

Tartu Ülikool
Loodus- ja täppisteaduste valdkond
Tehnoloogiainstituut

Hendrik Olesk

**Nägemisulatuses kaugjuhitava mobiilse robotmanipulaatori
kasutajamugavuse tõstmine**

Bakalaureusetöö (12 EAP)
Arvutitehnika eriala

Juhendaja:
Robotika kaasprofessor Karl Kruusamäe
Nooremteadur Robert Valner

Tartu 2021

Resümee

Nägemisulatuses kaugjuhitava mobiilse robotmanipulaatori kasutajamugavuse tõstmine

Mobiilset manipulaatorit KUKA youBot saab kasutada robotika-alaste ülesannete täitmiseks, kus on vaja keskkonnas liikuda ning sellega suhelda. Sellel robotil puudus tehniline lahendus, mille käivitamiseks ja kasutamiseks ei ole vaja eelnevaid teadmisi roboti tarkvarast. Käesolevas töös loodi tarkvarapakett youbot_ux, millega saab minimaalsete eelteadmistega kasutada KUKA youBoti juhtimispuldiga Sony DualShock 4 neljas erinevas režiimis: baasi sõidurežiim, lüli haaval manipulaatori juhtimine, manipulaatori trajektoori õpetamise- ja taasesituse režiim ning ohutu režiim. Tulemusena on võimalik robotit juhtida, lülitades selleks sisse roboti pardaarvuti, mootorid ning juhtimispult. Pardaarvuti ja mootorite töösolekul käivitatakse juhtimiseks vajalik tarkvara automaatselt.

Võtmesõnad: kasutajamugavus, kaugjuhtimine, mobiilne manipulaator, mängupult, youBot, ROS, manipulaator, robotika

CERCS: T120 Süsteemitehnoloogia, arvutitehnoloogia; T125 Automatiseerimine, robotika, control engineering

Abstract

Improving the usability of a mobile manipulator robot for line-of-sight remote control

The KUKA youBot mobile manipulator can be used to perform robotics tasks that require moving and interacting with the environment. The robot did not have a technical solution that does not require prior knowledge of the robot's software to start and use it. In this thesis, the youbot_ux software package was created, which can be used with the KUKA youBot remote control Sony DualShock 4 in four different modes with minimal prior knowledge: base driving mode, link-by-link manipulator control, manipulator trajectory teaching and playback mode, and safe mode. As a result, the robot can be controlled by turning on the robot's on-board computer, motors and control panel. When the on-board computer and motors are running, the control software is started automatically.

Keywords: user convenience, remote control, mobile manipulator, game console, youBot, ROS, manipulator, robotics

CERCS: T120 Systems engineering, computer technology; T125 Automation, robotics, control engineering

Jooniste loetelu

Joonis 1. <i>KUKA youBot [8], Curiosity kulgur [9], Digital Vanguard kaugjuhitav sõiduk [10]</i>	10
Joonis 2. <i>Comau roboti õpetamisseade [17]</i>	11
Joonis 3. <i>iRobot uPoint MRC süsteem [24]</i>	12
Joonis 4. <i>Xbox 360 [27] ja DualShock 4 [28] kontrollid</i>	13
Joonis 5. <i>Üldine ülevaade süsteemist, sisendseadmest robotini</i>	17
Joonis 6. <i>Ülevaade lahenduse ROSi sõlmedest</i>	18
Joonis 7. <i>Kontrolleri nuppude ja juhtkangide asetused</i>	21
Joonis 8. <i>Kontrolleri konfiguratsioon roboti sõitmise režiimis</i>	22
Joonis 9. <i>Kontrolleri konfiguratsioon manipulaatori lüli haaval juhtimise režiimis</i>	22
Joonis 10. <i>Kontrolleri konfiguratsioon õpetamise režiimis</i>	23
Joonis 11. <i>Kontrolleri konfiguratsioon ohutu režiimis</i>	23
Joonis 12. <i>Youbot_driver_ros_interface sõlme ühendused</i>	25
Joonis 13. <i>Youbot_states olekumasina sõlme ühendused</i>	26
Joonis 14. <i>Youbot_drive_joy sõlme ühendused</i>	27
Joonis 15. <i>Youbot_velocity_move ja youbot_velocity_move_grasp sõlme ühendused</i>	28
Joonis 16. <i>Youbot_trajectory_record, trajectory_recorder ja trajectory_replayer sõlme ühendused</i>	29
Joonis 17. <i>Ds4_driver ja youbot_states_feedback sõlme ühendused</i>	31

Tabelite loetelu

Tabel 1. <i>Sõlmede kirjeldused</i>	19
Tabel 2. <i>Puldi LEDi ja vibratsiooni tagasiside režiimi valikul</i>	24

Sisukord

Resümee	2
Abstract	3
Jooniste loetelu	4
Tabelite loetelu	5
Sisukord	6
1 Sissejuhatus	8
2 Kirjanduse ülevaade	9
2.1 Robotite juhtimine, autonoomsus ja automaatika	9
2.2 Mobiilsed manipulaatorid	10
2.3 Roboti juhtimisseadmed	10
2.3.1 Roboti õpetamisseade	11
2.3.2 Klaviatuur	12
2.3.3 Puutetundlik ekraan	12
2.3.4 Mängukontroller	13
2.4 Teadusuuringud juhtseadmete kasutatavuse kohta	13
3 Ülevaade probleemist ja töö nõuded	15
3.1 Eesmärk	15
3.2 Olukord töö alguses	15
3.3 Nõuded tööle	16
3.3.1 Funktsionaalsed nõuded	16
3.3.2 Muud nõuded	16
4 Töö lahendus	17
4.1 Arhitektuuri ülevaade	17
4.2 Süsteemi kasutamine	20
4.3 Süsteemi ülesehitus	24
4.3.1 YouBoti draiveri sõlme ühendused	24
4.3.2 Olekumasin ja ressursihaldur	25
4.3.3 KUKA youBoti baasi juhtimine	27
4.3.4 KUKA youBoti manipulaatori juhtimine	28
4.3.4.1 Manipulaatori lüli haaval liigutamine	28
4.3.4.2 Manipulaatori trajektoori salvestamine ja taas esitamine	29
4.3.5 Kontrolleri kasutus ja tagasiside	31
5 Tulemused	33
Kokkuvõte	35

Viited	36
Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks	40

1 Sissejuhatus

Robotid on praegusel ajal tihti kasutuses kodudes ning veel enam tööstustes [1] töödeks, mida saab robot teha efektiivsemalt kui inimene või mis on inimese jaoks ohtlikud [2]. Sellest hoolimata ei ole robotite kasutamine alati lihtne, sest keerukamatel juhtudel vajavad need kasutajapoolset programmeerimisoskust või spetsiifilist väljaõpet.

Tartu Ülikooli tehnoloogiainstituudis on roboteid, mida kasutatakse arendusprojektide läbiviimiseks õppe- ja teadustööde raames. Need on head tehniliste ülesannete harjutamiseks ja arendusprojektide elluviimiseks, kuid aeg-ajalt on tarvis roboteid kiiresti ja minimaalse vaevaga demonstreerida või liigutada. Selle tarvis oleks head kasutajaliidesed, millega roboti poolt pakutavaid võimalusi saaks lihtsalt näidata. Mobiilsel manipulaatoril KUKA youBotil sellist liidest pole ning kuna roboti tootmine on lõpetatud [3], siis toimub selle arendustegevus valdavalt roboti kasutajate poolt ülikoolides ja ettevõtetes.

Käesoleva lõputöö eesmärgiks on luua intuitiivne ja hõlpsasti käitav moodus KUKA youBoti juhtimiseks. Loodav lahendus lubab roboti baasi ja manipulaatorit juhtida ilma, et kasutaja peaks robotit programmeerima või käsurealt demoprogramme käivitama. Lisaks otsesele juhtimisele luuakse võimalus robotmanipulaatori kinesteetiliseks õpetamiseks: liigutustrajektoore salvestamiseks, kui kasutaja liigutab robotit, ja hilisemaks iseseisvaks taasesitamiseks.

Kogu lõputöö on organiseeritud viide peatükki. Kirjanduse ülevaates (peatükk 2) tuuakse välja info robotite juhtimismeetoditest (autonoomsus, automaatika, teleopereerimine), mobiilsetest manipulaatoritest ning roboti juhtseadmetest. Kolmandas peatükis tuuakse välja töö detailne eesmärk ning tehnilised nõuded. Neljandas peatükis tutvustatakse valminud lahendust ja viiendas peatükis analüüsitakse tulemusi.

2 Kirjanduse ülevaade

2.1 Robotite juhtimine, autonoomsus ja automaatika

Roboteid kasutatakse paljudes valdkondades, näiteks logistikas, tööstuses ja tervishoius ning üha rohkem ka kodudes [1]. Robotid võivad erineda valdkonnasiseselt ka töö otstarvete poolest. Näiteks logistikas kasutatakse neid pakkimisel, kuid ka objektide vedamisel [1]. Sellest tulenevad ka erinevad viisid, kuidas roboti juhtimine on üldiselt üles seatud. Need võivad olla programmeeritud täitma rangelt ettemääratud ülesannet (automatiseerimine), kasutaja poolt otseselt juhitud (teleopereerimine ehk kaugjuhtimine) ning ka autonoomsed [1].

Automaatikat kasutatakse korduva liikumisega tööde tegemisel inimesepoolse juhtimiseta [1]. Sellisel juhul programmeeritakse ette kindel liikumine ja tegevus, mida robot korduvalt täitma peab [1]. Automaatika on kasutusel näiteks autotööstustes autode osade paigaldamisel [1].

Autonoomsed robotid on kasutusel juhtudel, kus nende arendamine on erinevate sensorite abil olnud võimalik ning inimeste sekkumine pole vajalik [4]. Robotite korral tähendab autonoomsus suutlikust saada infot keskkonnast ning teha saadud infole vastavaid järeldusi ja otsuseid oma tegevuse korrigeerimiseks [4]. Näiteks on selline roboti juhtimisviis kasutusel robottolmuimejal Roomba [5], mis võimeline liikuma vastavalt ettetulevatele takistustele ning tuvastama ka mustuse taset, muutes sellest tulenevalt oma töö režiimi ja suunda [4].

Kuigi autonoomsed robotid on võimelised tegema ettenähtud tööd iseseisvalt, on valdkondi ja ülesandeid, kus kasutatakse otsest juhtimist, ehk kus kasutaja juhib robotit ise [2]. Otsese juhtimise korral saab muuta roboti käitumist vastavalt vajadusele, kuna robot on kasutaja poolt pidevalt kontrollitav, olles seejärel ka tööülesannete poolest paindlik [2]. Teleopereeritavat robotit kasutatakse ka olukordades, kus on vaja inimese taju ja oskusi, kuid kus kasutaja ise viibida ei saa [2]. Üheks taolise roboti kasutamise olukorraks võib olla radioaktiivse materjali liigutamine [2]. Teleopereeritavaid roboteid kasutatakse ka kirurgias, kus on vajalik inimese käest täpsem ning vibratsioonivaba liikumine [2].

2.2 Mobiilsed manipulaatorid

Mobiilsed manipulaatorid on robotid, mis koosnevad mobiilsest baasist ning manipulaatorist [6]. Mobiilne baas on robot, mis on võimeline liikuma [6]. Manipulaator on seade, mis koosneb enamasti haaratsist ja mitmest liikuvast lülis, millega saab keskkonda mõjutada, näiteks esemeid tõstes [7]. Sellisteks robotiteks on näiteks KUKA youBot, Curiosity kulgur ja Digital Vanguard kaugjuhitav sõiduk (joonis 1).



Joonis 1. *KUKA youBot [8], Curiosity kulgur [9], Digital Vanguard kaugjuhitav sõiduk [10]*

Mobiilsed manipulaatorid on võimelised liikuma ning viima läbi tegevusi sarnaselt inimesele. [6]. Digital Vanguard robotiga saab manipulaatori abiga vaadata ja olenevalt sellel olevatest seadmetest teha kahjutuks pomme, säästes seejuures inimeste tervist ja elusid [9]. Curiosity kulgur võimaldab Marsil liikudes manipulaatoriga võtta pinnaseproove, teha fotojäädvustusi jm soovitud kohtades [11].

Mobiilsete robotite kaugjuhtimise juures on olulisel kohal nende kasutatavus. Kaugjuhtimisel tekib ajaline viide kasutaja poolt robotile sisendi saatmise ning roboti tegeliku liikumise vahel. Samuti võib roboti tagasiside saabuda viitega, mille tõttu ei pruugi kasutaja piisavalt kiiresti roboti liikumisele reageerida. Roboti kaugelt kontrollimise efektiivsus oleneb ka kasutajaliidesest, näiteks selle kasutamise lihtsusest ja info kogusest. [12]

2.3 Roboti juhtimisseadmed

Robotite teleopereerimiseks on võimalik kasutada erinevaid sisendseadmeid. Sisendseadmete kaudu saab saata juhiseid, mille järgi robot oma tööd teeb või korrigeerib [13]. Võimalikud

sisendseadmed on näiteks roboti õpetamisseadmed (ingl *teach pendant*), klaviatuurid, puutetundlikud ekraanid ja mängukontrollerid [13][14][15].

2.3.1 Roboti õpetamisseade

Tööstuslikke roboteid kontrollitakse enamasti roboti õpetamisseadmega (joonis 2) [16]. Antud seade võib koosneda mitmest osast, näiteks klaviatuurist ja ekraanist, olles samal ajal mobiilselt kasutatav [16]. Õpetamisseadmes võivad koos olla nupud, lülitid, puutetundlik ekraan, kui ka ohutusabinõud, nagu roboti hädaseiskamisnupp [16]. Selle abil seadistab kasutaja roboti tegema ettenähtud tööd, andes roboti liikumiseks ette sobivaid käsklusi ja positsioone programmeerimis- või skriptimiskeeltes [16]. Samuti on õpetamisseadmega võimalik roboti otsesem juhtimine, näiteks roboti positsiooni korrigeerimiseks [16].



Joonis 2. Comau roboti õpetamisseade [17]

Õpetamisseadmete korral on olulisel kohal ühilduvus teiste seadmetega ehk kas ühte seadet on võimalik kasutada erinevate tootjate robotite programmeerimiseks [18]. Üldiselt on õpetamisseadmed aga tootjaspetsiifilised, mistõttu ühte seadet ei pruugi olla võimalik kasutada teise tootja robotil [18]. Seda ka seetõttu, et erinevatel robotitel on erinevad programmeerimis- või skriptimiskeeled, mida kasutatakse roboti programmeerimiseks [18]. Kasutaja jaoks on oluline roboti kiire ülesseadmine. Seda võimaldab seadmes olevate menüüde lihtsus [18].

2.3.2 Klaviatuur

Klaviatuuri kasutamine on robotite juhtimisel üldteada, kuna tegemist on seadmega, mida rakendatakse arvutitega seondult sageli. Seda on kasutatud nii liikuva valveroboti rataste kui ka manipulaatori kontrollimisel [19]. Klaviatuuri plussiks on tema odavus ning tänu rohketele klahvidele võimalus paljude erinevate funktsioonide rakendamiseks [20]. Klaviatuuri kasutamine võib aga olla keeruline juhul, kui soovitakse täpsemat väärtuste valikut - näiteks roboti kiiruse valik [20][21]. Klaviatuuri klahvi vajutuse väärtused on binaarse lülituse tõttu diskreetsed, varieerudes vaid 2 väärtuse vahel (näiteks 1 ja 0) [21].

2.3.3 Puutetundlik ekraan

Puutetundlike ekraanide korral on võimalik ekraanil kuvada samaaegselt nii juhitava seadme kohta käivat infot kui ka käske, mida kasutaja ekraani kasutades valida saab [22]. Seadmete, nagu näiteks nutitelefonide ning tahvelarvutite (joonis 3) puutetundlikku ekraani on kasutatud juhtimisseadmena, kus ekraanil kuvatakse käske, mida robot saab teostada [23]. Ühe sellise roboti näiteks on Lego Mindstorms, mille liikumise suunda ja kiirust saab muuta puutetundliku ekraani kaudu [14]. Lisaks roboti juhtelementide ekraanil kuvamisele on võimalik kasutajale näidata ka muud informatsiooni, nagu roboti hetkepositsiooni ja liikumise trajektoori [23].



Joonis 3. *iRobot uPoint MRC süsteem [24]*

2.3.4 Mängukontroller

Mängukontrollerid on seadmed, mis üldiselt koosnevad nii nuppudest kui ka juhtkangidest [25]. Mängukonsooli pulte on kasutatud mängudes erinevate objektide juhtimiseks, näiteks sõidukid ja lennukid, ning need on leidnud kasutust ka robotite juhtimisel [15]. Näiteks kaevanduste jaoks mõeldud päästerobotit Gemini-Scout kontrollitakse Xbox mängukonsooli puldiga [26].

Võrreldes klaviatuuriga on mängukontrolleril (nt näiteks Xbox 360 [27] ja DualShock 4 [28] joonisel 4) olemas ka analoog-juhtkangid, mille väljundiks on pidev signaal kangi asendi kohta [25]. Saadava pideva väärtusega on võimalik juhitava roboti kiirust täpsemini kontrollida, kui nupu vajutamisel [25]. Selliseid kontrollereid on kasutatud näiteks differentiaalse liikumismehhanismiga roboti juhtimiseks [29] ja ka roboti manipulaatori ning haaratsi juhtimiseks [30].



Joonis 4. *Xbox 360 [27] ja DualShock 4 [28] kontrollid*

2.4 Teadusuuringud juhtseadmete kasutatavuse kohta

Kasutajasõbraliku roboti juures on oluline selle kasutamise mugavus ja lihtsus. Politecnico di Torino ülikoolis viidi läbi uuring, mille eesmärk oli leida parim kontroll, mis sobib robotiga mööda ettenähtud trajektoori liikumiseks, ruumis ettenähtud asukoha otsimiseks ning esemete liigutamiseks [31]. Juhitavateks robotplatvormideks kasutati robotmanipulaatorit, maasliikuvat kulgurit ja lendavat drooni [31]. Leiti, et kõige kiiremini sai ülesandeid täita kasutades robotite juhtimiseks nutitelefonide võimalusi, nagu güroskoopi ja puutetundlikku ekraani [31]. Kuigi nutitelefonide kasutades saadud tulemused olid ülesannete täitmisel kiireimad, jäi uuringus

osalenud kasutajate tagasiside kohaselt parimaks liideseks oma üldise tuntuse tõttu mängukonsooli kontrollid [31].

Kechavarzi jt poolt läbi viidud uuringu eesmärk oli leida kasutajatele kõige sobivam kontrollid järgnevate ülesannete täitmiseks: objekti lükkamine soovitud asukohta, takistuse vältimine ning mööda ette seatud trajektoori liikumine [32]. Omavahel võrreldi arvuti klaviatuuri, mängukonsooli pulti ja tahvelarvutit - kasutajate vaatepunktist oli roboti juhtimiseks kõige parem valik mängukonsooli pult selle kasutamise lihtsuse, kasutama õppimise lihtsuse ja mugavuse poolest [32].

Mängukonsooli pulti on uuritud ka tööstusrobotite kasutamisel, olles alternatiiviks robotite õpetamisseadmetele [15]. Uuringus, kus ülesanneteks olid manipulaatoriga objekti tõstmine ja liigutamine ning trajektoori järgimine, leiti, et ülesanded olid mängukonsooli puldiga kiiremini läbitavad, kui roboti õpetamisseadmega [24]. Kasutajate tagasiside näitas, et konsoolipult oli kasutajate poolt kõige lihtsamini kasutatav [15].

3 Ülevaade probleemist ja töö nõuded

3.1 Eesmärk

Töö eesmärgiks on luua tarkvaralahendus, mille abil saaks KUKA youBot robotit juhtida roboti süsteemi detailselt tundmata.

Täpsemalt on käesoleva töö eesmärgiks luua võimalused:

- roboti baasi ja manipulaatori kontrolleriga juhtimiseks;
- roboti õpetamiseks trajektoori salvestamise ja ettemängimise viisil.

3.2 Olukord töö alguses

Töös kasutatav KUKA youBot koosneb nelja omnisuunalise rattaga liikuvast baasist ning sellega ühendatud 5-lülilisest haaratsiga manipulaatorist [33]. Olemasolev tarkvaralahendus töötab operatsioonisüsteemil Ubuntu 16.04 ja ROSi (Robot Operating System) versioonil Kinetic Kame, mille ametlik tugi aegus 2021. aasta aprillis [34]. Robot on kasutatav youBoti ROSi draiverite ja MoveIt tarkvarakimbuga. Roboti baasi juhtimine viiakse läbi klaviatuuri abil, mis on ühendatud roboti pardaarvutiga otse või võrgu kaudu. Baasi klaviatuuri abil juhtimiseks on vaja käivitada youBoti ROSi draiver ning seejärel *teleop_twist_keyboard* sõlm (ingl *node*). Sõlme kasutatakse rubriikide (ingl *topic*) kaudu sõnumite ehk soovitud info kuulutamiseks või ka lugemiseks. Seega, kuulutatakse *teleop_twist_keyboard* sõlmega roboti poolt seatavaid baasi kiiruseid. Manipulaatori kontrollimiseks kasutatakse RViz programmi, milles määratletakse roboti manipulaatori poolt jälgendatav positsioon. KUKA youBoti ROSi draiveri töösolekul saab manipulaatori juhtimiseks kasutada RViz programmi, mis käivitatakse MoveIt tarkvarapaketi poolt pakutava demo käivitusfailiga. Roboti manipulaatorit juhitakse soovitud positsiooni liigutades RViz programmis olevat roboti 3D mudeli manipulaatorit.

Kuigi roboti juhtimine on tehniliselt võimalik, puudub selleks kergelt kasutatav lahendus, mis ei vajaks roboti kasutamiseks kasutajapoolset ROSi spetsiifilist tarkvarasüsteemide tundmist ning pardaarvutiga ühendamist, näiteks klaviatuuri ja ekraaniga.

3.3 Nõuded tööle

3.3.1 Funktsionaalsed nõuded

- Roboti juhtimine käib juhtimispuldi kaudu;
- Võimalik on juhtida roboti baasi liikumist;
- Manipulaatoril on võimalik liigutada iga lüli eraldi kui ka ainult viimase lüli positsiooni, vastavalt millele valib ülejäänud lülide positsioonid tarkvara;
- Manipulaatori liikumise trajektoori on võimalik salvestada ning taasesitada;
- Eespool kirjeldatud funktsionaalsus käivitub automaatselt, kui robot sisse lülitada, st kasutaja ei pea ühendama roboti külge klaviatuuri ega ekraani, et roboti juhtimise tarkvaralahendust käivitada.

3.3.2 Muud nõuded

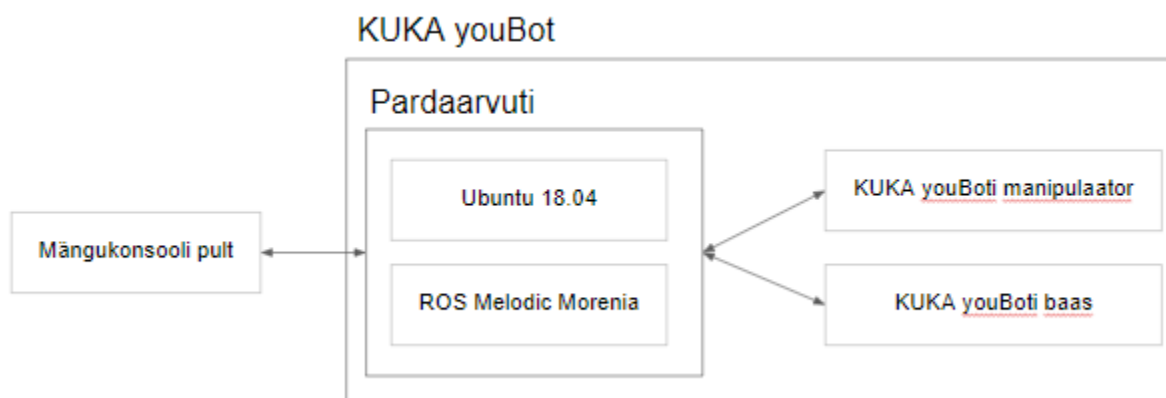
- Lahendust demonstreeritakse mobiilsel manipulaatoril KUKA youBot;
- Arendus ja lahendus toimib vahevaral ROS Melodic Morenia ja operatsioonisüsteemil Ubuntu 18.04;
- Kogu loodav lähtekood tehakse avatuna ja avaldatakse GitHub keskkonnas Apache 2.0 litsentsiga.

4 Töö lahendus

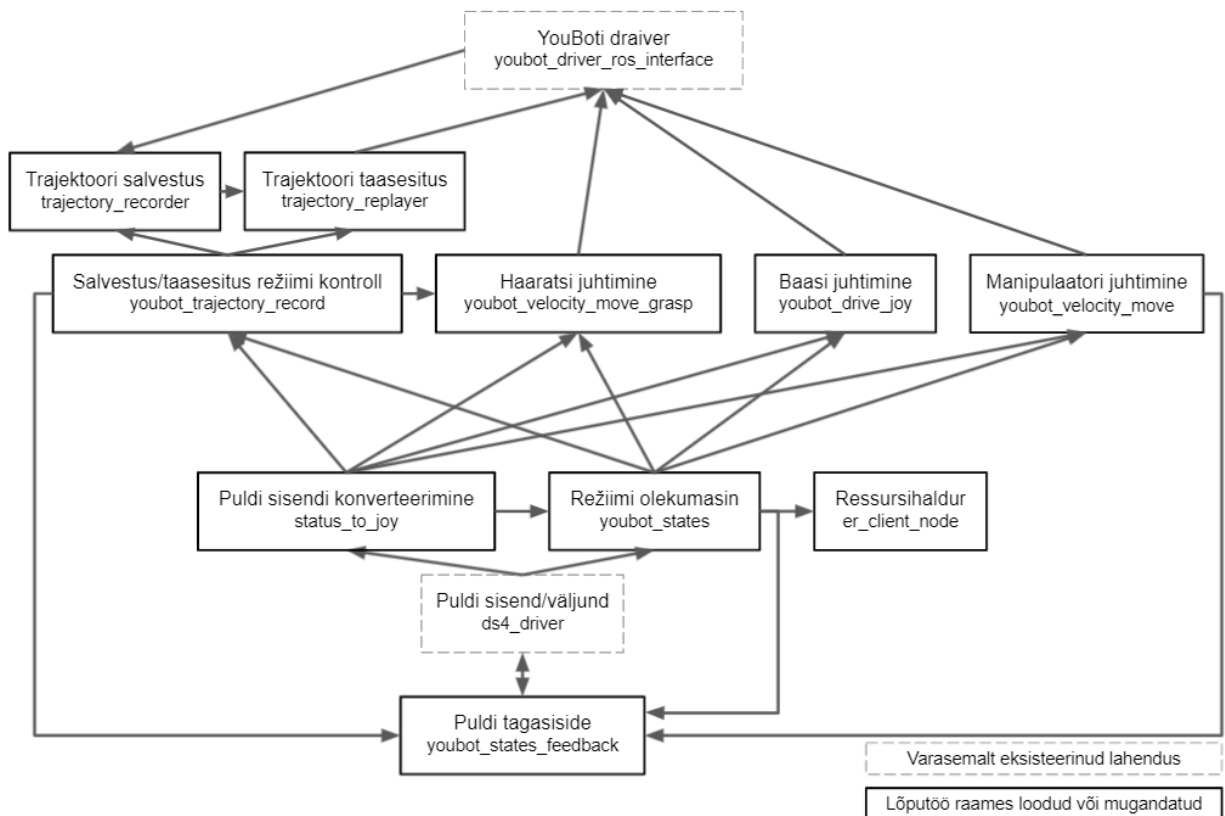
4.1 Arhitektuuri ülevaade

Valminud lahendus nimega youbot_ux [35] koosneb sisendseadmeks olevast DualShock 4 [28] mängukonsooli puldist, pardaarvutist ning KUKA youBoti baasist ja manipulaatorist (joonis 5). Konsoolipuldilt saadetakse pardaarvutile kasutajapoolne sisend ning vajadusel saadetakse puldile ka tagasisidet LEDi värvuse seadmisega või vibratsiooniga. Pardaarvutilt saadetakse youBotile edasi väljundinfo roboti liikumise kohta ning saadetakse tagasi roboti hetkeseis.

Loodud tarkvaraline lahendus on valminud operatsioonisüsteemil Ubuntu 18.04, millele on installeeritud ROSi versioon Melodic Morenia. Seda on kasutatud youBoti pardaarvutil, millel on i5-4250U protsessor ja 4 GB DDR3 mälu [36].



Joonis 5. Üldine ülevaade süsteemist, sisendseadmest robotini



Joonis 6. Ülevaade lahenduse ROSi sõlmedest

Joonisel 6 on näha lahenduses olevaid sõlmesid ning nende otstarbed. Varasemalt eksisteerinud lahenduste sõlmede korral ei ole nende tarkvaralist poolt enne kasutamist muudetud. Lõputöö raames loodud või mugandatud sõlmede korral on osaliselt kasutatud olemasolevat tarkvara või on tarkvara loodud täielikult uuena.

Ühendus roboti tarkvara (ROS) ja riistvara (manipulaatori lülid, baasi mootorid) vahel käib sõlme *youbot_driver_ros_interface* vahendusel. Sellele edastatakse soovitud mootori kiirused ja roboti lülide positsioonid. Sõlmelt on võimalik saada infot näiteks lülide hetkepositsioonide kohta, mida võtab lahenduses vastu *trajectory_recorder* sõlm. Sõlmede kirjeldused on leitavad tabelis 1.

Sõlm	Kirjeldus
<i>youbot_driver_ros_interface</i>	Ühendus tarkvara (ROS) ja roboti riistvara vahel. Lahenduses võtab vastu ja kuulutab manipulaatori, haaratsi ja baasi kiiruseid ning

	liikumise positsioone.
<i>trajectory_recorder</i>	Manipulaatori trajektoori salvestamine
<i>trajectory_replayer</i>	Manipulaatori trajektoori taasesitamine
<i>youbot_trajectory_record</i>	Manipulaatori trajektoori salvestamise, taasesitamise ja tavaoleku režiimi vahel valimine
<i>youbot_velocity_move</i>	Manipulaatori juhtimine
<i>youbot_velocity_move_grasp</i>	Manipulaatori haaratsi juhtimine
<i>youbot_drive_joy</i>	Baasi juhtimine
<i>youbot_states</i>	Olekumasin, kus käib üldine režiimi valik ning kuulutamine. Režiimideks on: 1. baasi juhtimine 2. manipulaatori ja haaratsi juhtimine 3. trajektoori salvestus ja taasesitus 4. ohutu
<i>er_client_node</i>	Ressursihaldur, millega lülitatakse režiimi tööks mõeldud sõlmesid sisse- ja väljalülitavateks sõlmedeks on: 1. <i>youbot_drive_joy</i> 2. <i>youbot_trajectory_record</i> 3. <i>youbot_velocity_move</i> 4. <i>youbot_velocity_move_grasp</i> 5. <i>trajectory_replayer</i>
<i>status_to_joy</i>	<i>Ds4_driver</i> sõlme <i>/status</i> teema kaudu kuulutatud puldi nuppude ja juhtkangide väärtuste konverteerimine <i>sensor_msgs/Joy</i> sõnumiks edasiste sõlmedele
<i>ds4_driver</i>	Juhtimispuldilt sisendi saamiseks ning väljundi saatmiseks
<i>youbot_states_feedback</i>	<i>Ds4_driver</i> sõlmele tagasiside sõnumite loomine (LEDi värvus, vibratsioon)

Tabel 1. Sõlmede kirjeldused

Sõlmed *youbot_velocity_move*, *youbot_velocity_move_grasp*, *youbot_drive_joy*, *youbot_trajectory_record* ja *er_client_node* saavad sõnumeid hetkeoleku ehk režiimi kohta sõlmelt *youbot_states*, mille järgi lülitatakse neid sisse ja välja ning võimaldatakse teha neile ettenähtud tööd. Lisaks saadetakse hetkeolek ka *youbot_states_feedback* sõlmele (joonis 6).

Trajectory_recorder ja *trajectory_replayer* sõlmesid vahendab sõlm *youbot_trajectory_record*, mille abil saab kasutaja teenuste kaudu valida manipulaatori liikumise trajektoori salvestamise ja ettemängimise vahel (joonis 6).

Kui youBoti pardaarvuti pole eelnevalt üles seatud, leiab selleks juhendi *GitHubi* *youbot_ux* lahendusest [35].

4.2 Süsteemi kasutamine

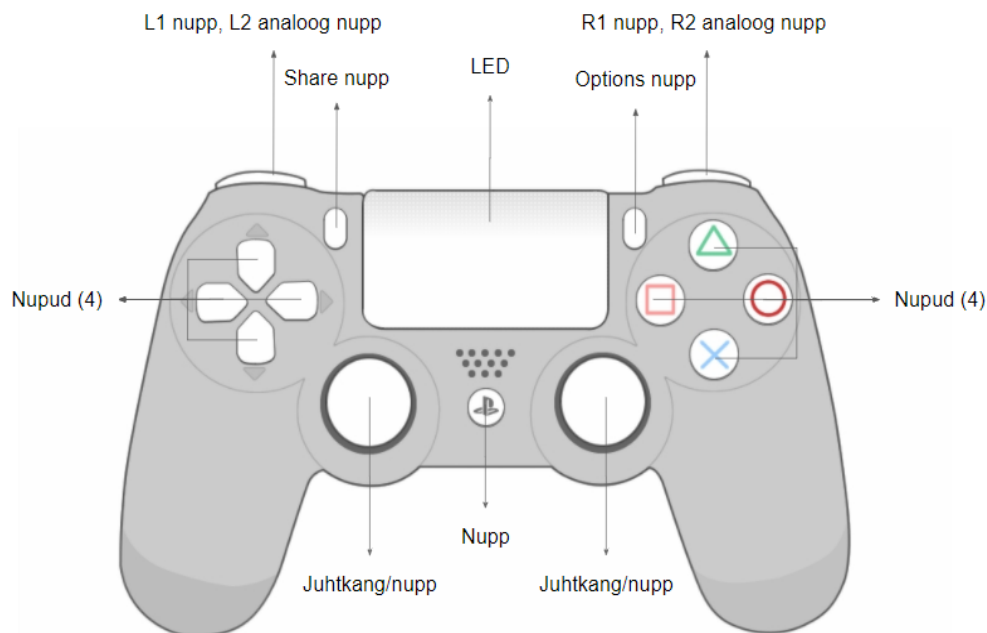
Lahendus töötab operatsioonisüsteemi teenusena, mis käivitub automaatselt pardaarvuti sisselülitamisel. Enne pardaarvuti käivitamist lülitatakse sisse roboti manipulaatori ja baasi mootorid. Puldi sisse lülitamisel ühendatakse konsoolipult bluetoothi vahendusel pardaarvutiga. Kui pult on pardaarvutiga ühenduse saanud, lõpeb konsoolipuldil oleva LEDi vilkumine ning jääb ühtlaselt siniselt põlema. Kui käivitatud on ka eelnevalt mainitud teenus, põleb konsoolipuldil LED siniselt, näidates, et robot on nn. ohutus režiimis, mille korral robotit puldiga liigutada ei saa.

Kasutades konsoolipuldil olevat ringi nuppu, saab kasutaja valida erinevate režiimide vahel:

- Baasi juhtimine;
- Manipulaatori juhtimine;
- Õpetamine;
- Ohutu režiim.

Režiimi vahetamine koos erinevate režiimide juhtimise konfiguratsioonidega on leitavad joonistel 8 (sõitmise režiim), 9 (manipulaatori juhtimise režiim), 10 (manipulaatori õpetamise režiim) ja 11 (ohutu režiim).

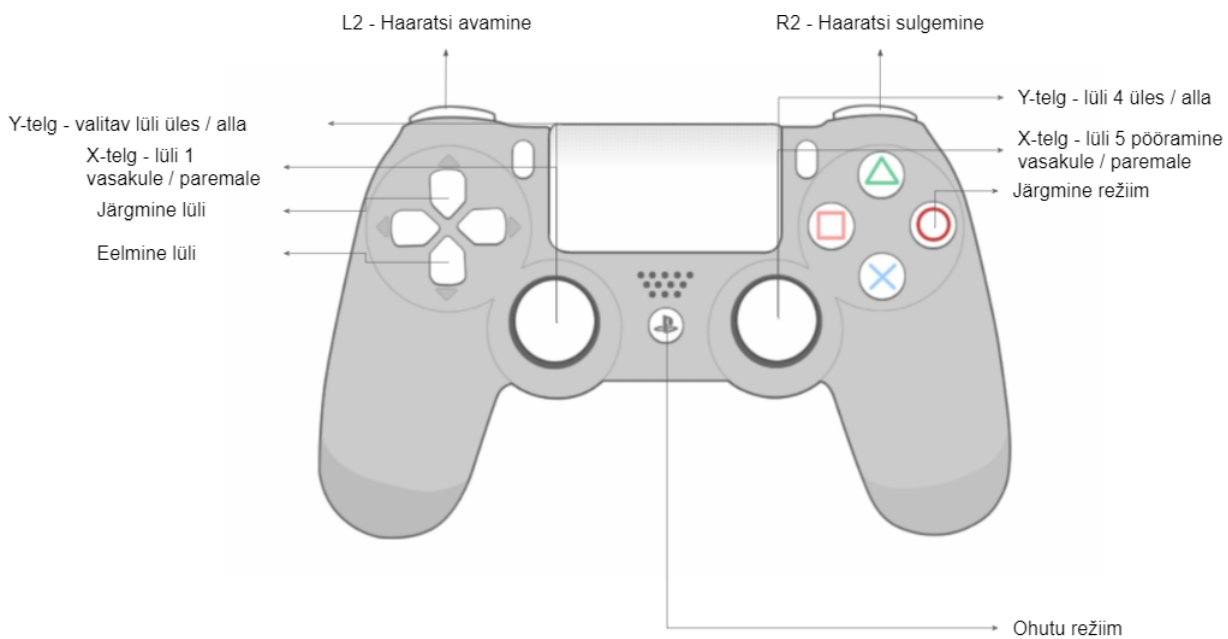
Juhtimiseks kasutatavate puldi nuppude ja juhtkangide funktsioonid sai valitud vastavalt sellele, kas kasutamisel on vaja analoogsisendit või digitaalsisendit. Baasi, manipulaatori ja haaratsi liikumisel kasutatakse analoog-juhtkange täpsemaks kiiruse kontrollimiseks. Seega neid on kasutatud juhtudel, kus täpsemat sisendi väärtust ei vajata, nagu näiteks režiimi valik või juhitava lüli valimine. Täpsemalt on valitud nupp ohutusse režiimi minekuks ning nupud ROSi sõlmede sulgemiseks, mis asuvad kas puldi horisontaalteljel keskel või ka eraldatult teistest puldil olevatest nuppudest.



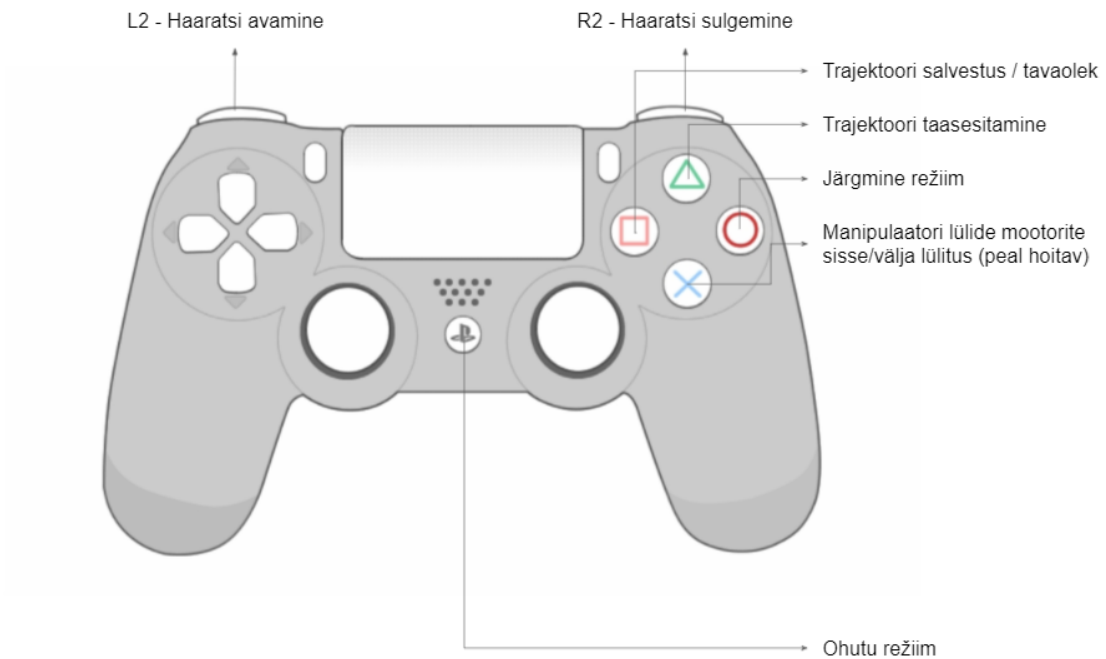
Joonis 7. *Kontrolleri nuppude ja juhtkangide asetused*



Joonis 8. *Kontrolleri konfiguratsioon roboti sõitmise režiimis*



Joonis 9. *Kontrolleri konfiguratsioon manipulaatori lüli haaval juhtimise režiimis*



Joonis 10. *Kontrolleri konfiguratsioon õpetamise režiimis*



Joonis 11. *Kontrolleri konfiguratsioon ohutu režiimis*

Vastavalt režiimile ning režiimis tehtud valikutele antakse kasutajale tagasiside hetkesituatsiooni kohta puldi vibratsioonide ja LEDi värvi kaudu vastavalt tabelile 2.

Režiim	LED	Vibratsioon
Baasi juhtimine	Roheline	-
Manipulaatori juhtimine	Kollane	Lüli 2 valik: 0.1s Lüli 3 valik: 0
Manipulaatori õpetamine	Helesinine: tavaolek Punane: trajektoori salvestus Lilla: trajektoori taasesitus	-
Ohutu	Sinine	-

Tabel 2. Puldi LEDi ja vibratsiooni tagasiside režiimi valikul

4.3 Süsteemi ülesehitus

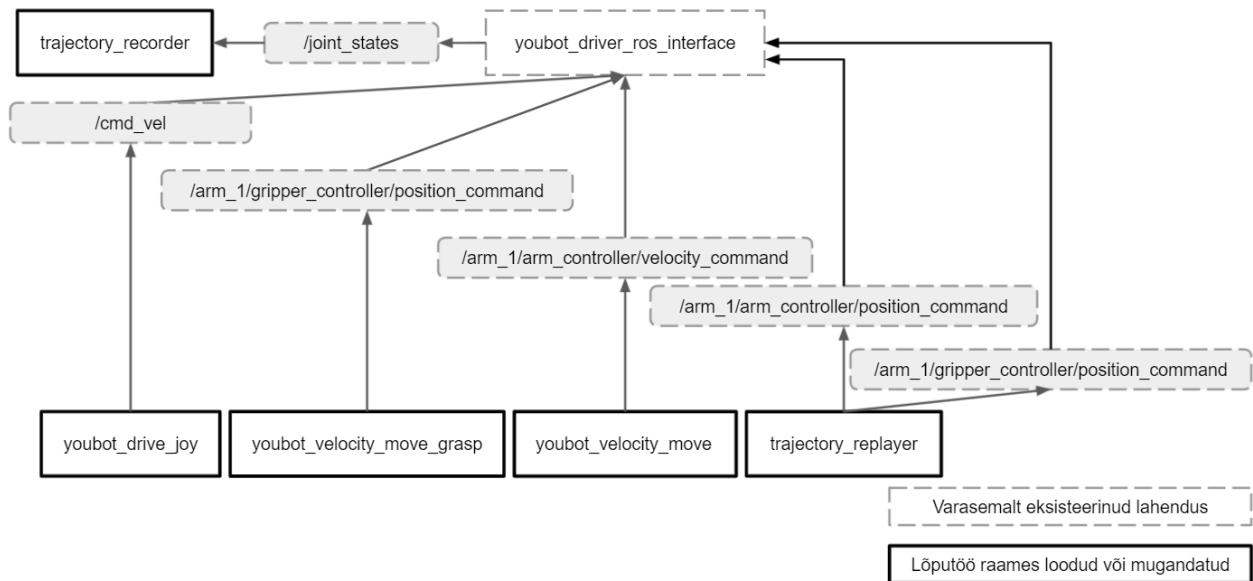
ROS komponentidest koosneva lahenduse ülevaade on nähtav joonisel 6.

4.3.1 YouBoti draiveri sõlme ühendused

Ühendus ROS tarkvara ja roboti riistvaraga, nagu roboti manipulaatori ja roboti baasiga käib sõlme *youbot_driver_ros_interface* vahendusel. Draiverina kasutati youBoti draiverit [37] ning ROSi youboti liidest [38]. Sõlm *youbot_driver_ros_interface* võtab vastu baasi ja manipulaatori mootorite kiiruseid ning roboti lülide positsioonide väärtusi (joonis 12). Selle sõlme kaudu kuulutatakse infot lülide hetkeolekute kohta (kiirused, positsioonid) (joonis 12). Selle sõlmega suhtlevad järgnevad loodud sõlmed:

- **Trajectory_replayer** - võtab *youbot_driver_ros_interface* sõlmelt vastu manipulaatori lülide lõpp-positsioone */arm_1/arm_controller/position_command* rubriigilt ning haaratsi positsioone */arm_1/gripper_controller/position_command* rubriigilt.

- **Youbot_velocity_move_grasp** - saadab *youbot_driver_ros_interface* sõlmele manipulaatori haaratsi lõpp-positsioone */arm_1/gripper_controller/position_command* rubriigi kaudu.
- **Youbot_velocity_move** - saadab *youbot_driver_ros_interface* sõlmele manipulaatori lülide liikumise kiirusi */arm_1/arm_controller/velocity_command* rubriigi kaudu.
- **Youbot_drive_joy** - saadab *youbot_driver_ros_interface* sõlmele roboti baasi rataste liikumise kiirusi *cmd_vel* rubriigi kaudu.
- **Trajectory_recorder** - võtab *youbot_driver_ros_interface* sõlmelt vastu roboti lülide positsioone */joint_states* rubriigilt.



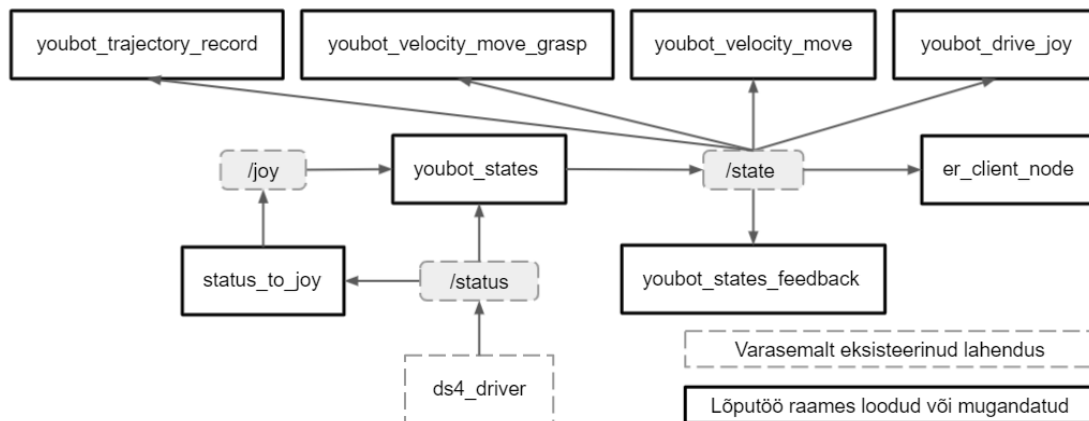
Joonis 12. *Youbot_driver_ros_interface* sõlme ühendused

4.3.2 Olekumasin ja ressursihaldur

Olekumasinaks on ROSi Smach [39] paki põhjal töötav sõlm *Youbot_states*, mis seab vastavalt kasutaja sisendile paika, millises režiimis robot hetkel on. Sõlme sisenditeks on *status_to_joy* sõlme */joy* rubriigi kaudu kuulutatav sõnum *sensor_msgs/Joy* [40] puldi nuppude ja juhtkangide sisendi väärtustega ning *ds4_driver* sõlme */status* rubriigi kaudu kuulutatavad *ds4_driver/Status*

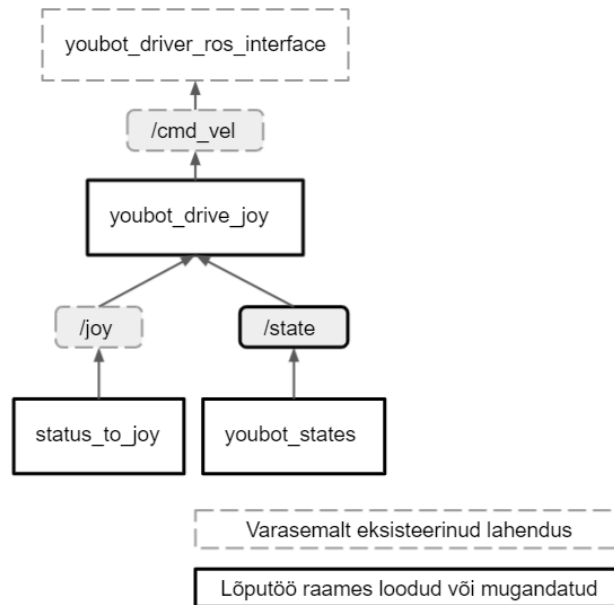
sõnumid. Väljundiks on olekumasina sõlme poolt režiimi kohta käiv sõne, mis kuulutatakse */state* rubriigi kaudu edasistele sõlmedele (joonis 13).

Olekumasina poolt valitud režiimi põhjal lülitatakse lahenduses kasutatava TeMoto [41] ressursihalduriga sisse või välja selle jaoks vajalikud sõlmed, säästes pardaarvuti ressursse. Lisaks saab välja lülitada režiimiks mittevajalikke sõlmi eesmärgil, et kasutaja poolt saadetak sisend ei jõuaks mõne muu režiimi sõlme väljundiks, mille tõttu võiks robot liikuda mittesoovitud viisil. Näiteks, kui töötavad roboti baasi ning manipulaatori juhtimise sõlmed, ning mõlemat kontrollitakse samade kontrolleri sisenditega, võib mõlema korraga töötamisel liigutada nii baasi kui ka manipulaatori lüli. Kui soovitakse juhtida ainult roboti baasi, saab konflikti eemaldamiseks manipulaatori sõlme välja lülitada. Lahenduses tegeleb sellega sõlm *er_client_node*, mis vastavalt *youbot_states* sõlme */state* rubriigis kuulutatavale režiimi sõnele käivitab ning lülitab välja sõlmesid *youbot_velocity_move_grasp*, *youbot_drive_joy*, *youbot_velocity_move*, *youbot_trajectory_record* ning *trajectory_replayer* (joonis 13).



Joonis 13. *Youbot_states* olekumasina sõlme ühendused

4.3.3 KUKA youBoti baasi juhtimine

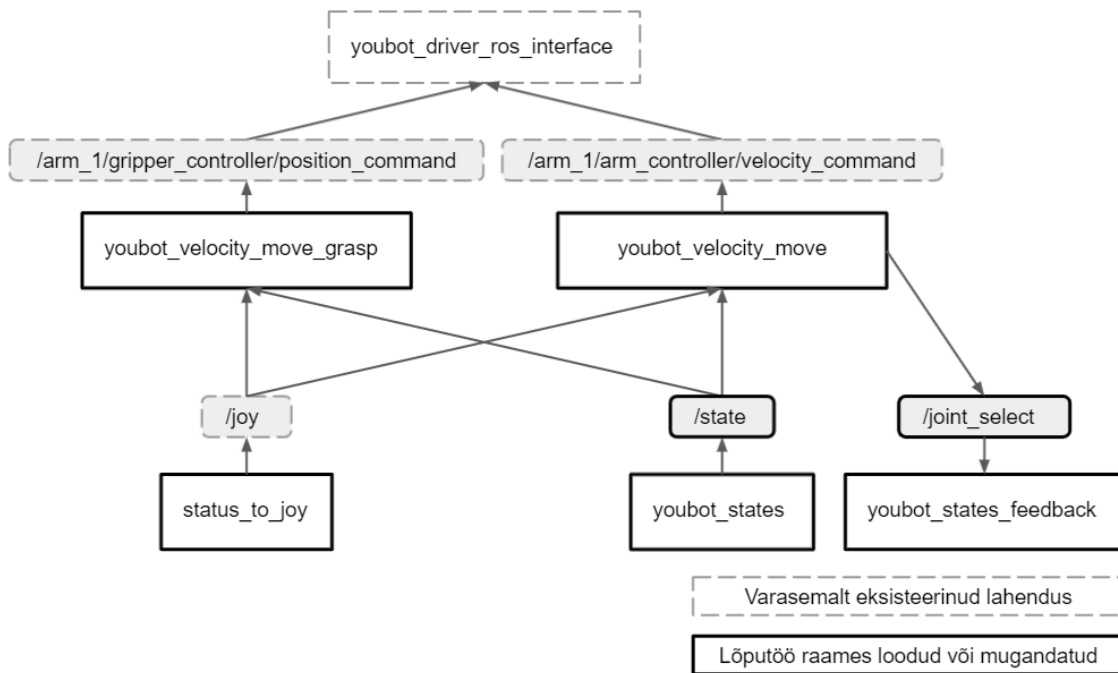


Joonis 14. *Youbot_drive_joy sõlme ühendused*

Roboti baasi mootorite kiiruste seadmiseks on vaja konsoolipuldi nuppude ja juhtkangide *sensor_msgs/Joy* [42] sõnumid konverteerida *geometry_msgs/Twist* [43] sõnumiks. Seejärel saadetakse need `/cmd_vel` rubriigi kaudu youBoti draiverile (joonis 14). Vastavalt olekumasina *youbot_states* poolt määratud režiimile seatakse paika, kas *youbot_drive_joy* sõlm töötab ning kuulutatab *youbot_driver_ros_interface* sõlme jaoks liikumist määravaid sõnumeid. Kui *youbot_drive_joy* töötab, seatakse mootori kiirused vastavalt puldi analoogjuhtkangi asendile väärtusega 0.0 kuni 1.0, mis korrutatakse konfiguratsioonis seatud kordaja väärtusega 0.0 kuni 1.0 maksimaalse võimaliku kiiruse limiteerimiseks.

4.3.4 KUKA youBoti manipulaatori juhtimine

4.3.4.1 Manipulaatori lüli haaval liigutamine



Joonis 15. *Youbot_velocity_move* ja *youbot_velocity_move_grasp* sõlme ühendused

Manipulaatori juhtimine käib kahe sõlme kaudu: *youbot_velocity_move_grasp* ning *youbot_velocity_move* (joonis 15).

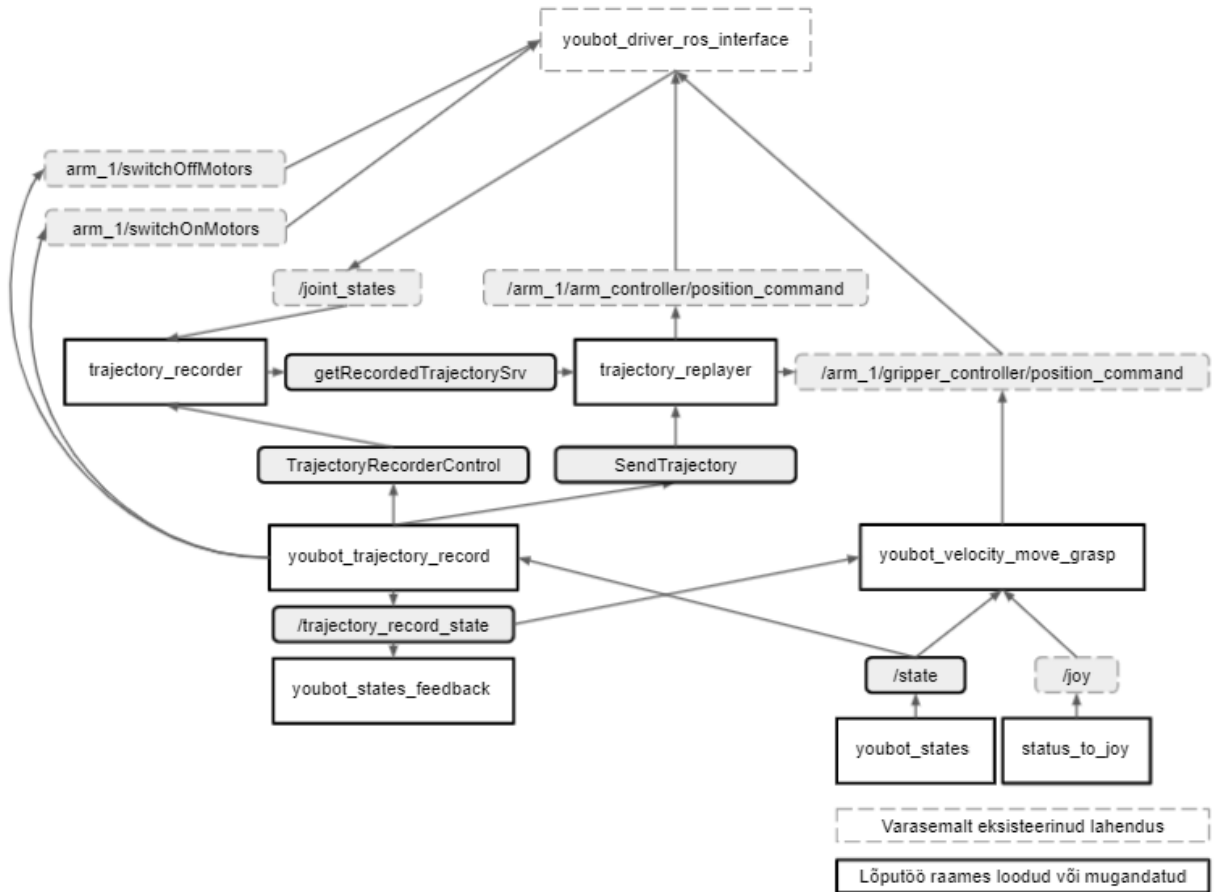
KUKA youBoti korral on võimalik manipulaatori haaratsile saata vaid soovitud haaratsi lüli positsioone, mitte nende kiirusi, seega saadetakse need *brics_actuator/JointPositions* [44] sõnumina *youbot_driver_ros_interface* sõlmele */arm_1/gripper_controller/position_command* rubriigi kaudu.

Manipulaatori lülide kiiruste infot saadetakse *brics_actuator/JointVelocities* [45] sõnumiga *youbot_velocity_move* sõlmelt *youbot_driver_ros_interface* sõlmele */arm_1/arm_controller/velocity_command* rubriigi kaudu.

Youbot_velocity_move sõlmest saadetakse */joint_select* rubriigi kaudu täisarv tüüpi sõnumeid kasutaja poolt valitud lüli määramiseks *youbot_states_feedback* sõlmes, millele vastavalt teavitatakse kasutajat valitud lülist.

Kasutajapoolne sisend saadakse *status_to_joy* sõlmelt */joy* rubriigilt. Seda, kas kasutajapoolset sisendit lubatakse muuta manipulaatori liigutamiseks *sensor_msgs/Joy* [42] tüüpi sõnumiks youBoti draiveri jaoks, määrab *youbot_states* sõlme */state* rubriigis kuulutatav režiimi määrav sõnum.

4.3.4.2 Manipulaatori trajektoori salvestamine ja taas esitamine



Joonis 16. *Youbot_trajectory_record*, *trajectory_recorder* ja *trajectory_replayer* sõlme ühendused

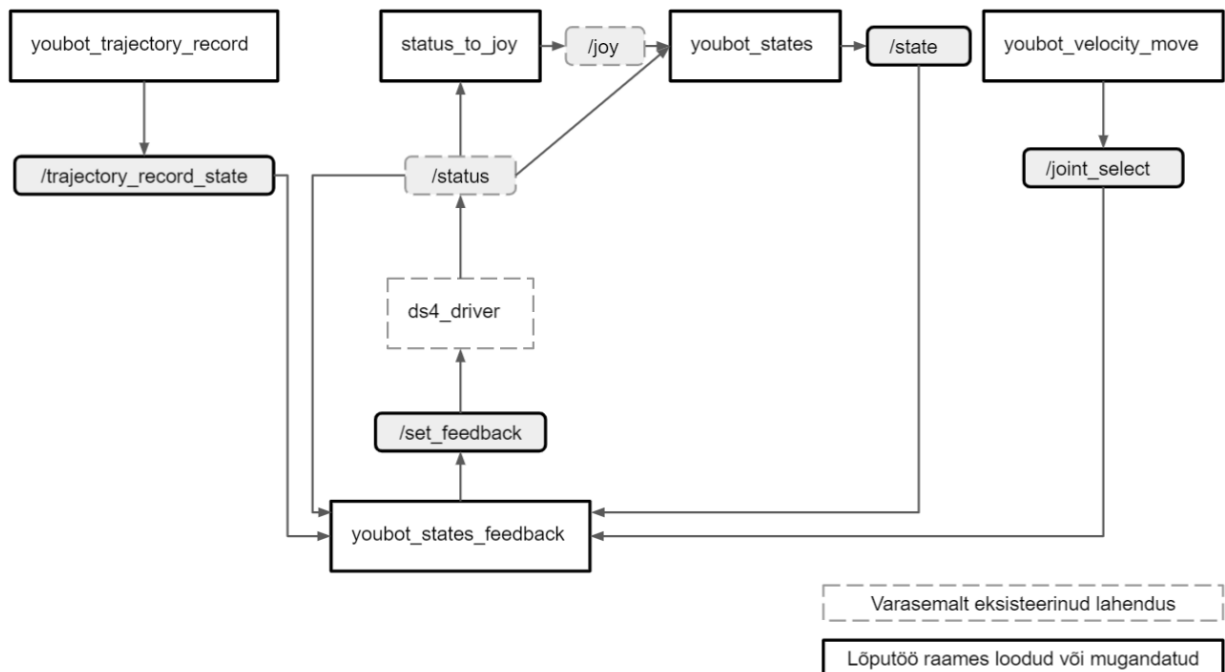
Trajektooride salvestamiseks rakendatakse *trajectory_recorder* [46] ja taasesitamiseks *youbot_manual_operation* [47] tarkvarapaketti.

Trajectory_recorder sõlmes salvestatakse iga 0.05 sekundi tagant */joint_states* rubriigilt *sensor_msgs/JointState* [48] sõnumitena saadavad roboti lülide positsioonide ja kiiruste väärtused. Selleks, et manipulaatori lülisid salvestamise käigus käsitsi liigutada, on vaja eemaldada mootorite takistus. Manipulaatori lülide välja ja sisse lülitamiseks saab kasutada konsoolipuldil olevat ringi nuppu (joonis 11), mis kutsub vastavalt teenuseid *arm_1/switchOffMotors* ja *arm_1/switchOnMotors*. Samuti ei ole käsitsi võimalik liigutada ka manipulaatori haaratsi lülisid, mis on seetõttu lahenduses kontrollitavad konsoolipuldiga ning *youbot_velocity_move_grasp* sõlme abil.

Salvestatud trajektoori taasesitamiseks kasutatakse *trajectory_replayer* sõlme. See sõlm saab *getRecordedTrajectory* teenuse kaudu info salvestatud lülide positsioonide ning kiiruste kohta *trajectory_recorder* sõlmelt. Saadud positsioonide taasesitamiseks saadetakse *youbot_driver_ros_interface* sõlmele uued manipulaatori ja haaratsi lülide lõpp-positsioonid iga 0.1 sekundi järel, vastavalt salvestatud positsioonide järjekorrale. Saadetavate sõnumite tüübiks on *brics_actuator/JointPositions*. Manipulaatori lülide positsioonid saadetakse */arm_1/arm_controller/position_command* ning manipulaatori haaratsi positsioonid */arm_1/gripper_controller/position_command* rubriigi kaudu (joonis 16).

Trajectory_recorder ja *trajectory_replayer* sõlmede salvestamise, taasesitamise ja neutraalse oleku režiimi kontrollib sõlm *youbot_trajectory_record*. Trajektoori salvestamiseks ja salvestamise lõpetamiseks kutsutakse *youbot_trajectory_record* sõlme kaudu teenust *TrajectoryRecorderControl* ning taasesitamiseks *SendTrajectory* teenust. Konsoolipuldil režiimi valiku näitamiseks kuulutatakse selle valikut */trajectory_states_feedback* rubriigis. Seda infot kasutab *youbot_states_feedback* sõlm konsoolipuldil LEDi värvi määramiseks ning *youbot_velocity_move_grasp* manipulaatori liigutamise võimaluse seadmiseks, mida on võimalik teha vaid trajektoori salvestamisel ning tavaolekus.

4.3.5 Kontrolleri kasutus ja tagasiside



Joonis 17. *Ds4_driver ja youbot_states_feedback sõlme ühendused*

Kasutaja konsoolipuldi sisendi võtab vastu sõlm *ds4_driver*, mis kuulutab */status* rubriigi kaudu *ds4_driver/Status* sõnumeid puldi aku staatuse, nuppude ja juhtkangide olekust. Rubriigist */status* loeb aku infot *youbot_states* ja nuppude ning juhtkangide väärtuste infot *status_to_jog* sõlm. (joonis 17).

Kuna loodud lahenduses kasutatakse nuppude ja juhtkangide staatuse lugemiseks ROSi joy paketi [40] */joy* rubriiki, kasutatakse sõlme *status_to_jog* selleks, et *ds4_driver* sõlme väljundit muuta */joy* rubriigi jaoks Joy sõnumi vormi. Joy sõnumis on info analoogkangide ning nuppude oleku väärtustest.

Sõlme *ds4_driver* rakendatakse ka kasutaja puldile tagasiside saatmiseks. Tagasiside sõnum luuakse *youbot_states_feedback* sõlmes ning kuulutatakse *ds4_driver* sõlme jaoks *ds4_driver/Feedback* sõnumina rubriigi */set_feedback* kaudu. Infot saadetakse puldi mootorite ehk vibratsiooni kiiruse väärtuse seadmiseks ja puldi LEDi värvi kohta. Infot kuulutatakse

sõlmele *youbot_states_feedback* sõlmest *youbot_velocity_move /joint_select* rubriigiga *youbot_trajectory_move* ning sõlmest *youbot_states /state* rubriigiga (joonis 17).

5 Tulemused

Töö tulemusena on kasutajal võimalik puldi kaudu juhtida KUKA youBoti baasi ja manipulaatorit ning kasutada manipulaatorit trajektoori õpetamise ja taasesitamise viisil. Lahendus loodi operatsioonisüsteemil Ubuntu 18.04 ning vahevaraga ROS Melodic Morenia. Roboti juhtimiseks kasutatakse DualShock 4 [28] juhtmevaba kontrolleri.

YouBoti baasi juhitakse puldi abil vastavalt joonisel 8 näidatud konfiguratsioonile. Vastavalt puldiga antavast sisendist seatakse igale roboti baasi mootorile kiirused, mis saadakse loodud lahenduse sõlmest *youbot_drive_joy*. Kasutajale antakse baasi juhtimise režiimi valikust teada konsoolipuldil oleva LEDi värviga.

Roboti manipulaatori kontrollimine käib lahenduses kahe sõlme abil: *youbot_velocity_move* ja *youbot_velocity_move_grasp*. *Youbot_velocity_move* sõlm saadab edasi manipulaatori lülide kiiruseid ning *youbot_velocity_move_grasp* sõlm manipulaatori haaratsile soovitud positsioone selle avamiseks ja sulgemiseks. Mõlema korral saadakse kasutajapoolset sisendinfot kasutatavast puldist vastavalt joonisel 9 näidatud konfiguratsioonile. Baasi juhtimise režiimist annab teada konsoolipuldi LEDi värv ning manipulaatori lüli valikust vibratsioon.

Integreeritud õpetamise režiimiga on võimalik salvestada manipulaatori ja haaratsi positsioone ning liikumise kiiruseid *trajectory_recorder* sõlmega. Manipulaatori lülide ning haaratsi positsioone saab taasesitada *trajectory_replayer* sõlmega. Õpetamise režiimi kontrollimine käib *youbot_trajectory_record* abil, mille kaudu seatakse paika, kas trajektoori salvestatakse, salvestatud trajektoori taasesitatakse või ollakse tegevusetu. Kasutajapoolne juhtimine käib puldiga vastavalt joonisele 10. Salvestamise korral liigutatakse manipulaatorit käsitsi ning haaratsi lülisid puldi kaudu. Selleks, et manipulaatorit saaks käsitsi liigutada, saab kasutatava puldi sisendiga mootoreid lülitada sisse ja välja. Kuigi õpetamise režiim on võimeline trajektoori taasesitama, ei ole seda võimalik kogu manipulaatori liikumisulatuses teha. Seda seetõttu, et youBoti draiveri poolt seatud tarkvaralised limiidid võimalikele manipulaatori positsioonidele on väiksemaid, kui füüsiliselt oleks neid võimalik liigutada. Seega manipulaatori positsioone, mis on roboti draiveri tarkvaras seatud limiitidest väljas, ei taasesitata.

Töö eesmärgina on välja toodud manipulaatori viimase manipulaatori lüli juhtimine, mille korral ülejäänud lülide positsioonid seab paika tarkvara. Loodud lahenduses selline võimalus puudub manipulaatori liikumisastmete vähesuse tõttu. KUKA youBoti manipulaatoril on olemas 5 liikuvusastet, kuid nende liikumise suunast tulenevalt ei ole manipulaatorit igal teljel võimalik sirgjooneliselt liigutada. Võimaliku lahendusena saaks siduda roboti manipulaatori ja baasi liikumise. Roboti baasi liikumine lisaks ühe liikuvusastme, mis lubaks manipulaatori viimast lüli sirgjooneliselt vasakule ja paremale liigutada. Sellise lahenduse korral oleks vaja baasi mootorite ja manipulaatori lülide juhtimine kokku integreerida, võttes arvesse youBoti ROSi draiveri poolt pakutavaid rubriike roboti manipulaatori ja baasi juhtimiseks (positsiooni või kiiruste saatmise võimalust).

Loodud lahendus nimega youbot_ux on leitav Robotics at IMS Lab GitHub keskkonnas Apache 2.0 [35].

Kokkuvõte

Töö eesmärgiks oli luua võimalus KUKA youBot roboti baasi ja manipulaatori kontrollimiseks, mille juhtimine käib konsoolipuldi kaudu ning mis ei vaja kasutajalt teadmisi roboti tarkvaralisest süsteemist. Töö lahendus loodi operatsioonisüsteemil Ubuntu 18.04 ning ROS versioonil Melodic Morenia. Praktilise töö tulemusena loodi ROSi kimp youbot_ux [35], mille abil saab roboti baasi ja manipulaatorit juhtida DualShock 4 [28] mängukonsooli kontrolliga.

Kontrollerit saab kasutada roboti baasi ja manipulaatori juhtimiseks, mille kasutamine on välja toodud peatükis 4.2. Kontrollerit kasutades saab valida 4 režiimi vahel: roboti baasi juhtimine, lüli haaval manipulaatori juhtimine, õpetamise ehk manipulaatori trajektoori salvestamise ja taasesitamise režiim ning ohutu režiim. Nende režiimide tarkvaralised ülesehitused on leitavad peatükis 4.3.3, 4.3.4 ja 4.3.5.

Režiimide jaoks loodud komponentide laadimine ja mahalaadimine viiakse läbi TeMoto poolt pakutava ROS ressursihalduse abil. Seda kasutatakse pardaarvuti ressursside kokkuhoiuks ning komponentide sisendite konfliktist hoidumiseks, kus kasutaja sisend ühtib, kuid väljund ehk roboti liikumine on erinev ning ei ole üheaegselt soovitud. Režiimide valik viiakse tarkvaras läbi ROSi Smach paki baasil töötava olekumasinaga. Nende ülesehitus lahenduses on toodud välja peatükis 4.3.2.

Viited

- [1] Builtin, „What Is Robotics? Types Of Robots | Built In“. <https://builtin.com/robotics> (vaadatud märts 15, 2021).
- [2] D. Anta ja K. Bejczy, „TELEOPERATION, TELEROBOTICS“, lk 1–2, 14.
- [3] „youBot Store“, *youBot Store*. <http://www.youbot-store.com/> (vaadatud mai 01, 2021).
- [4] J. Walker, „A Guide to Autonomous Robots & 8 AMR Applications“, *Waypoint Robotics*, jaan 11, 2017. <https://waypointrobotics.com/blog/what-autonomous-robots/> (vaadatud märts 16, 2021).
- [5] C. Woodford, „Roomba robot vacuum cleaners“, *Explain that Stuff*, apr 07, 2020. <http://www.explainthatstuff.com/how-roomba-works.html> (vaadatud märts 29, 2021).
- [6] J. Pedersen, „The 5 Key Abilities of Mobile Manipulation Systems“, *Robotics Business Review*, jaan 04, 2021. <https://www.roboticsbusinessreview.com/opinion/the-5-key-abilities-of-mobile-manipulation-systems/> (vaadatud apr 07, 2021).
- [7] „Robot Manipulator - an overview | ScienceDirect Topics“. <https://www.sciencedirect.com/topics/engineering/robot-manipulator> (vaadatud mai 02, 2021).
- [8] „KUKA youBot, robot mobile omni-directionel avec bras“, *Génération Robots*. <https://www.generationrobots.com/en/402093-kuka-youbot-robot-mobile-omni-directionel-avec-bras.html> (vaadatud apr 05, 2021).
- [9] „Curiosity Rover Isolated stock illustration. Illustration of exploration - 129211364“, *Dreamstime*. <https://www.dreamstime.com/curiosity-rover-isolated-white-background-d-render-curiosity-rover-isolated-image129211364> (vaadatud mai 01, 2021).
- [10] „Digital Vanguard ROV“, *EOD Technology*. <https://www.eod-technology.com/catalog/medeng/digital-vanguard-rov/> (vaadatud mai 01, 2021).
- [11] „Arm | Rover – NASA’s Mars Exploration Program“. <https://mars.nasa.gov/msl/spacecraft/rover/arm/> (vaadatud mai 01, 2021).
- [12] R. Valner, K. Kruusamäe, ja M. Pryor, „TeMoto: Intuitive Multi-Range Telerobotic System with Natural Gestural and Verbal Instruction Interface“, *Robotics*, kd 7, nr 1, Art. nr 1, märts 2018, doi: 10.3390/robotics7010009.
- [13] „Glossary of Robotics Terms | Robotics Definitions & Examples“. <https://www.motoman.com/en-us/about/company/robotics-glossary> (vaadatud mai 19, 2021).

- [14] J. Nadvornik ja P. Smutný, „Remote control robot using Android mobile device“, mai 2014, lk 373–378. doi: 10.1109/CarpathianCC.2014.6843630.
- [15] M. Wagner, D. Avdic, ja P. Heß, „Gamepad Control for Industrial Robots - New Ideas for the Improvement of Existing Control Devices“, *Proceedings of the 13th International Conference on Informatics in Control, Automation and Robotics*, Lisbon, Portugal, 2016, lk 368–373. doi: 10.5220/0005982703680373.
- [16] „What are Robotic Teach Pendants?“, *Robots Done Right*.
<https://robotsdoneright.com/Articles/what-are-robotic-teach-pendants.html> (vaadatud märts 23, 2021).
- [17] „Teach Pendant“, *Comau*.
<https://www.comau.com/en/our-competences/robotics/teach-pendant-tp5> (vaadatud apr 05, 2021).
- [18] L. Tuttle ja B. Gibbs, „Long Live the Robot Teach Pendant“, *Assembly*, dets 02, 2020.
<https://www.assemblymag.com/articles/96022-long-live-the-robot-teach-pendant?v=preview> (vaadatud märts 23, 2021).
- [19] K. Dumbre, S. Ganeshkar, ja A. Dhekne, „Robotic Vehicle Control using Internet via Webpage and Keyboard“, *Int. J. Comput. Appl.*, kd 114, nr 17, lk 15–19, märts 2015, doi: 10.5120/20069-1981.
- [20] „Methods For Controlling Your Robot – Rants of an old git“, aug 25, 2015.
<http://www.corteil.co.uk/Blog/raspberry-pi/methods-for-controlling-your-robot/> (vaadatud märts 30, 2021).
- [21] GAMER2GO, „Controller vs Keyboard and Mouse“, *GAMER2GO*, aug 17, 2018.
<https://gamer2go.com/blogs/gamers2go/controller-vs-keyboard-and-mouse> (vaadatud mai 19, 2021).
- [22] Digital Office Solutions, „7 Benefits Of Touchscreen Technology - Digital Office Solutions“, *Digital Office Solutions*, märts 12, 2018.
<https://www.digitalofficesol.co.uk/2018/03/12/7-benefits-touchscreen-technology/> (vaadatud märts 24, 2021).
- [23] M. Lutovac Banduka, „Remote Monitoring and Control of Industrial Robot based on Android Device and Wi-Fi Communication“, *Autom. – J. Control Meas. Electron. Comput. Commun.*, kd 56, nr 3, lk 282, dets 2015, doi: 10.7305/automatika.2015.10.1057.
- [24] D. Szondy, „iRobot unveils one tablet-based control system to rule them all“, *New Atlas*, okt 15, 2014. <https://newatlas.com/irobot-upoint/34222/> (vaadatud apr 05, 2021).
- [25] L. Howard ja B. Heatherly, „Video Game Controllers“, lk 1,2, 2014.
- [26] K. Fulton, „Robot to the Mine Rescue“, *Energy.gov*, aug 18, 2011.
<https://www.energy.gov/articles/robot-mine-rescue> (vaadatud märts 30, 2021).
- [27] „Xbox 360 Controller for Windows | Microsoft Accessories“, *Microsoft*.

- <https://www.microsoft.com/accessories/en-ww/products/gaming/xbox-360-controller-for-windows/52a-00004> (vaadatud märts 25, 2021).
- [28] „DualSense wireless controller | The innovative new controller for PS5“, *PlayStation*. <https://www.playstation.com/en-us/accessories/dualshock-4-wireless-controller/> (vaadatud märts 25, 2021).
- [29] „Robot Control - Joystick for Differential Steering“, *ImpulseAdventure*. <https://www.impulseadventure.com/elec/robot-differential-steering.html> (vaadatud märts 25, 2021).
- [30] Kinova, *KINOVA® Remote robotic manipulation system User Guide*. [Online]. Available at: https://www.kinovarobotics.com/sites/default/files/UG-011_KINOVA%20Remote%20robotic%20manipulation%20system-User%20Guide_EN_R02%20%281%29.pdf
- [31] S. Bonaiuto, A. Cannavò, G. Piumatti, G. Paravati, ja F. Lamberti, „Tele-operation of Robot Teams: A Comparison of Gamepad-, Mobile Device and Hand Tracking-Based User Interfaces“, *2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC)*, juuli 2017, kd 2, lk 555–560. doi: 10.1109/COMPSAC.2017.278.
- [32] B. D. Kechavarzi, S. Sabanovic, ja K. Weisman, „Evaluation of control factors affecting the operator’s immersion and performance in robotic teleoperation“, *2012 IEEE RO-MAN: The 21st IEEE International Symposium on Robot and Human Interactive Communication*, Paris, France, sept 2012, lk 608–613. doi: 10.1109/ROMAN.2012.6343818.
- [33] „YouBot Detailed Specifications“, *YouBot wiki*. http://www.youbot-store.com/wiki/index.php/YouBot_Detailed_Specifications (vaadatud apr 16, 2021).
- [34] MIURA Yasuyuki, „Distributions“, *ROS Wiki*. <http://wiki.ros.org/Distributions> (vaadatud mai 14, 2021).
- [35] „Ut-ims-robotics/youbot_ux“, *GitHub*. https://github.com/ut-ims-robotics/youbot_ux (vaadatud apr 18, 2021).
- [36] K. Julle, „Roboti KUKA youBot riistvara ja ROS-tarkvara uuendamise“, Thesis, Tartu Ülikool, 2019. Vaadatud: mai 14, 2021. [Online]. Available at: <https://dspace.ut.ee/handle/10062/64279>
- [37] „Youbot/youbot_driver“, *GitHub*. https://github.com/youbot/youbot_driver (vaadatud apr 16, 2021).
- [38] „Youbot/youbot_driver_ros_interface“, *GitHub*. https://github.com/youbot/youbot_driver_ros_interface (vaadatud apr 16, 2021).
- [39] „Smach“, *ROS Wiki*. <http://wiki.ros.org/smach> (vaadatud apr 16, 2021).
- [40] „Joy“, *ROS Wiki*. <http://wiki.ros.org/joy> (vaadatud apr 28, 2021).

- [41] „TeMoto“, *GitHub*. <https://github.com/temoto-telerobotics> (vaadatud apr 15, 2021).
- [42] „Sensor_msgs/Joy Documentation“, *ROS Docs*.
http://docs.ros.org/en/api/sensor_msgs/html/msg/Joy.html (vaadatud mai 15, 2021).
- [43] „Geometry_msgs/Twist Documentation“, *ROS Docs*.
http://docs.ros.org/en/melodic/api/geometry_msgs/html/msg/Twist.html (vaadatud mai 16, 2021).
- [44] „Brics_actuator/JointPositions Documentation“, *ROS Docs*.
http://docs.ros.org/en/hydro/api/brics_actuator/html/msg/JointPositions.html (vaadatud mai 16, 2021).
- [45] „Brics_actuator/JointVelocities Documentation“, *ROS Docs*.
http://docs.ros.org/en/groovy/api/brics_actuator/html/msg/JointVelocities.html (vaadatud mai 16, 2021).
- [46] „Ut-ims-robotics-sandbox/robert_v_sandbox“, *GitHub*.
https://github.com/ut-ims-robotics-sandbox/robert_v_sandbox/tree/teach-mode-devel/trajectory_recorder (vaadatud apr 16, 2021).
- [47] „Ut-ims-robotics/youbot“, *GitHub*.
https://github.com/ut-ims-robotics/youbot/blob/c4775dff0f3957db6103da8b6a8a0e971f009179/youbot_manual_operation/src/trajectoryRecorder.cpp#L597 (vaadatud apr 16, 2021).
- [48] „Sensor_msgs/JointState Documentation“, *ROS Docs*.
http://docs.ros.org/en/melodic/api/sensor_msgs/html/msg/JointState.html (vaadatud mai 17, 2021).

Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks

Mina, Hendrik Olesk

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

"Nägemisulatuses kaugjuhitava mobiilse robotmanipulaatori kasutajamugavuse tõstmine"

mille juhendajad on Karl Kruusamäe, Robert Valner

reprodutseerimiseks eesmärgiga seda säilitada, sealhulgas lisada digitaalarhiivi DSpace kuni autoriõiguse kehtivuse lõppemiseni.

2. Annan Tartu Ülikoolile loa teha punktis 1 nimetatud teos üldsusele kättesaadavaks Tartu Ülikooli veebikeskkonna, sealhulgas digitaalarhiivi DSpace kaudu Creative Commons'i litsentsiga CC BY NC ND 3.0, 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.

Hendrik Olesk
20.05.2021