

Tartu Ülikool  
Loodus- ja täppiseaduste valdkond  
Tehnoloogainstituut

Mattias Mäe

**Sõiduki kaugjuhtimise viivituse mõõtmine**

Bakalaureusetöö (12 EAP)

Arvutitehnika eriala

Juhendajad:

projektijuht Tambet Matiisen

robotika kaasprofessor Karl Kruusamäe

Tartu 2025

# Resümee/Abstract

## Sõiduki kaugjuhtimise viivituse mõõtmine

Kaugjuhtimine viitab olukorrale, kus sõidukit juhtiv operaator ei ole füüsiliselt sõiduki läheduses. Isejuhtivate sõidukite kontekstis kasutatakse seda olukordades, kus sõiduk ei ole võimeline tegema otsuseid iseseisvalt. Kaugjuhtimise ohutuse puhul on suurimaks riskiteguriks andmeedastuse viivitus. Käesoleva bakalaureusetöö eesmärgiks oli mõõta ja analüüsida viivitust, mis esineb Tartu Ülikooli isejuhtivate sõidukite labori testauto ja Clevoni kaugjuhtimiskonsooli vahelises suhtluses. Eelkõige keskenduti taju ja käsu viivituse mõõtmisele ja mõistmisele. Taju viivituse all mõeldakse ajakulu sündmuse toimumisest kaamera ees kuni selle kuvamiseni monitori ekraanil. Käsu viivitus tähendab juhtimiskäsu andmisest kuni selle füüsilise rakendamise lõpuni kuluvat aega, mõõtmisi sooritati pööramise ja pidurdamise andmetega. Töö käigus disainiti ja rakendati vastavad mõõteseadmete prototüübid, kasutades varasematel uurimustel põhinevaid, kuid kasutatud süsteemile kohandatud, meetodeid.

**CERCS:** P170 Arvutiteadus, arvutusmeetodid, süsteemid, juhtimine (automaatjuhtimisteooria), T121 Signaalitöötlus

**Märksõnad:** kaugjuhtimine, isejuhtivad sõidukid, viivitus, mõõtemetodid, signaalitöötlus

## Measuring Vehicle Teleoperation Latency

Teleoperation refers to a scenario where an operator has control over a vehicle from a distance. It is used in situations where a self-driving vehicle is unable to make decisions independently. A critical safety factor in teleoperation is data transmission latency. The aim of this bachelor's thesis was to measure and analyze the latency of communication between the self-driving vehicle used by the University of Tartu Autonomous Driving Lab and the Clevon teleoperation console. In particular, the focus was on understanding and quantifying both perception and command latency. Perception latency is defined as the time difference between an event happening in front of the camera and the event being displayed on the console display. Command latency refers to

the time difference between the generation of a control command and finishing its physical execution. Command measurements were collected from steering and braking data. Throughout this work, appropriate latency measurement prototypes were designed and implemented, drawing upon methods developed in prior research, but customized for the specific system.

**CERCS:** P170 Computer science, numerical analysis, systems, control, T121 Signal processing

**Keywords:** teleoperation, autonomous vehicles, latency, measurement methods, signal processing

# Sisukord

|   |           |
|---|-----------|
| <b>Resümee/Abstract.....</b>                | <b>2</b>  |
| <b>Sisukord.....</b>                        | <b>4</b>  |
| <b>Jooniste loetelu.....</b>                | <b>6</b>  |
| <b>Tabelite loetelu.....</b>                | <b>8</b>  |
| <b>Kasutatavad mõisted ja terminid.....</b> | <b>9</b>  |
| <b>Sissejuhatus.....</b>                    | <b>10</b> |
| <b>1. Ülevaade viivitusest.....</b>         | <b>12</b> |
| 1.1. Taju viivitus.....                     | 12        |
| 1.1.1. Varem kasutatud meetodid.....        | 12        |
| 1.2. Käsu viivitus.....                     | 13        |
| 1.2.1. Varem kasutatud meetodid.....        | 14        |
| <b>2. Meetodid.....</b>                     | <b>15</b> |
| 2.1. Taju viivituse mõõtmised.....          | 16        |
| 2.1.1. Kaamerapoolne süsteem.....           | 18        |
| 2.1.2. Operaatoripoolne süsteem.....        | 19        |
| 2.2. Käsu viivituse mõõtmised.....          | 21        |
| 2.3. Andmete analüüs.....                   | 23        |
| 2.4. Testimistingimused.....                | 26        |
| <b>3. Tulemused.....</b>                    | <b>27</b> |
| 3.1. Taju viivitus.....                     | 27        |
| 3.2. Käsu viivitus.....                     | 29        |
| 3.2.1. Piduramise viivitus.....             | 29        |
| 3.2.2. Pööramise viivitus.....              | 32        |
| <b>4. Järeldused.....</b>                   | <b>34</b> |
| 4.1. Taju viivitus.....                     | 34        |

|                                   |           |
|-----------------------------------|-----------|
| 4.2. Käsu viivitus.....           | 34        |
| <b>5. Arutelu.....</b>            | <b>36</b> |
| 5.1. Taju viivituse mõõtmine..... | 36        |
| 5.2. Käsu viivituse mõõtmine..... | 36        |
| <b>Kokkuvõte.....</b>             | <b>38</b> |
| <b>Tänuavaldused.....</b>         | <b>39</b> |
| <b>Viidatud kirjandus.....</b>    | <b>40</b> |
| <b>Lisad.....</b>                 | <b>43</b> |
| <b>Lihtlitsents.....</b>          | <b>44</b> |

# Jooniste loetelu

|  |    |
|--|----|
| Joonis 1. Tartu Ülikooli isejuhtivate sõidukite labori kasutuses olev Lexus RX450h [14] ja Clevoni juhtimispunkt [16].....                       | 15 |
| Joonis 2. G2G andmevahetuse etapid; Halli taustaga kujutatakse sõiduki arvuti, sinisega juhtimispunkti, valgega kaamera ja kuvari protsesse..... | 16 |
| Joonis 3. Taju mõõtmise protsessid;  |    |
| 1. Sõiduki ja juhtpunkti Raspberry Pi seadmed sünkroniseeritakse NTP serveriga enne mõõtmise alustamist;   |    |
| 2. Sõidukipoolne Raspberry vahetab LED-i olekut ja salvestab ajatempli;  |    |
| 3. Kaamera andurid tuvastavad LED-i muutust;   |    |
| 4. Kaamera edastab andmed sõiduki arvutile;  |    |
| 5. Sõiduki arvuti kodeerib andmed ja edastab üle võrgu juhtimispunkti arvutile;  |    |
| 6. Juhtimispunktis andmed dekodeeritakse, puhverdatakse ja renderdatakse;  |    |
| 7. Renderdatud andmed saadetakse monitorile kuvamiseks;  |    |
| 8. Valguse sensor loeb uue oleku ja Raspberry salvestab ajatempli; tsükkel jätkub 2. punkti juurest.....   | 17 |
| Joonis 4. Sõiduki poolse süsteemi ühendamine (a); LED paigutati koos 3D-prinditud kupliga kaamera ette ja kinnitati isoleerteibiga (b).....      | 18 |
| Joonis 5. Juhtimispunktipoolne elektriskeem (b) ja süsteem kinnitatuna telefoni statiivi külge (b).  | 19 |
| Joonis 6. Süsteemi asetus juhtkonsooli kuvari ees.....   | 20 |
| Joonis 8. Täituri andmevahetuse etapid; Sinise taustaga kujutatakse juhtpunkti, halliga sõiduki arvuti protsesse.....                            | 21 |
| Joonis 9. G2G andmete töötlemise ja analüüsimise etapid.....   | 23 |
| Joonis 10. Täituri andmete töötlemise ja analüüsimise etapid.....  | 24 |
| Joonis 11. Teekonnad Delta õppehoonest õppesõidu platsile (a) ja tagasi sõites (b).....  | 26 |
| Joonis 12. Histogramm taju viivitustest, liikudes õppesõiduplatsilt Delta suunas.....  | 27 |
| Joonis 13. Tõusva ja langeva signaali histogrammid, liikudes õppesõiduplatsilt Delta suunas....  | 28 |

|   |    |
|---|----|
| Joonis 14. Joongraafik pidurdamise andmetest. Sisendi ja edastatud käsu vahel mõõdetakse võrgu viivitust, käsu ja väljundi vahel täituri viivitust..... | 30 |
| Joonis 15. Joongraafik pööramise andmetest. Sisendi ja edastatud käsu vahel mõõdetakse võrgu viivitust, käsu ja väljundi vahel täituri viivitust.....   | 32 |

# Tabelite loetelu

|  |    |
|--|----|
| Tabel 1. Taju viivituse mõõtmiste tulemused.....   | 28 |
| Tabel 2. Pidurdamise viivituse tulemused 50 Hz ja 125 Hz interpoleerimissagedustega..... | 31 |
| Tabel 3. Pööramise viivituse tulemused.....  | 33 |

## **Kasutatavad mõisted ja terminid**

**G2G** - *glass-to-glass latency*, klaas-klaas viivitus

**LED** - *light emitting diode*, valgusdiod

**NTP** - *network time protocol*, arvutitevaheline kellaaja sünkroniseerimise protokoll

**HD** - *high definition*, kõrglahutusega videopilt

**XCORR** - *cross-correlation*, ristkorrelatsioon

**FFT** - *fast Fourier' transform*, kiire Fourier' teisendus

**DTW** - *dynamic time warping*, dünaamiline ajadeformatsioon

# Sissejuhatus

Tehisintellekti ja autonoomsete süsteemide areng on viimase aastakümnega teinud tohutu arenguhüppe, tuues meid lähemale tulevikule, kus isejuhtivad sõidukid muudavad liikluse tõhusamaks ja ohutumaks [1]. Siiski ei ole täielikult autonoomsed sõidukid veel laialdaselt kasutusele võetud, kuna nende töökindlust ja turvalisust ei ole piisavalt järjepidevalt tõestatud [2, 3]. Üks peamisi takistusi on vajadus järelevalve ja sekkumise järele olukordades, kus autonoomne süsteem pole võimeline iseseisvalt otsuse tegemiseks. Sellistes olukordades kasutatakse kaugjuhtimist, mille puhul operaator juhib sõidukit eemalt [4].

Kaugjuhtimise üks kriitilisemaid aspekte on viivitus, ehk operaatori ja sõiduki vahel andmete edastamisele kuluv aeg. Kui viivitus on suur, väheneb juhtimistäpsus, põhjustades hilinenud pööramist või pidurdamist. Kuigi 5G-ühendus on viivituse vähendamisele kaasa aidanud, ei ole sellest võimalik täielikult vabaneda, kuna see sõltub mitmest tegurist, nagu võrgutingimused, riistvara töötluskiirus ja kaugjuhtimissüsteemi ülesehitus. [5]

Käesoleva bakalaureusetöö eesmärk on mõõta ja kaardistada viivitust mõjutavaid tegureid Tartu Ülikooli isejuhtivate sõidukite labori kasutuses oleva testauto ja Clevoni juhtkonsooli vahel. Selleks analüüsitakse varasemalt kasutatud mõõtemetodeid, et leida sobivaimad lahendused kasutatud süsteemi kontekstis, ning disainitakse taju ja täituri viivituse mõõteseadmete prototüübid. Seejärel sooritatakse reaalsed mõõtmised sõiduki ja juhtkonsooli vahel. Saadud tulemused visualiseeritakse ning analüüsitakse, et tuvastada suurimad viivituse allikad.

Töö annab ülevaate, millised on süsteemide viivitused, kui suures ulatuses need mõjutavad sõiduki juhtimist ning millised tegurid on kõige kriitilisemad kaugjuhtimise usaldatavuse tagamiseks. Taju mõõtmiste puhul leiti 428 andmepunkti keskmiseks viivituseks 149 ms, standardhällbega 78 ms ning miinimum ja maksimum väärtustega 44 ms ja 1355 ms. Täituritest sooritati pööramise ja pidurdamise viivituse mõõtmised. Pööramise puhul leiti akendatud dünaamilist ajadeformatsiooni (DTW) kasutades keskmiseks viivituseks 294 ms, pidurdamise puhul leiti kogu andmestiku ristkorrelatsiooni (XCORR) kasutades keskmiseks viivituseks 96 ms.

Kui liita kokku viivitused (taju keskmine ja pööramise keskmine) saame otspunktide vaheliseks viivituseks umbes 450 ms, mis on liiga suur selleks, et operaator suudaks turvaliselt linnaliikluses manöövreid sooritada. Lisades inimese ligikaudse reaktsiooniaja 300 ms [6], siis saaksime kokku viivituseks 750 ms. Oletades, et sõidukile satub ette takistus ja arvestades sõidukiiruseks 50 km/h, jõuaks sõiduk läbida üle 10 m enne seda, kui täitur vajalikul määral rakenduks.

Töö jaguneb viieks peatükiks. Esmalt antakse ülevaade viivituse olemusest isejuhtivate sõidukite kontekstis, keskendudes taju ja käsu viivitusele ning varasematele mõõtemetoditele. Seejärel kirjeldatakse kasutatud metoodikat, sealhulgas mõõteseadmete disaini, andmesalvestust ja -analüüsi. Tulemuste peatükis esitatakse reaalselt läbi viidud mõõtmiste tulemused ning nende tähendus. Järgnev peatükk koondab peamised järeldused, nagu videoedastuse ebastabiilsus ja täituri reageerimise piirangud. Lõpuks arutletakse töö kitsaskohtade ja edasiste arenguvõimaluste üle.

# 1. Ülevaade viivitusest

Viivitus või latentsus on viiteaeg, mis esineb andmete edastamisel süsteemi erinevate osade või süsteemide vahel. Kaugjuhitava sõiduki puhul tähendab näiteks taju viivitus ajavahet anduri poolt tajutud sündmuse ja operaatori reaktsiooni vahel. Liiga kõrge viivitus raskendab juhtimist ja vähendab juhtimise täpsust, põhjustades üle- ja alapööramist või hilinevad pidurdamist. Kui viiteaeg ületab 300 ms, muutub sujuv juhtimine praktiliselt võimatuks, sundides operaatorit kasutama "liigu ja oota" taktikat [7]. See tähendab, et operaator annab sõidukile käskluse, peatab selle ning ootab kuvaril käskluse tulemise kajastumist, enne järgmise käskluse andmist. Selliselt ei ole võimalik sõita tavapärasel liikluses, kus sõiduk peab manööverdama sujuvalt ja reageerima kiiresti ohtlikes olukordades [4].

Kaugjuhtimise puhul saab viivitust jagada kolmeks suuremaks osaks:

1. taju viivitus - sõiduki sensorist operaatori silmani,
2. operaatori reaktsiooniaeg - sündmuse nägemisest füüsilise reageerimiseni,
3. käsu viivitus - operaatori juhtkonsoolist sõiduki täiturini.

Käesolevas töös ei koostatud eraldi lahendust reaktsiooniaja mõõtmiseks ning lähtuti inimese keskmisest reaktsiooniajast [6].

## 1.1. Taju viivitus

*Glass-to-glass* viivitus (G2G) viitab ajakulule andmete edastamise ja ekraanil kuvamise protsessis, alates kaamera sensoril registreeritud sündmusest kuni sündmuse kuvamiseni monitori ekraanil. Viivitust põhjustavad peamiselt kaamera kaadrisagedus, video kodeerimine, andmeedastus üle võrgu, video dekodeerimine ja kuvaril renderdamine. [8]

### 1.1.1. Varem kasutatud meetodid

G2G viivituse mõõtmiseks on kasutatud mitmeid meetodeid, kuid paljud neist on kas ebatäpsed või liiga palju manuaalset tööd nõudvad:

- Kaamera ees käe viipamine - mõõdetav kaamera on suunatud inimese poole, mõõtmiseks kasutatav kaamera on suunatud esimese kaamera pilti nätava kuvari ja sama inimese poole. Kui ta oma kätt liigutab, siis on näha, kuidas kuvaril liikumine realiseerub viivitusega. Teise kaamera salvestist on võimalik käsitsi analüüsida, et leida, kui kaua aega kulub, et käsi samasugusesse asendisse jõuaks. See on aga ebapraktiliselt aeganõudev ja tülikas. Samuti tekitaks teise kaamera kaasamine ning käsitsi analüüsimine ebatäpsust. [9]
- Kella pildistamine monitoriga kõrvuti - kuvari kõrvale asetatakse jooksva stopperiga telefon või arvuti. Mõõdetav kaamera suunatakse stopperi peale ning mõõtmiseks kasutatav kaamera suunatakse nii, et stopper ja kuvar oleksid kaadris. Kui teise kaamera kaadrit vaadata, siis võib olla võimalik välja lugeda diskreetne viivitus, leides kummagi jäädvustatud stopperi aegade vahe. Selle meetodi puhul esineb täpsus stopperi täpsusest (tänapäeval telefoni stopper tihti 10 ms täpsusega). Lisaks raskendab täpsemat mõõtmist teise kaamera säriaeg, kuna kaadritel võivad numbrid olla udused ja varasemate väärtustega ülekattuvad. [8, 9]
- Intervalliga vilkuv LED ja valguse andur - juhitava seadme külge (näiteks Raspberry Pi) on ühendatud LED ja valguse andur (fotodiod või -takisti). LED pannakse mõõdetava kaamera ette ning andur kaamerapilti kajastava kuvari vastu. Mõlemad komponendid on ühendatud ostsilloskoobi erinevate kanalite külge. Kui kaamera ees oleva LED-i olek muutub, siis saaks ostsilloskoobi kuvarilt näha valguse sensori väljundi viivitust. Olenevalt ostsilloskoobist, oleks võimalik andmeid salvestada ja programmiselt analüüsida. Alternatiivselt saab ostsilloskoobi asemel edastada andmed otse arvutisse, kus neid kuvada ja analüüsida. [8–10]

## 1.2. Käsu viivitus

Käsu viivitus, ehk mehhanismi reageerimisviivitus, viitab juhtimiskäsu loomisest kuni selle füüsilise rakendamiseni kuluvale ajale. See hõlmab näiteks rooli keeramisel rataste pööramist või pedaalivajutuse peale pidurdamist. Kuna täituri puhul pole enam tegu ainult andmete edastamisega, tuleb viivituse puhul arvestada mehaaniliste teguritega, nagu rehvi ja maapinna vaheline hõõrdumine, mis võib viivitust pikendada. [11]

### 1.2.1. Varem kasutatud meetodid

Käsu viivituse mõõtmiseks on kasutatud näiteks lineaarse liikumise sensoreid, mis on kinnitatud juhtkonsooli ja sõiduki rooli külge rakisega. See lahendus võimaldab täpselt mõõta rooli asendi muutust, kuid nõuab eraldi mehhanismi disaini ja paigaldamist. Alternatiivselt oleks võimalik mõlemas arvutis välja lugeda rooli enkooderi väljund. Selleks, et vähendada hõõrdumise mõju viivitusele, saaks panna esirataste alla vabalt pöörlevad mehhanismid või kergitada sõiduki esiotsa, tõstes esirattad maast. [11]

## 2. Meetodid

Selles peatükis kirjeldatakse lähemalt, kuidas Tartu Ülikooli isejuhtivate sõidukite labori Lexus RX450h (joonis 1(a)) [12–14] ja Clevoni juhtimispunkti (joonis 1(b)) [15, 16] vahel viivituste kaardistamist sooritati. Alustuseks antakse ülevaade kasutatavatest mõõtemetoditest, mis põhinevad varasematel lahendustel ning on kohandatud mõõdetava süsteemi jaoks. Seejärel käsitletakse taju ja käsu viivituse mõõteseadmete prototüüpide loomist, mõõtmiste läbiviimist ning andmete analüüsimist ja olulise statistika leidmist. Meetodite kirjeldus annab aluse töö tulemustele, järeldustele ja laseb hinnata kasutatud lahenduste täpsust.



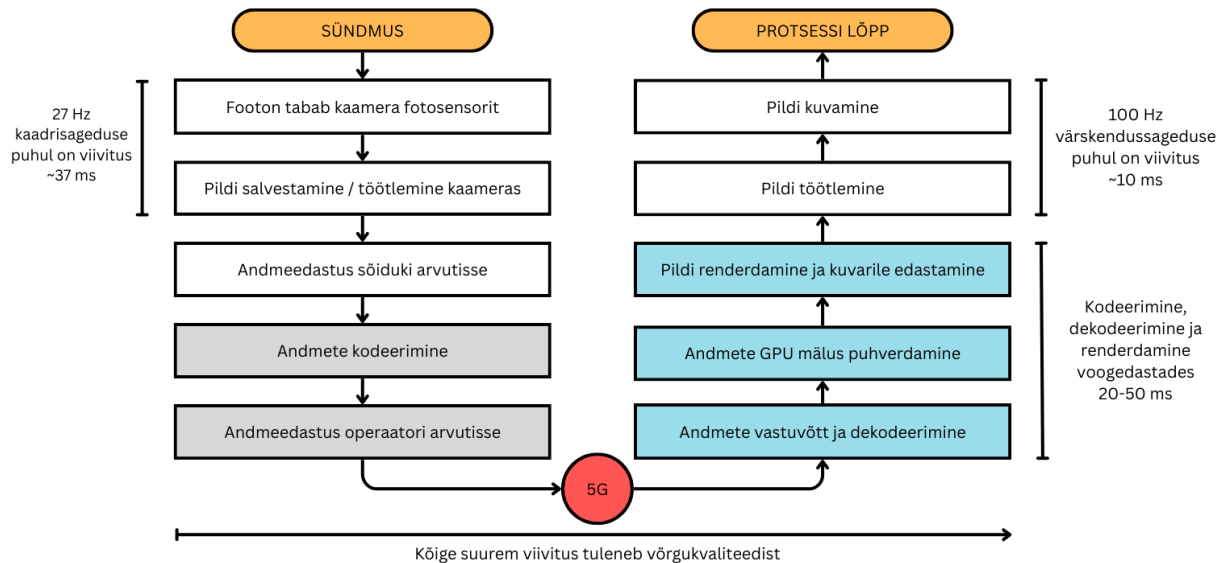
(a)



(b)

*Joonis 1. Tartu Ülikooli isejuhtivate sõidukite labori kasutuses olev Lexus RX450h [14] ja Clevoni juhtimispunkt [16].*

## 2.1. Taju viivituse mõõtmised

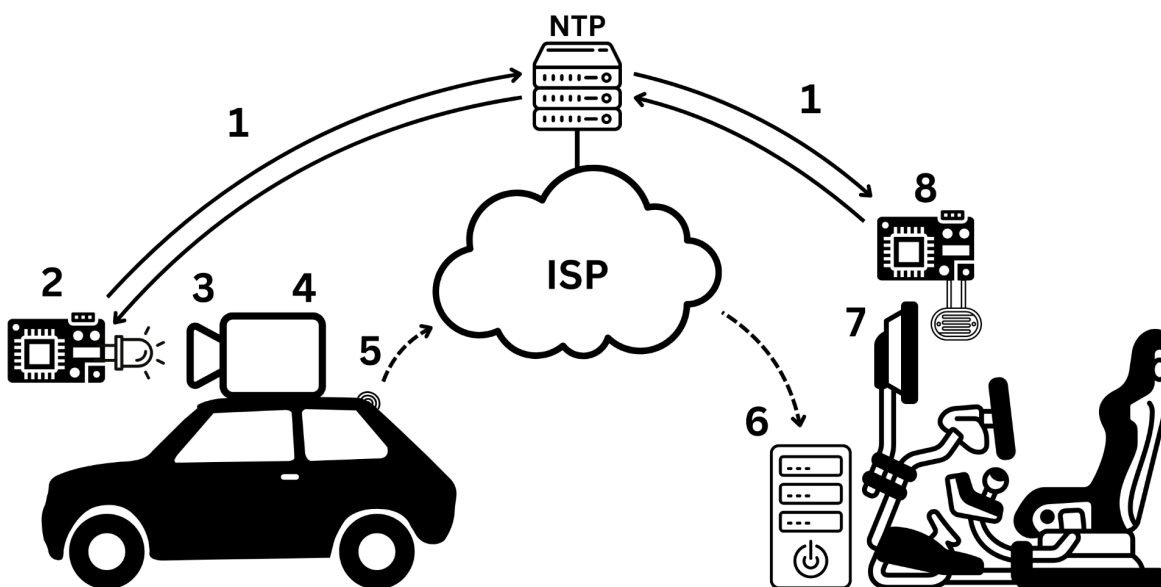


Joonis 2. G2G andmevahetuse etapid; Halli taustaga kujutatakse sõiduki arvuti, sinisega juhtimispunkti, valgega kaamera ja kuvari protsesse.

Joonisel 2 on kujutatud plokk skeem taju andmevahetuse etappidest. Kaamera ja kuvari põhjustatud viivitust saab ligikaudset hinnata nende kaadri- ja värskendussageduse põhjal. Mõõtmisteks kasutati sõiduki Sekonix SF3324 kaamerat, mille kaadrisagedus on ligikaudu 27 Hz, ehk ühe kaadri jäädvustamiseks kulub keskmiselt 37 ms [12, 17]. Juhtimispunkt kasutab Samsung ViewFinity CJ79 kuvarit, mille värskendussagedus on 100 Hz, ehk kaadri värskendamiseks kulub umbes 10 ms [18]. Andmete kodeerimisele, dekodeerimisele ja renderdamisele kuluv aeg oleneb kaadri suurusest, kasutatud failitüübist ja pakkimismeetodist [19, 20]. Üldpildis toimuvad need protsessid suhteliselt kiiresti ning ei põhjusta märkimisväärset viivitust (HD voogedastuse juures tänapäeval <50 ms) [20, 21]. Kõige olulisem tegur on arvutitevaheline andmeedastus üle mobiilse võrguside (4G/5G), mis sõltub levi mõjutavast keskkonnast ning kasutatud riistvara robustsusest, näiteks on kahe SIM kaardiga modem töökindlam levialade vahel liikudes.

G2G mõõtmiste sooritamiseks kasutati [8] meetodile sarnast disaini. Kuna operaatori juhtimispunkt asus Delta hoone 3. korrusel ja sõidukit juhiti Tartu linna tänavatel, siis ei oleks võimalik ühe süsteemiga LED-i ja sensorit juhtida. Selle probleemi lahendamiseks kasutati kahte

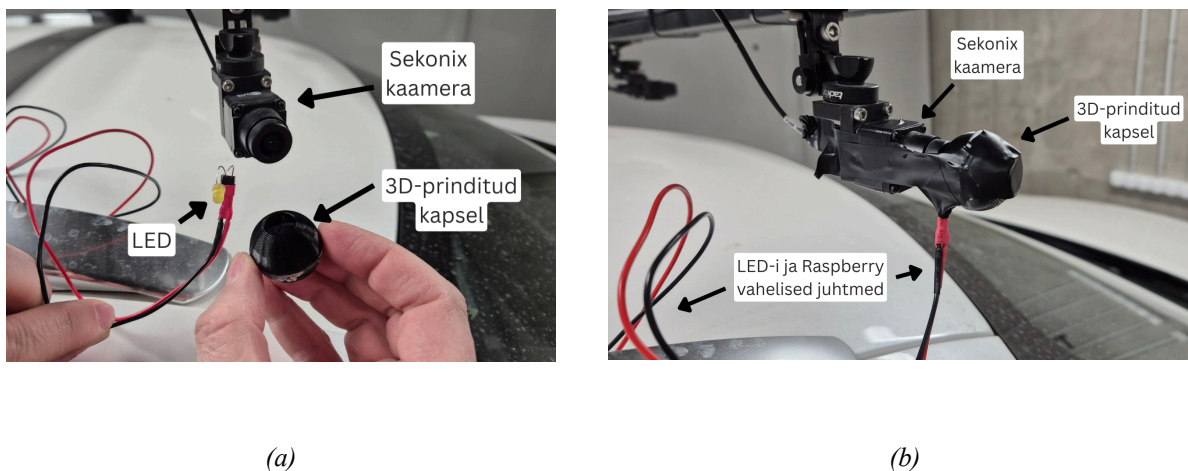
eraldi arvutit, mille süsteemi kellad sünkroniseeriti kasutades NTP protokollit [12]. Lisaks kasutati KY-018 fototakistimoodulit [13], mis on fototransistori või -diodiga võrreldes palju aeglasem, kuid piisavalt täpne käesoleva töö eesmärkide täitmiseks [22, 23]. Mõõtmise protsessi selgitav skeem on kujutatud joonisel 3.



*Joonis 3. Taju mõõtmise protsessid;*

1. Sõiduki ja juhtpunkti Raspberry Pi seadmed sünkroniseeritakse NTP serveriga enne mõõtmise alustamist;
2. Sõidukipoolne Raspberry vahetab LED-i olekut ja salvestab ajatempli;
3. Kaamera andurid tuvastavad LED-i muutust;
4. Kaamera edastab andmed sõiduki arvutile;
5. Sõiduki arvuti kodeerib andmed ja edastab üle võrgu juhtimispunkti arvutile;
6. Juhtimispunktis andmed dekodeeritakse, puhverdatakse ja renderdatakse;
7. Renderdatud andmed saadetakse monitorile kuvamiseks;
8. Valguse sensor loeb uue oleku ja Raspberry salvestab ajatempli; tsükkel jätkub 2. punkti juurest.

### 2.1.1. Kaamerapoolne süsteem



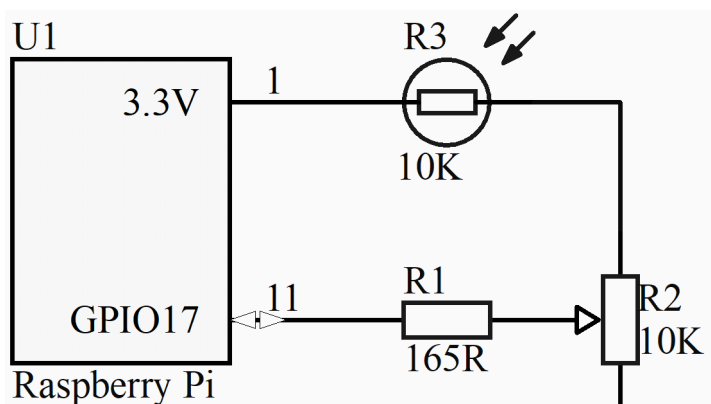
Joonis 4. Sõiduki poolse süsteemi ühendamine (a); LED paigutati koos 3D-prinditud kupliga kaamera ette ja kinnitati isoleerteibiga (b).

Süsteemi juhtis Raspberry Pi 4 [24], mille GPIO pordi külge on jadamisi ühendatud  $330 \Omega$  takisti ja kollane LED. Selleks, et kaitsta Raspberry Pi väliskeskkonna eest, kasutati ligikaudu 2 m pikkuseid kaableid LED-i ühendamiseks. Raspberry Pi asetses sõiduki kabiinis ning juhe suunati sõiduki uksele vahelt kaamera ette. Lisaks 3D-prinditi kaks 35 mm läbimõõduga musta värvi poolkera (joonis 4(a)) [25], et kaitsta valgusreostuse vastu. PrusaSlicer tarkvaras [26] loodi ühele poolkerale 10 mm ava juhtmete jaoks. Pooled liimiti kokku ja LED asetati kuplisse nii, et see oleks suunatud kaamera poole. Joonisel 4 on kujutatud sõiduki küljekaamerat, Raspberry Pi külge ühendatud LED-i ja kuplit. Kupli ja juhtmete kaamera külge kinnitamiseks kasutati isoleerteipi (joonis 4(b)).

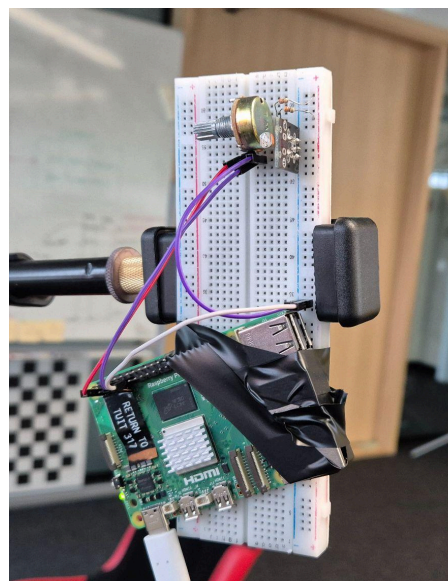
Mõõtmiste läbiviimiseks kirjutati Pythoni programm, mis seadistas LED-i juhtiva pordi, lülitas tsüklis vaheldumisi selle olekut ning salvestas ajatempli koos olekuga CSV faili. LED-i olek muutus iga 1 sekundi tagant. Mõõtmistsükli järjepidevuse kontrollimiseks pikendati iga 5. süttimise kestust 2 sekundini, et andmete hilisemal analüüsimisel oleks tuvastada võimalike vigu ja kontrollida andmestiku usaldusväärsust.

## 2.1.2. Operaatoripoolne süsteem

Operaatori poolset süsteemi juhtis Raspberry Pi 5 [27], mille GPIO pordiga ühendati jadamisi KY-018 fototakisti moodul [28], 10 k $\Omega$  potentsiomeeter [29] ja 165  $\Omega$  takisti (joonis 5). Kuna kasutatud Raspberry seadmel puudusid analoogsisendiga pordid [30, 31], ühendati fototakisti digitaalse GPIO pordi külge. Selleks, et määrata valgustundlikuse lävendit, ühendati pingejaguri loomiseks mooduliga jadamisi potentsiomeeter. Tänu potentsiomeetrile oli võimalik valida sobiv takistus, et Raspberry Pi port eristaks pimedat ja eredat signaali [32]. Takisti oli lisatud voolu piiramiseks.



(a)

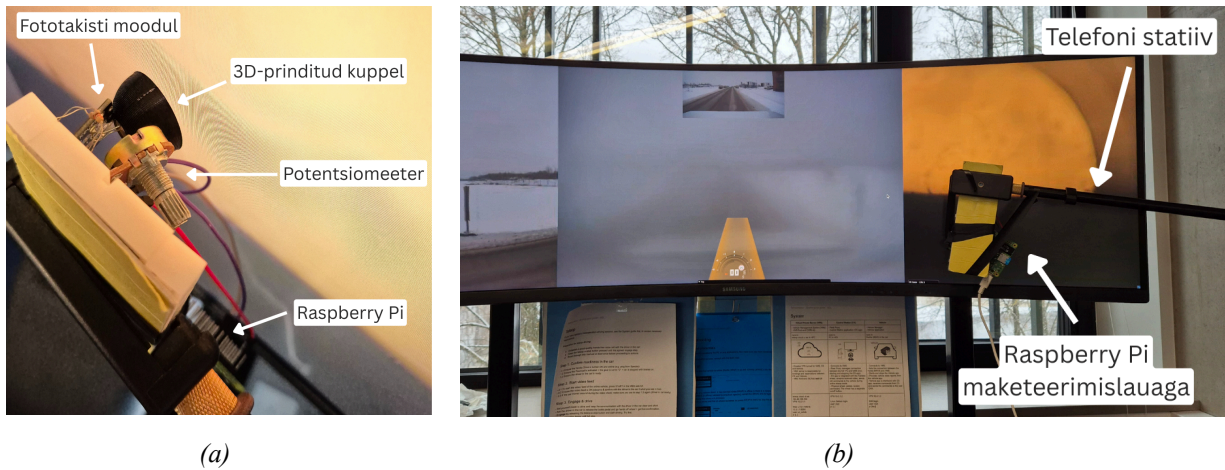


(b)

Joonis 5. Juhtimispunktipoolne elektriskeem (a) ja süsteem kinnitatuna telefoni statiivi külge (b).

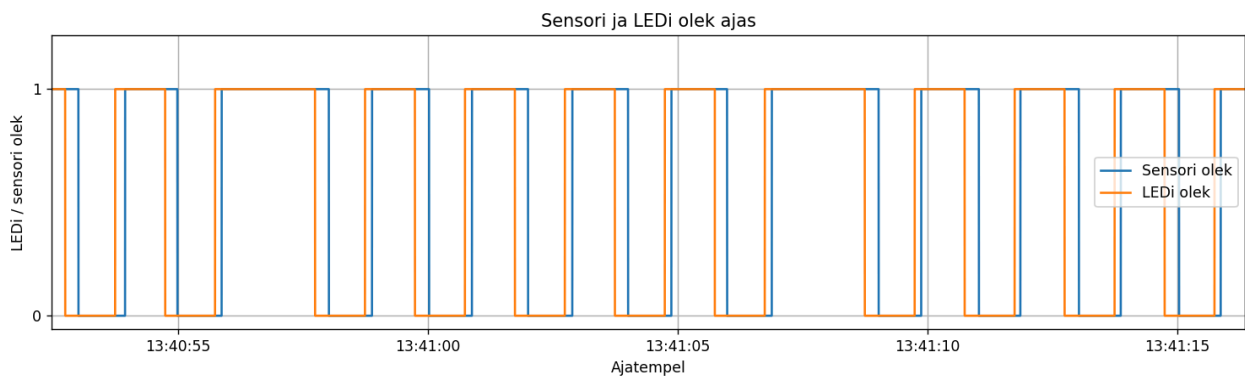
Fototakisti pidi olema võimalikult lähedal kuvarile, et vähendada taustavalguse mõju. Joonisel 5(b) on kujutatud statiivi külge kinnitatud maketeerimislauda koos komponentide ja Raspberry Pi-ga. Statiivi abil oli võimalik süsteemi täpselt suunata LED-i asukohale kaamerapildis, nagu on kujutatud joonisel 6. Kuna juhtimispunkti polnud võimalik täielikult valguse eest isoleerida,

valmistati fototakisti katmiseks 3D-prinditud must poolkera läbimõõduga 20 mm [25], mis aitas valgusreostuse vastu. PrusaSlicer tarkvaras loodi poolkerale umbes 5 mm ava fototakisti jaoks.



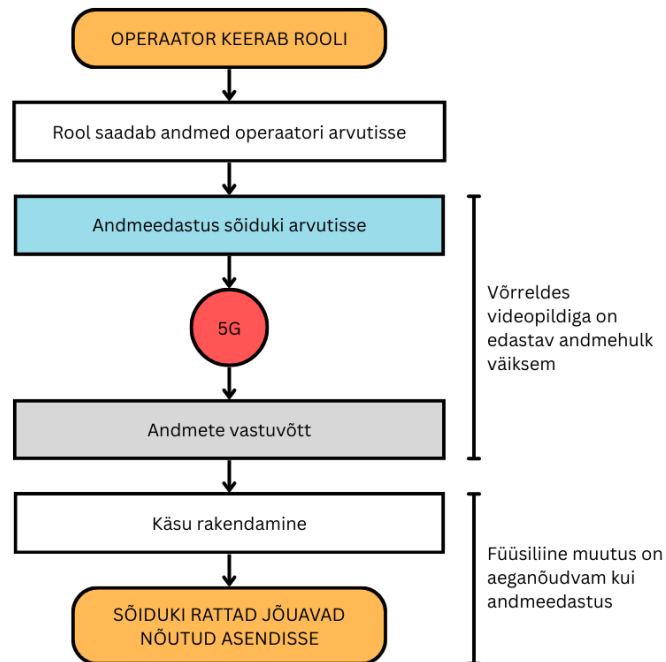
Joonis 6. Süsteemi asetus juhtkonsooli kuvari ees.

Mõõtmiste läbiviimiseks kirjutati Pythoni programm, mis salvestas pidevalt fototakisti loetud oleku koos ajatempliga CSV faili (andmed visualiseeritud joonisel 7). Salvestamise periood oli ligikaudu 2 ms, mis tulenes peamiselt faili kirjutamise kiirusest.



Joonis 7. Taju mõõtmiste käigus salvestatud andmed.

## 2.2. Käsu viivituse mõõtmised



Joonis 8. Täituri andmevahetuse etapid; Sinise taustaga kujutatakse juhtpunkti, halliga sõiduki arvuti protsesse.

Joonisel 8 on kujutatud plokk skeem täituri juhtimise andmevahetuse etappidest. Operaatori juhtkonsooli sündmused logitakse ligikaudu 133 Hz sagedusel, ehk andmepunktide vaheline viivitus on 7 ms ringis. Sõidukis salvestatakse andmeid 30 Hz sagedusel. Sarnaselt tajule on käskude puhul üks viivituse allikas andmeedastus üle mobiilse võrgu, mille kvaliteet sõltub keskkonnast ja seadme vastuvõttu kvaliteedist. Erinevalt G2G mõõtmistest on edastatava käsu andmemaht kordades väiksem võrreldes videopildi mahuga. Samas tuleb täiturite puhul arvestada ka füüsilise liikumise ajakuluga, näiteks rataste pööramine ei ole kohene ning sõltub sõiduki mehaanilistest omadustest kui ka pinnakatte hõõrdetegurist. Lisaks oli ohutuskaalutlustel kaugjuhtimisele määratud pööramiskiiruse piirang, et sõidukis viibiv julgestaja jõuaks reageerida juhtimise üle võtmisel.

Töö käigus kaaluti erinevate sensorite, näiteks kvadratuurkooderite ja Halli andurite kasutamist, et mõõta rooli või esiratta nurka. Need meetodid oleks nõudnud kalibreerimist ning ei olnud

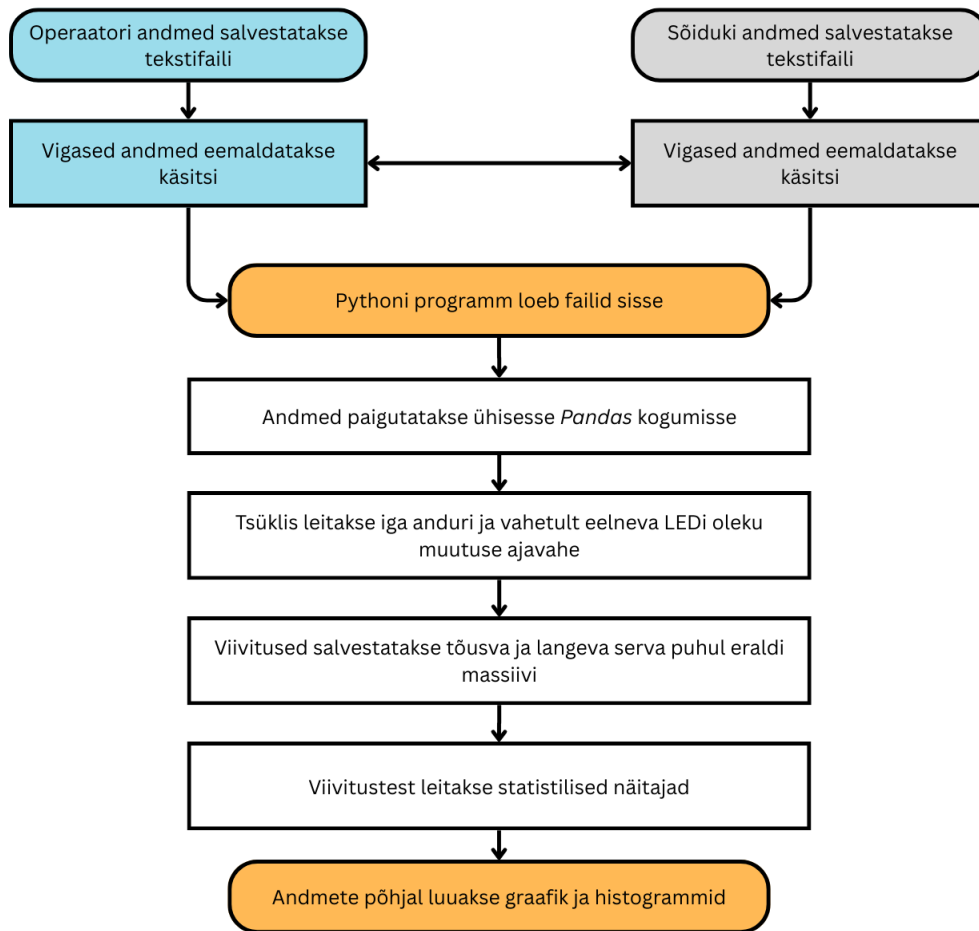
sobilikud rakendamiseks sõidu ajal, kuna oleks nõudnud täiendava riistvara paigaldust. Seetõttu kasutati tarkvarapõhist lahendust.

Juhtimispunkti arvutis salvestatakse rooli ja pedaalide sündmused koos ajatemplitega otse tekstifaili. Selleks kasutati Linuxi terminalis käsklust “`evtest /dev/input/event4 > failinimi.txt`”, mis suunas sündmusi logiva käsu väljundi otse tekstifaili [33]. Pythonis kirjutati andmetöötlusfunktsioon, mis luges salvestatud tekstifaili, filtreeris välja kõik peale piduramise ja pööramise sündmuste ning tagastas rooli ja piduri pedaali asendid koos ajatemplitega.

Sõidukis käivitati ROS-i keskkonnas käsk “`rosvag record -a`”, mis salvestas kõik andmevood, sealhulgas “`/can_tx`” rubriigist CAN-liidese sõnumid koos ajatemplitega. Olulised olid sõnumid identifikaatoritega 556 ja 516, milles kajastuvad rataste ja pidurite puhul operaatori käsud (*commanded\_value*) ja täituri tegelik asend (*output\_value*) [34]. Andmete töötlemiseks loodi Pythoni programm, mis avas *bag*-faili, tuvastas vajalikud sõnumid, dekodeeris vajalikud väärtused vastavalt CAN-andmebaasile [34] ning salvestas andmed ja ajatemplid järjenditesse.

## 2.3. Andmete analüüs

Andmete analüüs toimus kahes eraldiseisvas osas, sõltuvalt sellest, kas tegemist oli G2G või käsu mõõtmistega. Mõlemal juhul kasutati andmete töötlemiseks ja visualiseerimiseks Pythonit koos vastavate andmetöötlushpakettidega, nagu NumPy, SciPy ja Pandas.

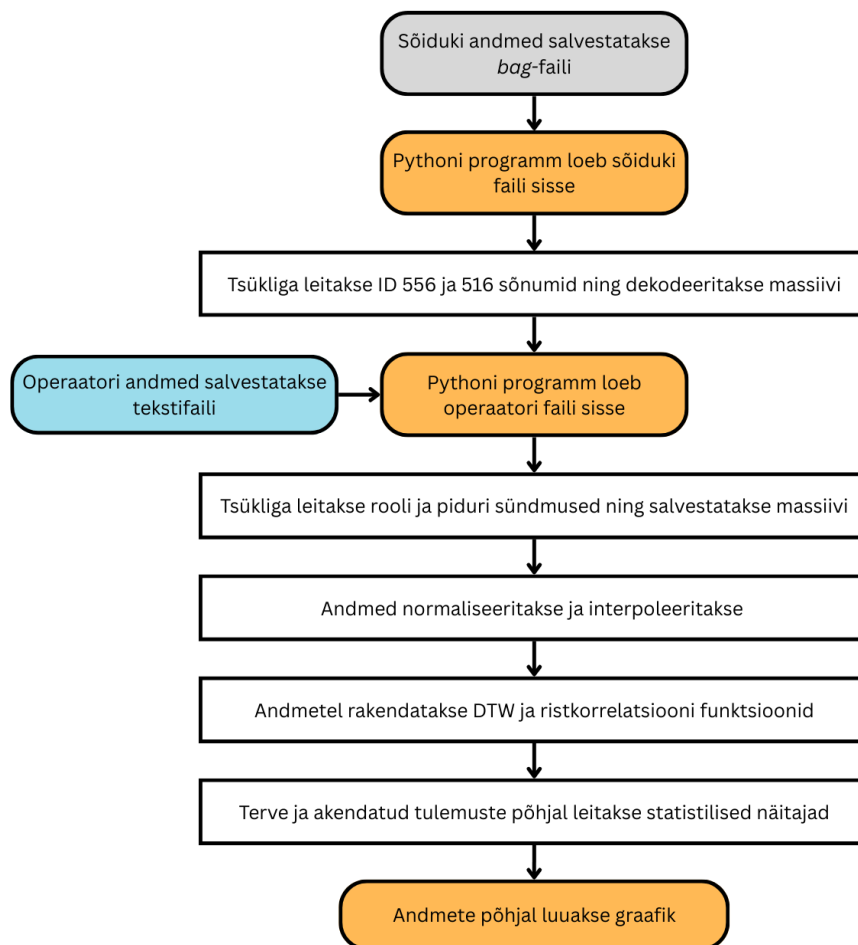


Joonis 9. G2G andmete töötlemise ja analüüsimise etapid.

Joonisel 9 on kujutatud plokk skeem taju andmetöötlemise etappidest. G2G mõõtmiste puhul tehti esmalt andmetele eeltöötlus, mille käigus eemaldati moonutatud ja ebakorrektsed mõõtmised. Allesjäänud tekstifailid loeti programmi ning neist moodustati struktureeritud andmekogumid sündmuste ja ajatemplitega kasutades Pandas paketti.

Viivituste leidmiseks läbiti andmekogumit tsükliliselt tagurpidi. Iga tuvastatud fototakisti oleku muutuse korral leiti sellele lähim eelnev samasuunaline LED-i oleku muutus. Nende kahe sündmuse ajavahe arvutati viivitusena. Kõik leitud viivitused salvestati massiivi, mille põhjal arvutati statistilised näitajad: minimaalne, maksimaalne ja keskmine viivitus ning standardhälve.

Analüüsi tulemused visualiseeriti joongraafiku ja histogrammidena. Histogrammid koostati kogu viivitusandmestiku kohta ning eraldi tõusvate ja langevate servade jaoks, kuna analüüsi käigus tuvastati nende vahel märkimisväärsed erinevused.



Joonis 10. Täituri andmete töötlemise ja analüüsimise etapid.

Joonisel 10 on kujutatud plokkskeem täituri andmetöötlemise etappidest. Käskude andmete puhul kasutati kahte erinevat salvestusmeetodit: sõiduki *bag*-failid, millest eraldati täiturite asendid, ning operaatori tekitatud juhtsignaalid, mille logid salvestati tekstifaili. Mõlemad andmekogumid

loeti programmi, normaliseeriti ja interpoleeriti selleks, et saavutada ühtlane andmesagedus, mis võimaldab teha võrdlusanalüüsi.

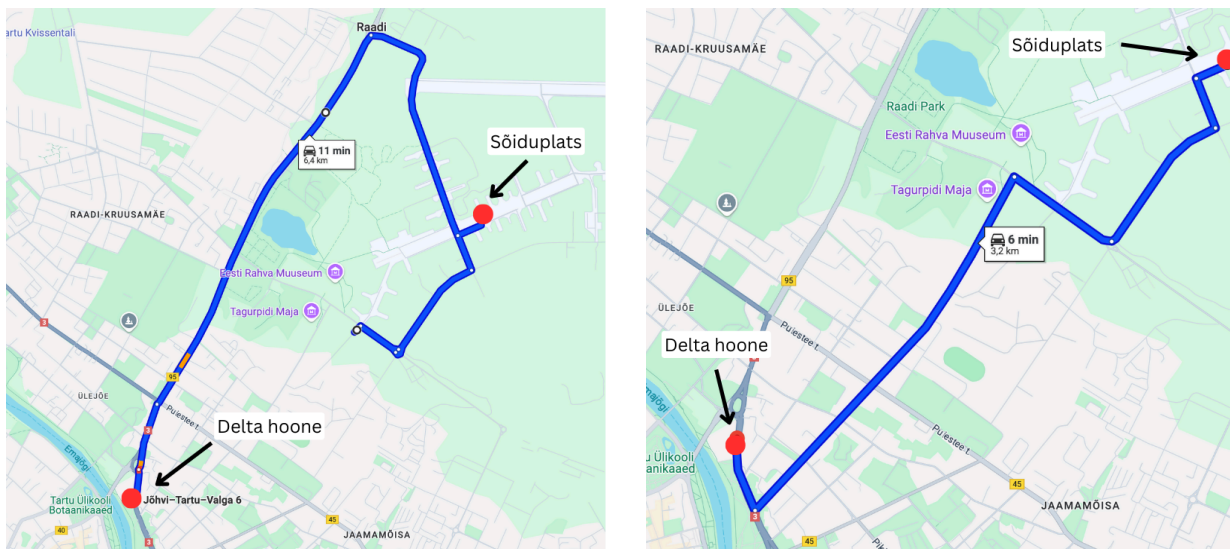
Erinevalt taju andmetest ei saanud täituri signaale omavahel otse joondada, kuna väärtuste vahemik ja dünaamika olid palju laiemad. Seetõttu kasutati sobivamaid ajaerinevuse mõõtmise meetodeid, mis võimaldasid arvestada signaalide mittekattuva iseloomuga. Rakendatud meetodid olid järgmised:

- Dünaamiline ajadeformatsioon (DTW, *dynamic time warping*) - algoritm, mis leiab kahe signaali punktide vahel optimaalse joonduse, võimaldades signaale venitada või kokku suruda ajas, et saavutada maksimaalne kattuvus. Käesolevas töös oli DTW sobiv, kuna käsu ja täituri vaheline viivitus ei pruukinud olla konstantse suurusega, sõltudes näiteks sõiduki kiirusest või hõõrduvusest. DTW võimaldas tuvastada keskmist ajalist nihet signaalide vahel ka siis, kui reageerimine oli ebäühtlane.
- Ristkorrelatsioon (XCORR, *cross-correlation*) - statistiline meetod, millega hinnatakse kahe signaali sarnasust erinevate ajanihete korral. Kõrgeim ristkorrelatsiooni tipp näitab, kui palju tuleb ühte signaali ajas nihutada, et see kattuks maksimaalselt teisega. Kuna ristkorrelatsioon põhineb lineaarsetel nihetel, oli see eriti kasulik siis, kui signaalid on sarnase kujuga, kuid ajaliselt nihkes.
- Fourier' teisendusel põhinev ristkorrelatsioon (FFT, *fast Fourier transform*) - eelnevale ristkorrelatsioonile sarnanev meetod, mida rakendatakse sagedusruumis. Ristspektri tagasi aegruumi teisendades saadakse ristkorrelatsioon, mille tipp näitab signaalide nihet. Antud töös kasutati meetodit teiste tulemustega võrdlemiseks.

Kuna viivitused ei ole ajas konstantse pikkusega, siis terve andmestiku peal arvutusmeetodite rakendamine ei annaks usaldusväärseid tulemusi, eriti väiksemate muutuste korral. Seetõttu kasutati sarnaste ajavahemike võrdlemiseks akendatud analüüsi: kogu andmestik jagati väiksemateks akendeks (näiteks 2-sekundilised aknad, mis on omavahel nihkes 1 sekund). Selle kasutamine laseb hinnata ajas muutuvat viivitust, arvestades andmete mittelineaarset iseloomu. Akendatud tulemuste põhjal oli võimalik arvutada statistilisi näitajaid (miinimum, maksimum, keskmine ja standardhälve). Lõpuks visualiseeriti andmeid joograafikutel, mis võimaldas hinnata viivituste varieeruvust.

## 2.4. Testimistingimused

Käsu viivituse mõõtmine eeldase kaugjuhtimise võimaldamist, mistõttu ei olnud seda võimalik sooritada Tartu linnatänavatel. Selle asemel kasutati Eesti Rahva Muuseumi (ERM) lähedal asuvat õppesõiduplatsi, mis tagas turvalise keskkonna roolimis- ja pidurduskäskude mõõtmiseks. Platsile olid paigutatud ka koonused, mille abil sai simuleerida reaalseid sõidumanöövreid. G2G mõõtmisi oli võimalik teostada pidevalt kogu sõidu vältel, kuna platsile liikumise ajal oli sõidukit juhtimas julgestusjuht ning videopildi olemasolu ei olnud juhtimiseks vajalik.



(a)

(b)

Joonis 11. Teekonnad Delta õppehoonest õppesõidu platsile (a) ja tagasi sõites (b).

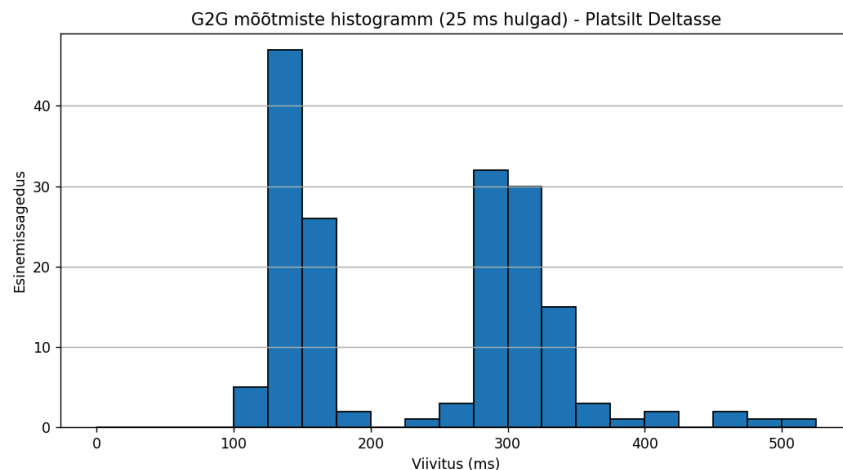
Esmased G2G mõõtmised koguti Delta hoonest väljudes ja platsile sõites, mille teekonnad on kujutatud joonisel 11. Kohale jõudes rakendati kaugjuhtimine, alustati uuesti taju ja täituri käsuandmete salvestamist ning viidi läbi lühike sõidutsükkel julgestaja juhendamisel. Kui mõõtmiste maht oli piisav, katkestati kaugjuhtimine ja tagasisõidul sooritati uuesti taju mõõtmised. Katsed toimusid veebruarikuus, mil maapinda kattis lumi ja ilmastikutingimused olid valdavalt pilvised.

### 3. Tulemused

Antud peatükis esitatakse eksperimentaalsete mõõtmiste tulemusi, keskendudes viivituse hindamisele erinevate andmekogumite ja meetodite lõikes. Tulemused on esitatud koos jooniste ja tabelitega, mis võimaldavad hinnata mõõtmiste ja signaalitöötluse täpsust. Arvesse on võetud kolme mõõtmiskomponenti - G2G mõõtmised ning täiturite vaatest pidurdamine ja pööramine.

#### 3.1. Taju viivitus

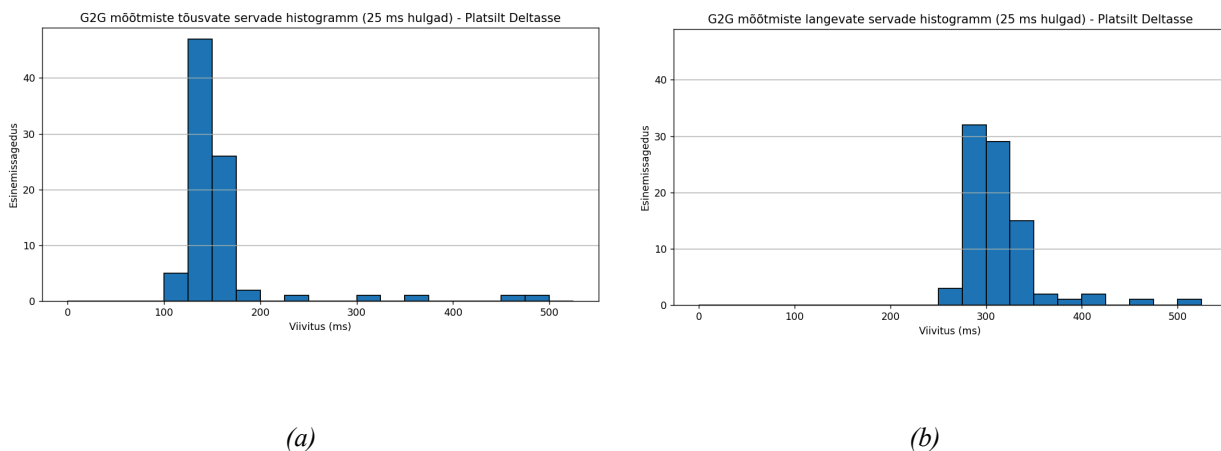
Esiialgu viidi läbi lokaalsed mõõtmised selliselt, et üks Raspberry Pi juhtis samaaegselt nii LED-i kui ka fototakistit. See lubas mõõta süsteemi viivitust, elimineerides võrgust ja videoedastamisest tulenevad viivitused. Nende mõõtmiste tõusva signaali keskmine tulemus oli 20 ms, mis on ootuspärane, kuna ainsana tekitavad viivitust LED-i ja anduri oleku muutumine ning andmete faili kirjutamine, millest ükski pole eriliselt aeganõudev. Sama mõõtmist sooritati kahe eraldi Raspberry Pi peal, kus saadi viivituseks 48 ms. Võib järeldada, et 30 ms vahe viivituses tuleneb NTP sünkrooni ebatäpsusest ning seda saab arvestada ülejäänud G2G tulemuste puhul.



Joonis 12. Histogramm taju viivitustest, liikudes õppesõiduplatsilt Delta suunas.

Joonisel 12 on kujutatud õppesõiduplatsilt Delta suunas sõidul mõõdetud viivitused, kusjuures on selgelt näha kahte eraldi paiknevat tippu. Tõusvate ja langevate signaalide eraldamisel selgus, et

langevad signaalid olid suurema viivitusega. Selle võimalikud põhjused olid fototakisti aeglane muutus pimenedes, LED-i või kuvari järelkuma kustumisel, kodeerimismeetod või kaamera automaatne heleduse kalibreerimine. Joonisel 13 on eraldatud andmed välja toodud ja edaspidi kasutatakse ainult tõusva serva andmeid taju viivituse määramiseks.



Joonis 13. Tõusva ja langeva signaali histogrammid, liikudes õppesõiduplatsilt Delta suunas.

Tabelis 1 on välja toodud kõigi kolme mõõtmiskorra tulemused ning tulemuste keskmine. Kõigis seadistustes jäi keskmine viivitus 143-156 ms vahemikku, viidates väikesele varieeruvusele erinevate keskkondade vahel. Langeva serva mõõtmised näitasid alati kõrgemaid viivitusi, mis toetab hüpoteesi riistvara põhjustatud hilinemisest langeva serva tuvastamisel.

Tabel 1. Taju viivituse mõõtmiste tulemused.

| Taju platsile minnes (ms)        | Keskmine viivitus | Miinum viivitus | Maksimum viivitus | Standardhälve |
|----------------------------------|-------------------|-----------------|-------------------|---------------|
| Kõik andmed                      | 212               | 44              | 1355              | 104           |
| Tõusev serv                      | 145               | 44              | 1355              | 91            |
| Langev serv                      | 279               | 142             | 568               | 65            |
| <b>Taju platsil sõites (ms)</b>  |                   |                 |                   |               |
| Kõik andmed                      | 333               | 118             | 994               | 230           |
| Tõusev serv                      | 144               | 118             | 221               | 15            |
| Langev serv                      | 525               | 272             | 994               | 182           |
| <b>Taju platsilt tulles (ms)</b> |                   |                 |                   |               |

|                                 |     |     |      |    |
|---------------------------------|-----|-----|------|----|
| Kõik andmed                     | 236 | 119 | 511  | 93 |
| Tõusev serv                     | 157 | 119 | 486  | 59 |
| Langev serv                     | 314 | 254 | 511  | 39 |
| <b>Taju mõõtmise kokku (ms)</b> |     |     |      |    |
| Tõusev serv                     | 149 | 44  | 1355 | 78 |

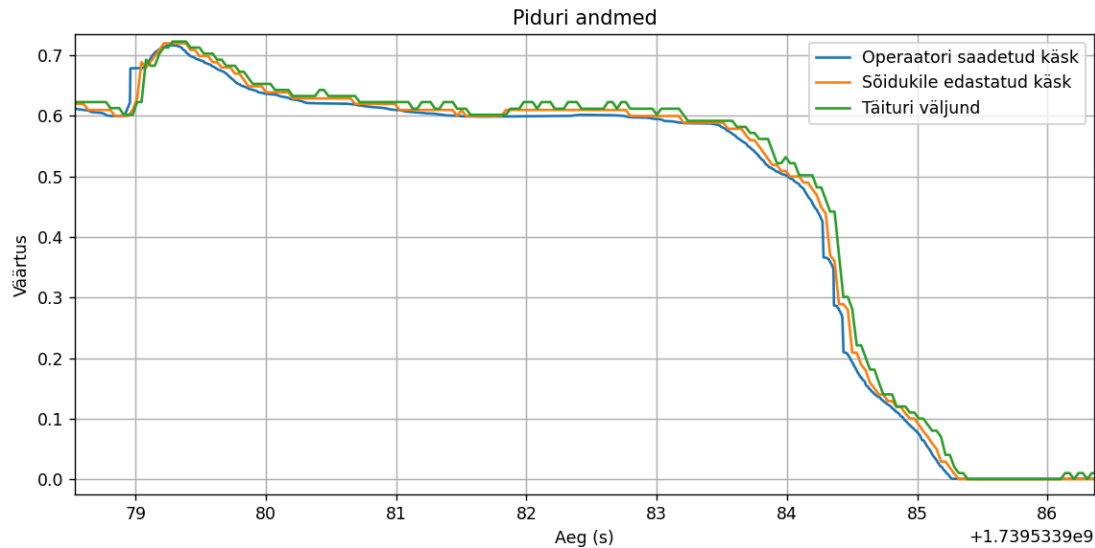
Kokkuvõttes, tõusva serva tulemuste keskmine oli 149 ms. Kui meetodite peatükis kirjeldatud komponentide hinnangulised viivitused kokku liita, saame tulemuseks 97 ms, ehk võrgu viivitus on ligikaudu 50 ms.

## 3.2. Käsu viivitus

Käsu viivituste osas analüüsiti pidurdamise ja pööramise andmeid eraldi. Läbi selle prooviti leida erinevate täiturite käitumises iseärasusi.

### 3.2.1. Pidurdamise viivitus

Pidurdamise viivituse hindamiseks kasutati andmeid kahest erinevast interpoleerimissagedusest: 50 Hz ja 125 Hz. Kummagi sageduse juures leitud tulemused on välja toodud tabelis 2 ning joonisel 14 on näidatud interpoleeritud andmeid. Akendamata analüüsi puhul näitasid XCORR ja FFT identseid viivitusi, mille keskmised olid 100 ms (50 Hz) ja 96 ms (125 Hz), kuna mõlemad meetodid lahendavad ristkorrelatsiooni. DTW tulemused olid palju madalamad, keskmistega 23 ms (50 Hz) ja 58 ms (125 Hz). Tõenäoliselt tulenevad DTW tulemused sellest, et andmetes oli suur osa 0-signaal, kuna enamuse sõidu ajast piduripedaali ei vajutatud. Terve andmestiku analüüsimisel ei eristata selliseid piirkondi, mistõttu arvestati suure arvu nullilähedaseid nihkumisi, keskmist omakorda madalamaks tõmmates.



*Joonis 14. Joongraafik pidurdamise andmetest. Sisendi ja edastatud käsu vahel mõõdetakse võrgu viivitust, käsu ja väljundi vahel täituri viivitust.*

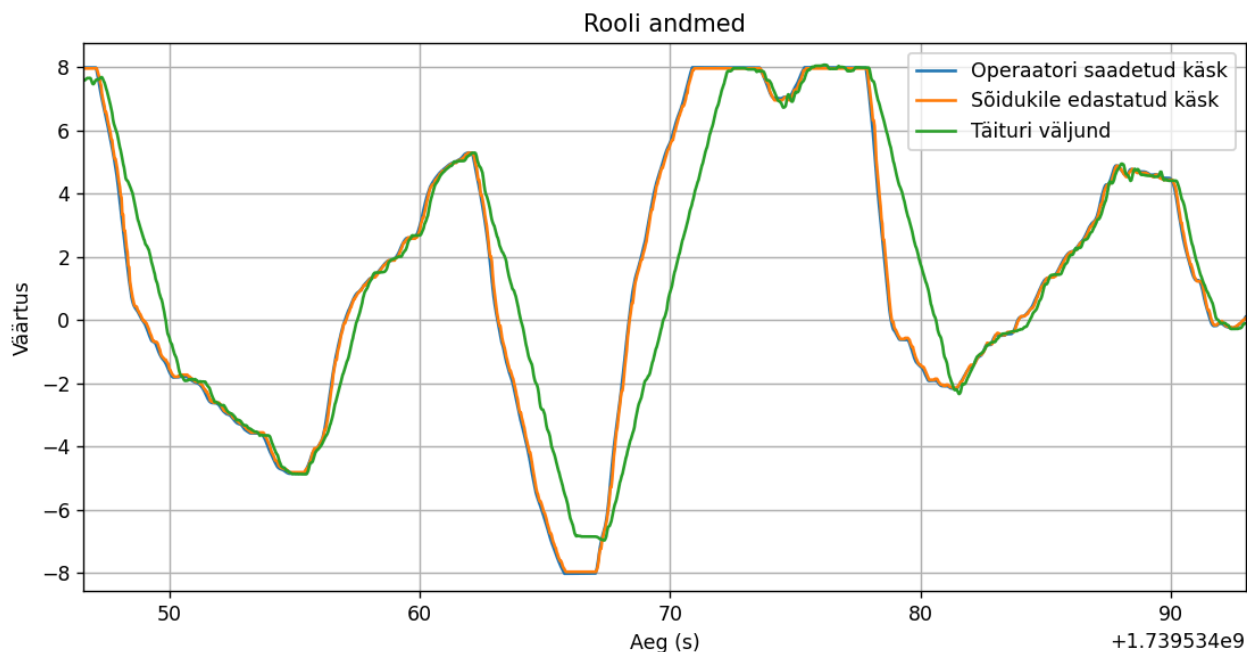
Kõigis akendatud meetodites olid erineva suurusega viivituste keskmised - XCORR-l 56 ms ja 58 ms, DTW-l 115 ms ja 114 ms ning FFT-l 176 ms ja 159 ms. Kui DTW tulemusi võrrelda kogu andmestiku ristkorrelatsiooni tulemustega, siis on ainus märkimisväärne erinevus täituri viivituses, mis on 89 ms (125 Hz).

Sageduste võrdluses selgus, et interpoleerimise sageduse tõstmine ei avaldanud akendatud tulemuste puhul olulist mõju, kuna keskmiste muutus väike. See tähendab, et ka madalama sagedusega andmetega on võimalik saavutada suhteliselt täpne tulemus, kuid kõrgem sagedus võimaldab peenemat viivituse eristamist.

Tabel 2. Pidurdamise viivituse tulemused 50 Hz ja 125 Hz interpoleerimissagedustega.

| <b>PIDURDAMINE 50 Hz (ms)</b>  | Kogu andmestiku keskmine | Akendatud keskmine | Akendatud miinimum | Akendatud maksimum | Akendatud standardhälve |
|--------------------------------|--------------------------|--------------------|--------------------|--------------------|-------------------------|
| <b>XCORR</b>                   |                          |                    |                    |                    |                         |
| võrgu viivitus                 | 40                       | 33                 | 20                 | 40                 | 9                       |
| täituri viivitus               | 40                       | 33                 | 20                 | 60                 | 15                      |
| kogu viivitus                  | 100                      | 56                 | 20                 | 100                | 27                      |
| <b>DTW</b>                     |                          |                    |                    |                    |                         |
| võrgu viivitus                 | 18                       | 50                 | 0                  | 183                | 36                      |
| täituri viivitus               | 12                       | 80                 | 1                  | 371                | 92                      |
| kogu viivitus                  | 23                       | 115                | 1                  | 434                | 107                     |
| <b>FFT</b>                     |                          |                    |                    |                    |                         |
| võrgu viivitus                 | 40                       | 28                 | 20                 | 40                 | 10                      |
| täituri viivitus               | 40                       | 38                 | 20                 | 60                 | 16                      |
| kogu viivitus                  | 100                      | 176                | 40                 | 820                | 250                     |
| <b>PIDURDAMINE 125 Hz (ms)</b> |                          |                    |                    |                    |                         |
| <b>XCORR</b>                   |                          |                    |                    |                    |                         |
| võrgu viivitus                 | 48                       | 29                 | 24                 | 32                 | 4                       |
| täituri viivitus               | 40                       | 37                 | 24                 | 48                 | 11                      |
| kogu viivitus                  | 96                       | 58                 | 32                 | 112                | 28                      |
| <b>DTW</b>                     |                          |                    |                    |                    |                         |
| võrgu viivitus                 | 18                       | 49                 | 2                  | 122                | 33                      |
| täituri viivitus               | 46                       | 89                 | 1                  | 562                | 122                     |
| kogu viivitus                  | 58                       | 114                | 5                  | 436                | 110                     |
| <b>FFT</b>                     |                          |                    |                    |                    |                         |
| võrgu viivitus                 | 48                       | 24                 | 8                  | 40                 | 11                      |
| täituri viivitus               | 40                       | 36                 | 16                 | 64                 | 16                      |
| kogu viivitus                  | 96                       | 159                | 8                  | 832                | 249                     |

### 3.2.2. Pööramise viivitus



Joonis 15. Joograafik pööramise andmetest. Sisendi ja edastatud käsu vahel mõõdetakse võrgu viivitust, käsu ja väljundi vahel täituri viivitust.

Pööramise andmeid (joonisel 15) analüüsiti 133 Hz andmesagedusel ning tabelis 3 on välja toodud selle tulemused. FFT meetodi puhul esines olukord, kus täituri viivitus on suurem kui kogu mõõtmise viivitus, mis ei ole füüsiliselt põhjendatav. Selline vastuolu võib viidata ebatäpsetele akendamise parameetritele kasutamisele või meetodi ebasobivusele.

Täituri võib olla seatud mõjutamise kiiruse piirang, ehk järsul rooli keeramisel pole rattad suutelised sama kiiresti asendit muutma ning viivitus suureneb aja jooksul kuni kästud asendi saavutamiseni. Seda on hästi näha suuremate olekumuutuste korral joonisel 15. Selle põhjal on DTW tulemus usaldusväärsem kuna meetod leiab igale signaali punktile sobivama nihkega punkti teisest signalist, samas kui XCORR ja FFT leiavad terve akna puhul keskmiselt kõige sobivama nihke.

Akendamata analüüsi puhul saadi XCORR ja FFT keskmisteks 418 ms, DTW tulemused oli keskmisega 585 ms. Kõigi akendatud meetoditega oli viivituste keskmine vahemikus 245-294 ms. Ka siin on akendatud DTW võrgu viivitus varasematega samas suurusjärgus (45 ms).

Tabel 3. Pööramise viivituse tulemused.

| <b>PÖÖRAMINE (ms)</b> | Kogu andmestiku keskmine | Akendatud keskmine | Akendatud miinimum | Akendatud maksimum | Akendatud standardhälve |
|-----------------------|--------------------------|--------------------|--------------------|--------------------|-------------------------|
| <b>XCORR</b>          |                          |                    |                    |                    |                         |
| võrgu viivitus        | 45                       | 33                 | 15                 | 60                 | 15                      |
| täituri viivitus      | 373                      | 243                | 7                  | 1180               | 301                     |
| kogu viivitus         | 418                      | 245                | 7                  | 1225               | 282                     |
| <b>DTW</b>            |                          |                    |                    |                    |                         |
| võrgu viivitus        | 9                        | 45                 | 0                  | 176                | 22                      |
| täituri viivitus      | 555                      | 274                | 0                  | 827                | 202                     |
| kogu viivitus         | 585                      | 294                | 0                  | 854                | 209                     |
| <b>FFT</b>            |                          |                    |                    |                    |                         |
| võrgu viivitus        | 45                       | 39                 | 7                  | 67                 | 14                      |
| täituri viivitus      | 373                      | 278                | 7                  | 829                | 238                     |
| kogu viivitus         | 418                      | 254                | 7                  | 799                | 197                     |

## 4. Järeldused

Käesolev töö näitas, et taju ja käsu viivitused on kaugjuhitava sõiduki usaldatavuse seisukohast olulised ja tihti suuremad, kui turvaliseks linnaliikluseks oleks vaja.

### 4.1. Taju viivitus

Tõusval serval mõõdetud keskmine viivitus oli 149 ms, millest ligikaudu 50 ms moodustab võrgu ja kodeerimise komponendid. Langeval serval tekkinud kõrgemad viivitused (kuni 525 ms) näitavad, et LED-i ja sensori reageerimise dünaamika mõjutab mõõtmist ja servasid tuleks arvesse võtta eraldi.

Viivitus oli kohati oodatust suurem, tekitades olukordi, kus LED-i olek muutus mitu korda enne anduri oleku muutust. Selle vastu saab pikendada LED-i põlemise ja kustus oleku aega selliselt, et nii pika viivituse tekkimine oleks vähem tõenäoline, näiteks > 5 sekundit.

### 4.2. Käsu viivitus

Pööramise viivitus on ligikaudu 300 ms, mille domineeriv osa on täituri mehaaniline reageerimisaeg. Akendatud DTW meetod andis kõige usaldusväärsema pildi, kuna ristkorrelatsioonil põhinevad lahendused eeldavad signaalide erinevate osade vahel võrdseid viivitusi, mis nende andmete puhul ei kehtinud. Pidurdamise viivitus on umbes 100 ms, mis jagunes üsna võrdselt võrgu viivituse ja täituri reageerimisaja vahel. Kuna andmed olid näiliselt stabiilse viivitusega ja vähem moonutatud võrreldes pööramise andmetega, siis olid akendamata XCORR ja FFT tulemused usaldusväärsemad. Samas leiti ka akendatud DTW abil sarnased tulemused.

Kui liita kokku taju viivitus (149 ms) ja suurim täituri keskmine viivitus (125 Hz pööramisel 294 ms) saame ligikaudse viivituse 450 ms, mis ületab turvalise manööverdamise piirina defineeritud 300 ms [7]. Lisades sellele inimese ligikaudse keskmise reaktsiooniaja (300 ms) [6] oleks terve otspunktide vaheline viivitus 750 ms. Kiirusega 50 km/h kulgeb sõiduk üle 10 m enne, kui täitur täielikult rakendub.

Seega ei ole praegune süsteem kaugjuhtimise kontekstis linna- ja maanteel manööverdamiseks piisavalt kiire. Tuleviku rakendustes peab latentsuse vähendamiseks täiendama videoedastust, võrguülest andmevahetust ja täiturite piiranguid.

## 5. Arutelu

Selles peatükis käsitletakse töö käigus tehtud mõõtmiste ja analüüsi peamisi kitsaskohti ning antakse soovitusi edaspidiseks täiustamiseks.

### 5.1. Taju viivituse mõõtmine

Mõõtmiseks kasutatud süsteemide dünaamika põhjustas kaks selget histogrammitippu. Sellel põhjusel pidi analüüsima tõusvaid ja langevaid servasid eraldi, mis vähendas analüüsitava andmepunktide arvu 2 korda. Võimalikeks põhjusteks peeti valguse sensori reageerimisaega, LED-i/kuvari lülitusaega või videopildi kodeerimist. Järgmiseks võiks kasutada kiiremini reageerivaid komponente, nagu fotodiodid või -transistor, ja sooritada mõõtmisi erinevate kuvarite ja sätetega.

Mõõtmistsüklite intervall (1 s) põhjustas aegajalt kattuvaid signaale, kui võrguside oli halvem. Edaspidi on soovitatav kasutada  $> 3$  s intervalle, et tagada andmepunktide isoleeritus ja vältida moonutusi.

Esines olukord, kus LED ühendus sõidu ajal Raspberry seadmest lahti, mistõttu oli vaja osa andmetest loobuda. Mõõteseadmete kinnitusmeetodid on vaja vastupidavamalt disainida.

NTP abil kella sünkroonimine ei ole täiesti täpne, kuna mõlemad Raspberry arvutid ei pruugi samal ajal serveri käest vastust saada. Seetõttu tasub uurida kas alternatiivseid sünkroniseerimise lahendusi või kellade omavahelise erinevuse määramise viise.

### 5.2. Käsu viivituse mõõtmine

Erinevalt taju viivitustest on andmete vahemik palju suurem, ehk analüüs on keerulisem. Terve andmestiku peal rakendatud meetodid tagastasid keskmised viivitused, mille tulemusi on keeruline visualiseerida.

XCORR-meetod andis akendatud analüüsis ebausaldusväärseid äärmuseid ( $> 1000$  ms), mis pälvis suure tõenäosusega müra või ebaoptimaalse akendamise tõttu. Soovitatav on XCORR asemel kasutada FFT-põhist korrelatsiooni või DTW-i, mis annavad stabiilsemaid tulemusi.

Signaalide interpoleerimine sagedustel 50 Hz ja 125 Hz näitas, et kõrgem andmesagedus võimaldab väiksemaid diskreetimisvigu, kuid üldiselt on keskmised viivitused sarnased. Tulevikus võiks kaaluda aga riistvaralist sünkroonimist, et vajadust interpoleerimiseks vähendada.

Pööramise mõõtmisel oli täheldatud vasturääkivust, et täituri viivitus oli suurem, kui kogu mõõtmise viivitus. See võib viidata vigasele akendamise konfiguratsioonile. Täienduslikult tuleks analüüsida parameetritundlikkust ja uurida võimalusi parameetrite optimeerimise automatiseerimiseks.

Leitud pööramise viivitus on liiga suur, et turvajuhti saaks asendada kaugjuhiga. Osaliselt mõjutab seda sõiduki ratastele rakendatud pööramiskiiruse piirang. Tuleks uurida, kas või mis suuruses oleks võimalik antud piiranguid vähendada. Siin tuleb arvestada ohuga, et ilma piiranguta ei pruugi sõiduk olla turvajuhile kontrollitav kaugjuhtimise ajal.

# Kokkuvõte

Käesoleva bakalaureusetöö eesmärgiks oli mõõta ja analüüsida viivitust, mis esineb Tartu Ülikooli isejuhtiva sõiduki ja Clevoni kaugjuhtimiskonsooli vahelises suhtluses. Eelkõige keskenduti taju ja käsu viivituse mõõtmisele ja mõistmisele. Taju viivituse all mõeldakse ajakulu kaamera sensoril registreeritud sündmusest kuni selle kuvamiseni monitori ekraanil. Käsu viivitus tähendab juhtimiskäsu loomisest kuni selle füüsilise rakendamiseni kuluvat aega ning mõõtmisi sooritati pööramise ja pidurdamise andmetega. Töö käigus disainiti ja rakendati vastavad mõõteseadmete prototüübid, kasutades varasemate uurimustel põhinevaid, kuid kasutatud süsteemile kohandatud meetodeid.

Taju viivituse mõõtmiste tulemusel leiti, et keskmine viivitus oli 149 ms, ulatudes maksimaalselt kuni 1355 ms-ni. Ristkorrelatsiooni abil leitud piduri viivitus oli keskmiselt 96 ms. Pööramise viivitus, mille puhul oli usaldusväärseim dünaamilise ajadeformatsiooni meetod, oli keskmiselt 296 ms. Kui liita kokku taju ning käskudest pööramise viivitused ja lisada juurde ka operatori reaktsiooniaeg (umbes 300 ms), on otspunktide vahel süsteemi viivitus ligikaudu 750 ms. Selline viivitus on kaugjuhtimise kontekstis probleemne, eriti linnaliikluses, kus olukorrad nõuavad kiiret ja sujuvat reageerimist.

Töö tulemused kinnitavad, et olemasolev kaugjuhtimissüsteem ei ole piisavalt täpne ega kiire, et tagada ohutu juhtimine keerulistes ja muutlikes liiklusoludes. Peamisteks kitsaskohtadeks tuvastati täiturite reageerimiskiirus, videoedastuse kõikuv viivitus ja võrguühenduse ebastabiilsus. Edasine arendus peaks keskenduma just nende tegurite optimeerimisele, sealhulgas väiksema viivitusega andmesidekanalite kasutusele võtmisele, andmetöötluse tõhustamisele ning üldise töökindluse suurendamisele.

Töö annab panuse autonoomsete ja kauhjuhitavate sõidukite arendusse, pakkudes praktilisi teadmisi ja mõõtemetodeid, mida saab kasutada ka tulevikus sarnaste süsteemide hindamisel ja täiustamisel.

# Tänuavaldused

Tänan Tambet Matiisenit ja Karl Kruusamäed juhendamise, metoodika uurimisel ja arendamisel nõustamise eest. Lisaks tänan testimisprotsessis abistamise eest Kertu Toompead ning mõttestadmete prototüüpide komponentide laenamise eest Veiko Vunderit ja Matevz Borjan Zoreci.

A handwritten signature in black ink, appearing to read 'Mäe'.

# Viidatud kirjandus

- [1] Creavin A. A Big Leap for Driverless Cars. *Wayleadr*, <https://wayleadr.com/blog/a-big-leap-for-driverless-cars/> (2024, vaadatud 12. aprill 2025).
- [2] Wakabayashi D, Fu C. Fatal Crash in China Puts Assisted Driving Tech Under Scrutiny. *The New York Times*, 1. aprill 2025, <https://www.nytimes.com/2025/04/01/business/xiaomi-driverless-electric-car-china.html> (1. aprill 2025, vaadatud 12. aprill 2025).
- [3] Data Analysis: Self-Driving Car Accidents [2019-2024]. *Craft Law Firm*, <https://www.craftlawfirm.com/autonomous-vehicle-accidents-2019-2024-crash-data/#key-findings> (vaadatud 12. aprill 2025).
- [4] Rosenzweig A. The success of self-driving vehicles will depend on teleoperation. *VentureBeat*, <https://venturebeat.com/ai/the-success-of-self-driving-vehicles-will-depend-on-teleoperation/> (2021, vaadatud 12. aprill 2025).
- [5] Testouri M, Elghazaly G, Hawlader F, et al. 5G-enabled teleoperated driving: An experimental evaluation. *arXiv [cs.ET]*, <http://arxiv.org/abs/2503.14186> (2025).
- [6] Human Benchmark, <https://humanbenchmark.com/tests/reactiontime/statistics> (vaadatud 28. aprill 2025).
- [7] Ellis SR, Young MJ, Adelstein BD, et al. Discrimination of changes in latency during head movement. *InHCI* 1999; 1129–1133.
- [8] Bachhuber C, Steinbach E. A system for high precision glass-to-glass delay measurements in video communication. *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE. Epub ahead of print september 2016. DOI: 10.1109/icip.2016.7532735.
- [9] Voysys. How we measure latency at Voysys, <https://www.youtube.com/watch?v=IKeLuvQAusQ> (2018, vaadatud 12. aprill 2025).
- [10] Jacobs MC, Livingston MA, Andrei State. Managing latency in complex augmented reality systems. *Proceedings of the 1997 symposium on Interactive 3D graphics - SI3D '97*. New York, New York, USA: ACM Press. Epub ahead of print 1997. DOI: 10.1145/253284.253306.
- [11] Georg J-M, Feiler J, Hoffmann S, et al. Sensor and actuator latency during teleoperation of automated vehicles. *2020 IEEE Intelligent Vehicles Symposium (IV)*. IEEE. Epub ahead of print 19. oktoober 2020. DOI: 10.1109/iv47402.2020.9304802.
- [12] Vehicle. *Autonomous Driving Lab*, <https://adl.cs.ut.ee/lab/vehicle> (vaadatud 5. mai 2025).
- [13] Tartu Ülikool ja Bolt esitlesid uue isejuhtivate sõidukite labori katseautot. *Tartu Ülikool*, <https://ut.ee/et/sisu/tartu-ulikool-ja-bolt-esitlesid-ue-isejuhtivate-soidukite-labori-katseautot> (vaadatud 19. mai 2025).
- [14] Isejuhtiv Lexus sõidab virtuaalses ülikoolilinnas. *Tartu Ülikool*, <https://ut.ee/et/sisu/isejuhtiv-lexus-soidab-virtuaalses-ulikoolilinnas> (vaadatud 19. mai 2025).

- [15] Cooperation between Clevon and ADL. *Autonomous Driving Lab*, <https://adl.cs.ut.ee/blog/cooperation-between-clevon-and-adl> (vaadatud 19. mai 2025).
- [16] Juht IK ee. Clevon ja Tartu Ülikool arendavad sõidukite kaugjuhtimist. “Uute lahenduste loomiseks”. *IT-uudised*, <https://www.ituudised.ee/uudised/2024/04/08/clevon-ja-tartu-ulikool-arendavad-soidukite-kaugjuhtimist-uute-lahenduste-loomiseks> (vaadatud 19. mai 2025).
- [17] SEKONIX, <http://www.sekonix.com/?param=camera> (vaadatud 5. mai 2025).
- [18] 34" CJ791 Series Curved Ultra-Wide Monitor LC34J791WTNXZA. *Samsung Electronics America*, [https://www.samsung.com/us/business/computing/monitors/curved/34-j791-thunderbolt-3-ultra-wide-screen-curved-monitor-lc34j791wtnxza/?utm\\_source=chatgpt.com](https://www.samsung.com/us/business/computing/monitors/curved/34-j791-thunderbolt-3-ultra-wide-screen-curved-monitor-lc34j791wtnxza/?utm_source=chatgpt.com) (vaadatud 5. mai 2025).
- [19] Lhostis R. Video Encoding & Compression: Why Storage & Quality Matter, <https://www.harmonicinc.com/insights/blog/video-encoding/> (2023, vaadatud 6. mai 2025).
- [20] Roux L, Gouaillard A. Performance of AV1 Real-Time Mode, <http://arxiv.org/abs/2009.14165> (2020, vaadatud 6. mai 2025).
- [21] Yi J, Islam MR, Aggarwal S, et al. An analysis of delay in live 360° video streaming systems. *Proceedings of the 28th ACM International Conference on Multimedia*. New York, NY, USA: ACM. Epub ahead of print 12. oktoober 2020. DOI: 10.1145/3394171.3413539.
- [22] Chandrika Sharma, Nidhi Sharma, Poornima Sharma, Veerendra Kumar. A Review of BJT Based Phototransistor. *International Journal of Engineering Research & Technology (IJERT)*, <https://www.ijert.org/research/a-review-of-bjt-based-phototransistor-IJERTV3IS040020.pdf> (vaadatud 19. mai 2025).
- [23] NTE Electronics, Inc. Photoresistors. *LDR 5mm, 12mm, & 20mm Radial Lead Type Photoresistors | NTE Electronics, Inc.*, <https://www.tme.eu/Document/a6885e43cec8543bb6ebffb078f868f/02-LDR1.pdf> (vaadatud 19. mai 2025).
- [24] Raspberry Pi Ltd. Raspberry Pi 4. *Raspberry Pi*, <https://www.raspberrypi.com/products/raspberrypi-4-model-b/> (vaadatud 13. aprill 2025).
- [25] Calibration Hemisphere by Balster. *Thingiverse*, <https://www.thingiverse.com/thing:888361> (vaadatud 28. aprill 2025).
- [26] PrusaSlicer. *Prusa3D by Josef Prusa*, [https://www.prusa3d.com/page/prusaslicer\\_424/](https://www.prusa3d.com/page/prusaslicer_424/) (vaadatud 4. mai 2025).
- [27] Raspberry Pi Ltd. Buy a Raspberry Pi 5. *Raspberry Pi*, <https://www.raspberrypi.com/products/raspberrypi-5/> (vaadatud 28. aprill 2025).
- [28] KY-018 Photoresistor, <https://sensorkit.joy-it.net/en/sensors/ky-018> (vaadatud 28. aprill 2025).
- [29] 10K Ohm Potentiometer. *QuartzComponents*, <https://quartzcomponents.com/products/10k-potentiometer> (vaadatud 28. aprill 2025).
- [30] 4 microphones with analog pins Raspberry Pi 5 - Raspberry Pi Forums,

<https://forums.raspberrypi.com/viewtopic.php?t=371941> (vaadatud 29. aprill 2025).

- [31] Raspberry Pi RP1 Peripherals, <https://datasheets.raspberrypi.com/rp1/rp1-peripherals.pdf> (vaadatud 29. aprill 2025).
- [32] Making an adjustable resistance. *Electrical Engineering Stack Exchange*, <https://electronics.stackexchange.com/questions/67333/making-an-adjustable-resistance> (vaadatud 29. aprill 2025).
- [33] How do I save terminal output to a file? *Ask Ubuntu*, <https://askubuntu.com/questions/420981/how-do-i-save-terminal-output-to-a-file> (vaadatud 5. mai 2025).
- [34] *as\_pacmod.dbc at 3.4.1.1 · astuff/pacmod\_dbc*. Github, [https://github.com/astuff/pacmod\\_dbc/blob/3.4.1.1/as\\_pacmod.dbc#L554-L565](https://github.com/astuff/pacmod_dbc/blob/3.4.1.1/as_pacmod.dbc#L554-L565) (vaadatud 5. mai 2025).

## Lisad

**Lisa 1.** Töö andmeanalüüsiks kasutatud programmidega Google Drive kaust koos testimise käigus salvestatud andmetega:

[https://drive.google.com/drive/folders/1Ho7C4OA6wIvY3VGyQ\\_D8JhbQuHKXFg7x?usp=drive\\_link](https://drive.google.com/drive/folders/1Ho7C4OA6wIvY3VGyQ_D8JhbQuHKXFg7x?usp=drive_link)

# **Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks**

Mina, Mattias Mäe

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) minu loodud teose

## **“Sõiduki kaugjuhtimise viivituse mõõtmine”**

mille juhendajad on Tambet Matiisen ja Karl Kruusamäe

reprodutseerimiseks eesmärgiga seda säilitada, sealhulgas lisada Tartu Ülikooli digitaalarhiivi kuni autoriõiguse kehtivuse lõppemiseni;

2. annan Tartu Ülikoolile loa teha punktis 1 nimetatud teos üldsusele kättesaadavaks Tartu Ülikooli veebikeskkonna, sealhulgas digitaalarhiivi kaudu Creative Commons'i litsentsiga CC BY NC ND 4.0, mis lubab autorile viidates teost reprodutseerida, levitada ja üldsusele suunata ning keelab luua tuletatud teost ja kasutada teost ärieesmärgil, kuni autoriõiguse kehtivuse lõppemiseni;
3. olen teadlik, et punktides 1 ja 2 nimetatud õigused jäävad alles ka autorile;
4. kinnitan, et lihtlitsentsi andmisega ei riku ma teiste isikute intellektuaalomandi ega isikuandmete kaitse õigusaktidest tulenevaid õigusi.

Mattias Mäe

**20.05.2025**