rezultatų galima teigti, kad DTW metodas nustatyti šuoliui nuo bortelio yra pats nepatikimiausias metodas. Atliekant rezultatų skaičiavimus buvo gaunami manevrų fiksavimai visuose eksperimentų tipuose. Šis metodas netinkamas aptikti staigiai krentančios amplitudės požymį, nes daugumą filtruotų įrašų turi tokios formos signalą. Pats patikimiausias metodas taip pat gautas šuolyje nuo bortelio manevre, tai yra kros-koreliacijos metodas. Šis algoritmas nei karto nefiksavo manevro, kai jo nebuvo. 3 kliūčių apvažiavimo manevre dažniausiai klydo DTW algoritmo gioskopo X ašis, tačiau tiksliausiai dirbo kros-koreliacijos gioskopo Z ašis. 2 kliūčių apvažiavime prasčiausi rezultatai buvo gauti DTW algoritmo akcelerometro Y ašies, geriausi – DTW algoritmo gioskopo Z ašies. Blogo kelio aptikime labai prasti rezultatai vidutinės energijos lygio metode, ženkliai geresni – gaubtinės vidutinės vertės metode.

Toliau buvo tiriama, kaip dažnai 3 kliūčių apvažiavimo požymiai aptikti 2 kliūčių apvažiavimo eksperimentuose ir atvirkščiai. Buvo lyginami 23 eksperimentų duomenys.

7 lentelė. 3 kliūčių požymių aptikimo 2 kliūčių apvažiavimo eksperimentuose rezultatai

Manevras	Tikrinta jutiklio ašis	Algoritmas	
		Kros-koreliacija	DTW
3 kliūčių apvažiavimas	acc y	21.79%	13.04%
	gyro x	43.48%	65.22%
	gyro z	8.70%	0%
2 kliūčių apvažiavimas	acc y	87.50%	70.83%
	gyro x	75%	29.17%
	gyro z	0%	0%

Iš šių rezultatų pavyko nustatyti, kad DTW metodo gioskopo X ašis dažniausiai fiksavo 3 kliūčių apvažiavimą, kai buvo apvažiuojamos 2 kliūtys. Tiksliausiai veikė gioskopo Z ašis, ji nei karto nefiksavo manevro. Ieškant 2 kliūčių apvažiavimo manevro, kai tikrinami 3 kliūčių apvažiavimo signalai, buvo gauta, kad kros-koreliacijos akcelerometro Y ašis šį manevrą fiksavo dažniausiai. Tiek DTW, tiek kros-koreliacijos gioskopo Z ašis nefiksavo manevro nei karto.



50 pav. DTW metodu klaidingai aptiktas manevras dvejuose ašyse

8 lentelė. 3 ir 2 kliūčių apvažiavimo manevrų aptikimo, kai amplitudė yra mažesnė, rezultatai

Manevras	Tikrinta jutiklio ašis	Algoritmas	
		Kros-koreliacija	DTW
3 kliūčių apvažiavimas	acc y	69.23%	76.92%
	gyro x	61.54%	92.31%
	gyro z	23.08%	69%
2 kliūčių apvažiavimas	acc y	36.36%	100%
	gyro x	18%	100%
	gyro z	73%	64%

Šiame tyrime buvo naudojami signalai, kai manevrai atliekami apvažiuojant kliūtis mažesniu, nei nustatytas, atstumu: 3 kliūčių apvažiavime – 30 cm ir 2 kliūčių apvažiavime – 100 cm. Buvo naudota 12 įrašų signalus. Šis testas parodo, kaip algoritmas geba išskirti pavojingą nuo nepavojingo manevro, nors jo forma yra panaši. Rezultatuose matyti, kad DTW metodo akcelerometro Y ir giroskopo Z ašys veikė prasčiausiai. Taip pat buvo tiriama, kaip dažnai algoritmai fiksuoja manevrus, kai važiuojama tvarkingai, atliekant nepavojingus posūkius, apsisukimus ar kitus įprastus manevrus. Buvo tiriama 50 įrašų.

9 lentelė. 3 ir 2 kliūčių apvažiavimo manevrų aptikimo esant tvarkingam važiavimui rezultatai

Manevras	Tikrinta jutiklio ašis	Algoritmas	
		Kros-koreliacija	DTW
3 kliūčių apvažiavimas	acc y	0%	2%
	gyro x	2%	0%
	gyro z	0%	0%
2 kliūčių apvažiavimas	acc y	2%	4%
	gyro x	2%	0%
	gyro z	0%	0%

Visais tirtais atvejais algoritmai ganėtinai retai fiksuoja manevrus. 3 kliūčių apvažiavimą fiksuoja tiek DTW tiek kros-koreliacijos algoritmai vienodai kartų skirtingose ašyse. 2 kliūčių apvažiavimą taip pat fiksuoja abu algoritmai vienodai, tačiau kros-koreliacijos akcelerometro Y ir giroskopo X ašis fiksuoja po 2 %, vadinasi šis metodas turi didesnę šansą registruoti manevrą, nes reikalinga aptikti manevrą 2 ašyse, kad jis būtų priimtas, kaip pavojingas. Toliau buvo atliktas bendras metodų klaidingų nustatymų tyrimas, kai priimamas sprendimas atsižvelgiant į 3 ašių rezultatus. Buvo naudoti 1 lentelėje testuoti signalai gauti algoritmų bendriems rezultatams.

10 lentelė. Bendri 3 ir 2 kliūčių apvažiavimo algoritmų klaidingų manevrų nustatymo rezultatai

Manevras	Algoritmas	
	Kros-koreliacija	DTW
3 kliūčių apvažiavimas	11.01%	12.84%
2 kliūčių apvažiavimas	19.44%	16.67%

Rezultatai buvo artimi ir skyrėsi kelių procentų režiuose, tačiau buvo nustatyta, kad rečiau 3 kliūčių apvažiavimą fiksavo kros-koreliacijos metodas, o 2 kliūčių apvažiavimą – DTW metodas.

11 lentelė. Blogos kelio dangos klaidingo nustatymo rezultatai

Manevras	Tikrinta jutiklio ašis	Algoritmas	
		Vidutinės energijos lygio vertė	Gaubtinės vidutinė vertė
Blogo kelio dangos nustatymas	acc z	82%	14%

Toliau buvo testuojami algoritmai, skirti nustatyti prastą kelio dangą, testuojami važiavimai trinkelėmis, dviračių taku bei automobilių keliu. Rezultatai buvo skaičiuojami iš 50 eksperimentų duomenų. Gauti rezultatai parodė tendenciją, kad energijos vertės nustatymo metodas fiksuoja prastą kelio dangą dažniausiai.

3.3. Rezultatų apibendrinimas ir pasiūlymai, kaip juos gerinti

Norint įvertinti manevro gebėjimą aptikti pavojingą manevrą svarbu ir jo tikslumas randant manevrą, ir jo patikimumas, manevro nefiksuoti, kur jo nėra. Iš gautų rezultatų, buvo nustatyta, kad pavojingą 3 kliūčių apvažiavimą geriausiai nustato kros-koreliacijos metodas. Šis metodas parodė geriausius rezultatus tiek bendrame manevro nustatyme, tiek klaidingų manevrų atpažinime. Taip pat šis algoritmas parodė geriausią kiekvienos ašies efektyvumą, tačiau tiek DTW, tiek kros-koreliacijos algoritme, prasčiausiai veikia giroskopo X ašis. 2 kliūčių apvažiavime efektyviausiai dirbo kros-koreliacija, tačiau šiuo atveju pastebėta, kad ašys veikia geriau prie vienu ar kitu scenarijų, į ką reiktų atsižvelgti gerinant sistemos veikimą. Šuolyje nuo bortelio geriausius rezultatus pateikė tiek ribinės vertės tiek kros-koreliacijos algoritmai. Abu šie metodai pateikė labai artimus ir panašius rezultatus, tiksliausiai manevrą nustatė ribinės vertės metodas, bet atspariausias netikriems manevrams buvo kros-koreliacija. Dėl resursų kaštų vertėtų rinktis ribinės vertės metodą, dėl jo paprasto realizavimo, bet vis tiek aukšto tikslumo. Blogo kelio nustatymui vienareikšmiškai geriausiai veikia vidutinės gaubtinės metodas, jis tiksliai nustatė esamą prastą kelio dangą ir ganėtinai retai fiksavo prastą dangą, kai buvo važiuojama įprastu keliu, tai galėjo lemti įvairūs kliūviniai, kaip šulinių dangčiai ar kelio nelygumai.

12 lentelė. Tinkamiausių naudoti algoritmų skirtingiems pavojingiems manevrams rezultatai

Pavojingas manevras	Tinkamiausias algoritmas
3 kliūčių apvažiavimas	Kros-koreliacija
2 kliūčių apvažiavimas	Kros-koreliacija
Šuolis nuo bortelio	Ribinės vertės metodas
Blogos kelio dangos nustatymas	Gautinės vidutinės vertės metodas

Norint pagerinti esamus rezultatus vertėtų atlikti didesnę eksperimentų kiekį modeliuojant algoritmą ir parinkti tinkamesnes ribines vertes, kada manevras turėtų būti fiksuojamas ir kada ne. Taip pat

didesnis imčių skaičius padėtų tiksliau charakterizuoti manevro požymius. Deviacija tarp imčių galėtų mažėti ir naudojant tikslesnius signalo apdorojimo metodus, vertėtų paieškoti geresnių filtravimo metodų. Taip pat labai svarbus eksperimento atlikimo atsikartojamumas, užtikrinant idealias sąlygas ir atkartojant manevrą kaskart vienodai mažins verčių išsibarstymą.

Išvados

1. Darbe buvo tiriami įvairūs mikromobilumo priemonės naudojimo pavojingų manevrų identifikavimo metodai. Išanalizuoti esami darbai susiję su manevrų atpažinimu, tyrinėta esama rinka, taip pat ieškoma jutiklių, tinkamų fiksuoti įvairius judesių duomenis. Kartu buvo ieškoma algoritmų, tinkamų aptikti signalų formas, jų trukmes ir požymius. Svarbus algoritmo pasirinkimo bruožas buvo gebėjimas jį realizuoti įterptinėje sistemoje, taigi buvo atsižvelgta į reikiamus realizavimo kaštus. Pasirinkti metodai ištestuoti „*Matlab*“ programinėje aplinkoje su realiais duomenimis. Charakterizuoti manevrų požymiai buvo įvertinti statistiniu deviacijos metodu, tai svarbu nustatant požymio tikslumą ir tai, kaip labai jis atspindi realius manevers. Gauti atpažinimo rezultatai palyginti tarpusavyje. Galiausiai buvo išrinkti geriausi metodai kiekvienam iš tirtų pavojingų manevrų;
2. Buvo panaudotos dvi duomenų surinkimo sistemos, tai mobiliojo telefono pagalba gauti akcelerometro duomenys ir sukurta įterptinė sistema, gebanti fiksuoti akcelerometro, giroskopo ir momentinio greičio duomenis. Buvo pasirinkta naudoti antrąją sistemą tolimesniam tyrimui, dėl jos parametrų paprastesnio valdymo, duomenų kiekio ir patikimumo;
3. Darbe ištirti įvairūs signalo filtravimo metodai, buvo parinkti optimaliausi filtrai požymio charakterizavimui ir duomenų paruošimui juos taikyti tirtiems algoritmams. Požymio charakterizavimui buvo panaudotas 3 Hz ŽD nulinio fazės poslinkio filtras, o signalams, taikytiems atpažinimo algoritmą - 50 eilės slenkamo vidurkio filtras;
4. Buvo paskaičiuotas gautų manevrų požymių statistinis parametras – standartinė deviacija. Iš rezultatų buvo nustatyta, kad mažiausiai nukrypęs, nuo naudotų verčių požymis yra giroskopo Z ašies. Manevro atpažinime ši ašis neparodė geriausių rezultatų manevro aptikime, tačiau parodė geriausius gebėjime nefiksuoti klaidingų manevrų. Pasitelkiant šuolio nuo bortelio požymio standartinę deviaciją buvo pasirinkta ribinė vertė, kuri parodė labai gerus rezultatus abejais testavimo atvejais. Prasčiausius rezultatus parodė giroskopo X ašies rezultatai, reikėtų atkreipti dėmesį į šios ašies požymio tikslinimą arba atsisakymą ir pakeitimą kita ašimi projektuojant realią sistemą;
5. Atlikus atpažinimo algoritmų testus ir realiomis sąlygomis išbandžius visus algoritmus buvo nustatyta, kad pavojingą 2 ir 3 kliūčių apvažiavimą geriausiai aptinka kros-koreliacijos metodas, su 84,78 % 3 ir 93.2 % 2 kliūčių apvažiavimo aptikimo procentu ir tik 11,01 % 3 ir 19.44 % 2 kliūčių apvažiavimo manevro aptikimu, kai jo signale nėra. Šuolio nuo bortelio manevrą geriausiai aptiko tiek kros-koreliacijos, tiek ribinės vertės metodai, tačiau dėl minimalaus skaičiavimo ir labai mažų atminties resursų eikvojimo buvo pasirinktas ribinės vertės metodas, su 100 % aptikimo ir 5.769 % klaidingo aptikimo procentu. Blogo kelio dangos nustatymui geriausius rezultatus parodė gaubtinės vidutinė vertės metodas, kuris pateikė 100 % tikslumą aptinkant blogą dangą ir 9,73 % aptinkant blogą dangą klaidingai.

Literatūros sąrašas

1. Jayana Locke. *What is Micromobility and what is the Market for Developers?* Interneto puslapis. Prieiga per: <https://www.digi.com/blog/post/what-is-micromobility>. [žiūrėta 2024-03-01].
2. Kazemzadeh Khashayar, Haghani Milad, Sprei Frances. *Electric Scooter Safety: An Integrative Review of Evidence from Transport and Medical Research Domains. Sustainable Cities and Society*. Interaktyvus, 2023, vol. 89. pp. 11. ISSN 2210-6707. Prieiga per: <https://www.sciencedirect.com/science/article/pii/S2210670722006175>. [žiūrėta 2024-03-01].
3. Townley Jay. *Micromobility Is the Future! – Human Powered Solutions*. Internetinė svetainė. Prieiga per: <https://humanpoweredolutions.com/2019/12/23/micromobility-is-the-future/>. [žiūrėta 2024-03-01].
4. Vectornav. *What is an Inertial Measurement Unit?* Interneto puslapis. Prieiga per: <https://www.vectornav.com/resources/inertial-navigation-articles/what-is-an-inertial-measurement-unit-imu>. [žiūrėta 2024-03-01].
5. Mohammadian Rad Nastaran. *Deep Learning for Abnormal Movement Detection using Wearable Sensors: Case Studies on Stereotypical Motor Movements in Autism and Freezing of Gait in Parkinson's Disease*. Daktaro disertacija. DEPARTMENT OF INFORMATION ENGINEERING AND COMPUTER SCIENCE ICT International Doctoral School, 2019. Prieiga per: https://www.researchgate.net/publication/337827590_Deep_Learning_for_Abnormal_Movement_Detection_using_Wearable_Sensors_Case_Studies_on_Stereotypical_Motor_Movements_in_Autism_and_Freezing_of_Gait_in_Parkinson%27s_Disease. [žiūrėta 2024-03-01].
6. Jeff Watson. *MEMS Gyroscope Provides Precision Inertial Sensing in Harsh, High Temperature Environments*. Interaktyvus. Prieiga per: <https://www.analog.com/en/resources/technical-articles/mems-gyroscope-provides-precision-inertial-sensing.html>. [žiūrėta 2024-03-01].
7. Jacob Moore, Majid Chatsaz, Agnes d'Entremont, Joan Kowalski, Douglas Mille. *Mechanics Map - Position Displacement Velocity Acceleration*. Prieiga per: http://mechanicsmap.psu.edu/websites/1_mechanics_basics/1-2_position/position.html. [žiūrėta 2024-03-03].
8. Zhao Ming, Jia Xiaodong, Lin Jing, Lei Yaguo, Lee Jay. *Instantaneous Speed Jitter Detection Via Encoder Signal and its Application for the Diagnosis of Planetary Gearbox. Mechanical Systems and Signal Processing*, 2018, vol. 98. pp. 17. Prieiga per: https://www.researchgate.net/publication/316632412_Instantaneous_speed_jitter_detection_via_encoder_signal_and_its_application_for_the_diagnosis_of_planetary_gearbox. [žiūrėta 2024-03-03].
9. United States, Department of Transportation. *How Alcohol Affects Driving Ability*. Interaktyvus. Prieiga per: <https://www.nhtsa.gov/>. [žiūrėta 2024-03-03].
10. Zhao Xiaohua, Zhang Xingjian, Rong Jian, Ma Jianming. *Identifying Method of Drunk Driving Based on Driving Behavior. International Journal of Computational Intelligence Systems*, 2012, vol. May 2011. pp. 9. Prieiga per: https://www.researchgate.net/publication/233352798_Identifying_Method_of_Drunk_Driving_Based_on_Driving_Behavior. [žiūrėta 2024-03-03].
11. James McGee. *5 Ways to Identify a Drunk Driver*. Interneto puslapis. Prieiga per: <https://www.mcgeepc.com/blog/2024/02/5-ways-to-identify-a-drunk-driver/>. [žiūrėta 2024-03-03].
12. Jin Teng, Xiaole Bai, Zhaohui Shen, Dong Xuan. *Mobile Phone Based Drunk Driving Detection*. ISBN 2153-1641. Prieiga per: <https://ieeexplore.ieee.org/document/5482295/citations#citations>. [žiūrėta 2024-03-03].

13. Sarmitė Mikulionienė, Gražina Rapolienė, Natalija Valavičienė. *Vyresnio Amžiaus Žmonės, Gyvenimas Po Vieną Ir Socialinė Atskirtis*. Mokslo monografija. Vilnius: 2018 Prieiga per: <https://www.lstc.lt/wp-content/uploads/2019/04/VIENASI.pdf>. [žiūrėta 2024-03-03].
14. Vavoulas George, Chatzaki Charikleia, Malliotakis Thodoris, Pediaditis Matthew, Tsiknakis Manolis. *The MobiAct Dataset: Recognition of Activities of Daily Living using Smartphones*. 2016. Prieiga per: <https://www.researchgate.net/publication/302973506> *The MobiAct Dataset Recognition of Activities of Daily Living using Smartphones*. [žiūrėta 2024-03-03].
15. Tensorflow. *An End-to-End Platform for Machine Learning*. Interneto puslapis. Prieiga per: <https://www.tensorflow.org/>. [žiūrėta 2024-03-03].
16. Kulwarun Warunsin, Thongchai Phairoh. *Wristband Fall Detection System using Deep Learning*. ISBN 978-1-6654-5060-7. Prieiga per: <https://ieeexplore.ieee.org/document/9846023>. [žiūrėta 2024-03-03].
17. Saini Anshul. *Decision Tree - A Step-by-Step Guide*. Forumas internete. Prieiga per: <https://www.analyticsvidhya.com/blog/2021/08/decision-tree-algorithm/>. [žiūrėta 2024-03-03].
18. IBM. *What is Linear Discriminant Analysis?* Interneto puslapis. Prieiga per: <https://www.ibm.com/topics/linear-discriminant-analysis>. [žiūrėta 2024-03-03].
19. *Support Vector Machine (SVM) Algorithm*. Forumas internete. Prieiga per: <https://www.geeksforgeeks.org/support-vector-machine-algorithm/>. [žiūrėta 2024-03-03].
20. Tavish Srivastava. *A Complete Guide to K-Nearest Neighbors*. Forumas internete. Prieiga per: <https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/>. [žiūrėta 2024-03-03].
21. Nugroho Hapsoro, Wicaksana Haryas, Hariyanto Hariyanto, Virgianto Rista. *Algorithm Performance Comparison for Earthquake Signal Recognition on Smartphone's Accelerometer*. TELKOMNIKA (Telecommunication Computing Electronics and Control), 2020, vol. 18. pp. 13. ISSN 1693-6930. Prieiga per: <https://www.researchgate.net/publication/344459436> *Algorithm performance comparison for earthquake signal recognition on smartphone's accelerometer*. [žiūrėta 2024-03-03].
22. *Neuron Introduces the World's most Sophisticated E-Scooter with Global Technology Trial!* Interneto puslapis. Prieiga per: <https://www.rideneuron.com/neuron-introduces-the-worlds-most-sophisticated-e-scooter-with-global-technology-trial/>. [žiūrėta 2024-03-07].
23. Włodzimierz Erdmann. *Center of Mass of the Human Body Helps in Analysis of Balance and Movement*. MOJ Applied Bionics and Biomechanics, 2018, vol. 2. pp. 6. Prieiga per: <https://www.researchgate.net/publication/327315249> *Center of mass of the human body helps in analysis of balance and movement/references*. [žiūrėta 2024-03-17].
24. Jan Urbancik. *Safety and Electric Scooters*. Interneto puslapis. Prieiga per: <https://www.maxblinker.com/en/info/escooter-guide>. [žiūrėta 2024-03-17].
25. *Average*. Forumas internete. Wikipedia. Prieiga per: <https://en.wikipedia.org/w/index.php?title=Average&oldid=1215581850>. [žiūrėta 2024-03-28].
26. Shaw Talebi. *The Fast Fourier Transform (FFT)*. Forumas internete. Prieiga per: <https://medium.com/swlh/the-fast-fourier-transform-fft-5e96cf637c38>. [žiūrėta 2024-03-28].
27. *The Sinc Function - MATLAB & Simulink - MathWorks Nordic*. Interaktyvus. Prieiga per: <https://se.mathworks.com/help/signal/gs/the-sinc-function.html>. [žiūrėta 2024-04-11].

28. Kheirati Roonizi Arman, Jutten Christian. *Forward-Backward Filtering and Penalized Least-Squares Optimization: A Unified Framework*. *Signal Processing*, 2021, vol. 178. pp. 31. ISSN 0165-1684. Prieiga per: <https://www.sciencedirect.com/science/article/pii/S0165168420303406>. [žiūrėta 2024-04-17].
29. Steven W. Smith. *Statistics, Probability and Noise*. In: *The Scientist and Engineer's Guide to Digital Signal Processing*. pp. 11-34. Interaktyvus. 1997, pp. 643. ISBN 0-9660176-6-8. Prieiga per: <https://www.dspguide.com/ch2.htm>. [žiūrėta 2024-04-24].
30. Steven W. Smith. *Moving Average Filters*. In: *The Scientist and Engineer's Guide to Digital Signal Processing*. pp. 277-284. Interaktyvus. 1997, pp. 643. ISBN 0-9660176-6-8. Prieiga per: <https://www.dspguide.com/ch15.htm>. [žiūrėta 2024-04-28].
31. Steven W. Smith. *Correlation*. In: *The Scientist and Engineer's Guide to Digital Signal Processing*. pp. 136-140. Interaktyvus. 1997, pp. 643. ISBN 0-9660176-6-8. Prieiga per: <https://www.dspguide.com/ch7/3.htm>. [žiūrėta 2024-04-30].
32. *interaural cross correlation*. Prieiga per: <https://asastandards.org/Terms/interaural-cross-correlation/>. [žiūrėta 2024-04-30].
33. Mathuranathan Viswanathan. *Power and Energy of a Signal: Demystified*. Interaktyvus. Prieiga per: <https://www.gaussianwaves.com/2013/12/power-and-energy-of-a-signal/>. [žiūrėta 2024-05-01].
34. *Envelope Analysis*. Internetinė svetainė. Prieiga per: <https://sensemore.io/envelope-analysis/>. [žiūrėta 2024-05-01].
35. H. Sun, M. Sun, J. Weng, Z. Liu. *Analysis of ID Sequences Similarity using DTW in Intrusion Detection for CAN Bus*. *IEEE Transactions on Vehicular Technology*, 2022, vol. 71, no. 10. pp. 16. ISSN 1939-9359. Prieiga per: <https://ieeexplore.ieee.org/document/9808189>. [žiūrėta 2024-05-05].

Priedai

1 priedas. Kros-koreliacijos metodo funkcija „Matlab“ programoje

```
function num_of_features = CC_metodas(input_signal, feature, feature_type,
sensor_type, sensor_axis)
% priskirti verte
if strcmp(feature_type, "sinc")
    if strcmp(sensor_type, "acc")
        if strcmp(sensor_axis, "x")

            elseif strcmp(sensor_axis, "y")
                minimum_corr_peak_val = 54.6981;
            elseif strcmp(sensor_axis, "z")

        else
            disp("netinkama ašis");
            return;
        end
    elseif strcmp(sensor_type, "gyro")
        if strcmp(sensor_axis, "x")
            minimum_corr_peak_val = 508420;
        elseif strcmp(sensor_axis, "y")

            elseif strcmp(sensor_axis, "z")
                minimum_corr_peak_val = 3965500;
        else
            disp("netinkama ašis");
            return;
        end
    else
        disp("netinkamas jutiklis");
        return;
    end
elseif strcmp(feature_type, "step")
    if strcmp(sensor_type, "acc")

        if strcmp(sensor_axis, "x")

            elseif strcmp(sensor_axis, "y")
                minimum_corr_peak_val = 29.5171;
            elseif strcmp(sensor_axis, "z")

        else
            disp("netinkama ašis");
            return;
        end
    elseif strcmp(sensor_type, "gyro")
        if strcmp(sensor_axis, "x")
            minimum_corr_peak_val = 336820;
        elseif strcmp(sensor_axis, "y")

            elseif strcmp(sensor_axis, "z")
                minimum_corr_peak_val = 3808700;
        else
            disp("netinkama ašis");
            return;
        end
    else
        disp("netinkamas jutiklis");
        return;
    end
end
```

```

elseif strcmp(feature_type, "suolis_nuo_bortelio")
    if strcmp(sensor_type, "acc")

        if strcmp(sensor_axis, "x")

            elseif strcmp(sensor_axis, "y")

                elseif strcmp(sensor_axis, "z")
                    minimum_corr_peak_val = 66.5791;
                    input_signal = input_signal - 1;
                    feature = feature - 1;
                else
                    disp("netinkama ašis");
                    return;
                end
            elseif strcmp(sensor_type, "gyro")
                if strcmp(sensor_axis, "x")

                    elseif strcmp(sensor_axis, "y")

                        elseif strcmp(sensor_axis, "z")

                            else
                                disp("netinkama ašis");
                                return;
                            end
                        else
                            disp("netinkamas jutiklis");
                            return;
                        end
                    else
                        disp("netinkamas požymio tipas");
                        return;
                    end
                end
            %kros-koreliacija
            cross_corr_results = xcorr(input_signal, feature); %kros-koreliacija tarp
            %pozymio ir turimo signalo
            feature_length = numel(feature);
            %rasti koreliacijos maksimumus
            manevrai = findpeaks(cross_corr_results, "MinPeakHeight",
            minimum_corr_peak_val, 'MinPeakDistance', feature_length);
            manevrai_inverted = findpeaks(-cross_corr_results, "MinPeakHeight",
            minimum_corr_peak_val, 'MinPeakDistance', feature_length);

            if (isempty(manevrai) && isempty(manevrai_inverted))
                disp("požymių nerasta")
                num_of_features = 0;
                return;
            else
                if (numel(manevrai) > numel(manevrai_inverted))
                    num_of_features = numel(manevrai);
                elseif (numel(manevrai) < numel(manevrai_inverted))
                    num_of_features = numel(manevrai_inverted);
                else
                    num_of_features = numel(manevrai);
                end
            end
            disp(sprintf('aptiktu manevru skaičius: %d', num_of_features));
        end
    end
end

```

2 priedas. DTW metodo funkcija „Matlab“ programoje

```
function num_of_features = dtw_metodas(input_signal, feature, feature_type,
sensor_type, sensor_axis)
% priskirti vertes
if strcmp(feature_type, "sinc")
    if strcmp(sensor_type, "acc")
        if strcmp(sensor_axis, "x")

            elseif strcmp(sensor_axis, "y")
                minimum_dtw_val = 63.110;
            elseif strcmp(sensor_axis, "z")

                else
                    disp("netinkama ašis ");
                    return;
                end
            elseif strcmp(sensor_type, "gyro")
                if strcmp(sensor_axis, "x")
                    minimum_dtw_val = 9782.751;
                elseif strcmp(sensor_axis, "y")

                    elseif strcmp(sensor_axis, "z")
                        minimum_dtw_val = 11192.417;
                    else
                        disp("netinkama ašis ");
                        return;
                    end
                else
                    disp("netinkamas jutiklis ");
                    return;
                end
            elseif strcmp(feature_type, "step")
                if strcmp(sensor_type, "acc")

                    if strcmp(sensor_axis, "x")

                        elseif strcmp(sensor_axis, "y")
                            minimum_dtw_val = 149.207;
                        elseif strcmp(sensor_axis, "z")

                            else
                                disp("netinkama ašis ");
                                return;
                            end
                        elseif strcmp(sensor_type, "gyro")
                            if strcmp(sensor_axis, "x")
                                minimum_dtw_val = 8734.647;
                            elseif strcmp(sensor_axis, "y")

                                elseif strcmp(sensor_axis, "z")
                                    minimum_dtw_val = 12116.296;
                                else
                                    disp("netinkama ašis ");
                                    return;
                                end
                            else
                                disp("netinkamas jutiklis ");
                                return;
                            end
                        elseif strcmp(feature_type, "suolis_nuo_bortelio")
                            if strcmp(sensor_type, "acc")
```

```

        if strcmp(sensor_axis, "x")

        elseif strcmp(sensor_axis, "y")

        elseif strcmp(sensor_axis, "z")
            minimum_dtw_val = 394.863;
        else
            disp("netinkama ašis ");
            return;
        end
    elseif strcmp(sensor_type, "gyro")
        if strcmp(sensor_axis, "x")

        elseif strcmp(sensor_axis, "y")

        elseif strcmp(sensor_axis, "z")

        else
            disp("netinkama ašis ");
            return;
        end
    else
        disp("netinkamas jutiklis ");
        return;
    end
else
    disp("netinkamas požymio tipas ");
    return;
end
%DTW skaičiavimas
result = dtw(input_signal, feature);

if (result > minimum_dtw_val)
    disp("požymių nerasta")
    num_of_features = 0;
    return;
elseif (result < minimum_dtw_val)
    num_of_features = 1;
    disp(sprintf('aptiktu manevru skaičius: %d', num_of_features));
end
end
end

```

3 priedas. Ribinés vertés metodo funkcija „*Matlab*“ programoje

```
function num_of_features = ribines_vertes_metodas(input_signal)
    signal_length = 0;
    signal_length = numel(input_signal);
    num_of_features = 0;
    i = 2;
    while (i < signal_length)
        if (input_signal(i - 1) > input_signal(i) && input_signal(i) < 0.4103)
            num_of_features = num_of_features + 1;
            % palaukti vidutine manevro trukme pries fiksuoiant kita manevra
            i = i + 80;
        end
        i = i + 1;
    end
end
```


4 priedas. Gaubtinės vidutinės vertės metodo funkcija „Matlab“ programoje

```
function num_of_features = gaubtines_metodas(input_signal, input_speed,
window_size, sample_space)
% gaubtines metodas
num_of_features = 0;
gaubtiniu_suma = 0;
start_index = 1;
kelio_dangos_gaubtine = envelope(input_signal, sample_space, 'peak');
number_of_calculations = floor(numel(kelio_dangos_gaubtine) / window_size);
for j = 1:number_of_calculations
    for k = 1:window_size
        gaubtiniu_suma = kelio_dangos_gaubtine(k + (window_size * (j - 1)))
+ gaubtiniu_suma;
    end
    vidutine_kelio_dangos_gaubtines_verte(j) = gaubtiniu_suma / window_size;
    gaubtiniu_suma = 0;
    if ((vidutine_kelio_dangos_gaubtines_verte (j) >= 3.526) &&
all(input_speed(start_index : j * window_size) >= 15))
        disp("vaziuojama prastu keliu")
        num_of_features = num_of_features + 1;
    else
        disp("vaziuojama per letai arba kelias nera prastas")
    end
    start_index = j * window_size + 1;
end
end
```

5 priedas. Vidutinės energijos metodo funkcija „Matlab“ programoje

```
function num_of_features = energijos_metodas(input_signal, input_speed,
window_size)
    % energijos metodas
    num_of_features = 0;
    start_index = 1;
    number_of_calculations = floor(numel(input_signal) / window_size)
    signalo_energija = zeros(1, number_of_calculations);
    for j = 1:number_of_calculations
        signalo_energija(j) = signal_energy(input_speed(start_index : j *
window_size));
        if ((signalo_energija(j) >= 2332) && all(input_speed(start_index : j *
window_size) >= 15))
            disp("vaziuojama prastu keliu")
            num_of_features = num_of_features + 1;
        else
            disp("vaziuojama per letai arba kelias nera prastas")
        end
        start_index = j * window_size + 1;
    end
end

function energy = signal_energy(signal)
sum_of_signals = 0;

    for i = 1:numel(signal)
        sum_of_signals = sum_of_signals + abs(signal(i)).^2;
    end
    energy = sum_of_signals;
end
```