



Kauno technologijos universitetas

Elektros ir elektronikos fakultetas

ABB roboto manipulatoriaus valdymo, naudojant virtualios realybės akinius, galimybių tyrimas

Baigiamasis magistro projektas

Rokas Bajoriūnas

Projekto autorius

doc. dr. Virginijus Baranauskas

Vadovas

Kaunas, 2025



Kauno technologijos universitetas

Elektros ir elektronikos fakultetas

ABB roboto manipulatoriaus valdymo, naudojant virtualios realybės akinius, galimybių tyrimas

Baigiamasis magistro projektas

Valdymo technologijos (6211EX014)

Rokas Bajoriūnas

Projekto autorius

doc. dr. Virginijus Baranauskas

Vadovas

doc. dr. Gintaras Dervinis

Recenzentas

Kaunas, 2025



Kauno technologijos universitetas

Elektros ir elektronikos fakultetas

Rokas Bajoriūnas

ABB roboto manipulatoriaus valdymo, naudojant virtualios realybės akinius, galimybių tyrimas

Akademinio sąžiningumo deklaracija

Patvirtinu, kad:

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

Rokas Bajoriūnas

Patvirtinta elektroniniu būdu

Bajoriūnas Rokas. ABB roboto manipulatoriaus valdymo, naudojant virtualios realybės akinius, galimybių tyrimas. Magistro baigiamasis projektas / vadovas doc. dr. Virginijus Baranauskas; Kauno technologijos universitetas, Elektros ir elektronikos fakultetas.

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

Reikšminiai žodžiai: Virtuali realybė, Aplikacijų programavimo sąsaja, ABB robotas.

Kaunas, 2025. 50 p.

Santrauka

Šiame darbe išsamiai nagrinėjamos virtualios realybės (VR) akinių pritaikymo galimybės valdant robotus manipulatorius. Tyrime apžvelgiama VR technologijos, kaip priemonės robotizuotų sistemų valdymo sąsajai patobulinti, panaudojimo galimybės ir veiksmingumas. Sukurtas sistemos prototipas ABB roboto manipulatoriaus valdymui naudojant VR akinių programėlę. Tiriant roboto valdymą per šią programėlę siekiama įžvelgti VR akinių naudojimo robotų manipuliavimo užduotims atlikti privalumus ir apribojimus.

Pirmoje darbo dalyje apžvelgiama mokslinė literatūra susijusi su virtualios realybės ir skaitmeninių dvynių taikymu šiuolaikinėje robotikoje, aparatūrinės ir programinės įrangos naudojimu šioms technologijoms realizuoti. Analizuojamos esamos robotizuotos sistemos su VR sąsaja.

Antroje darbo dalyje aprašoma naudota įranga bei VR programėlės kūrimas ABB roboto manipuliatoriui valdyti. Išsamiai pateikiamas programinės įrangos konfigūravimas ir modeliavimas, roboto valdymo algoritmų veikimas.

Trečioje darbo dalyje aprašomas roboto valdymo naudojant VR programėlę tyrimas. Tyrimas susideda iš tikslumo analizės valdant robotą su standartiniu valdymo pultu ir su išorine VR akinių programėle. Matuojami ir lyginami roboto galinio darbo įtaiso koordinačių skirtumai atliekant įvairias pozicionavimo užduotis.

Bajoriūnas Rokas. Investigation of ABB Robot Manipulator Control Possibilities Using Virtual Reality Glasses. Master's Final Degree Project / supervisor doc. dr. Virginijus Baranauskas. Kaunas University of Technology, Faculty of Electrical and Electronics Engineering.

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

Keywords: Virtual Reality, Application programming interface, ABB robot.

Kaunas, 2025. Number of pages – 50.

Summary

This thesis explores the potential of virtual reality (VR) goggles for the control of robotic manipulators. The study provides an overview of the feasibility and effectiveness of using VR technology as a tool to improve the control interface of robotic systems. Prototype system has been developed to control an ABB robotic manipulator using a VR glasses application. By exploring robot control using this application, the aim is to identify the advantages and limitations of using VR glasses for robotic manipulation tasks.

The first part of the thesis reviews the scientific literature related to the application of virtual reality and digital twins in modern robotics, the use of hardware and software to implement these technologies. Analysis of existing robotic systems with VR interface.

The second part of the thesis describes the hardware used and the development of VR application for the control of an ABB robot manipulator. Software configuration and modelling, operation of the robot control algorithms are presented in detail.

The third part of the thesis describes a study on robot control using a VR application. The study consists of a precision analysis of the robot control with a standard control panel and with an external VR glasses application. The differences in the coordinates of the robot's end-effector are measured and compared for different positioning tasks.

Turinys

Paveikslų sąrašas	8
Lentelių sąrašas	9
Įvadas.....	10
1. Teorinė dalis.....	11
1.1. Skaitmeninių dvynių naudojimo sritys šiuolaikinėje robotikoje	11
1.1.1. Kosmoso robotų technika	11
1.1.1.1. Kosmoso technikos MRO (angl. <i>Maintenance, Repair and Operations</i>)	12
1.1.1.2. NASA programa „Artemis“	12
1.1.2. Medicininė robotika.....	12
1.1.2.1. Medicininių komponentų rūšiavimas	13
1.1.2.2. Medicininių komponentų surinkimas	13
1.1.3. Minkštoji robotika	13
1.1.3.1. Minkštojo roboto skaitmeninio dvynio projektavimas.....	14
1.1.4. Žmogaus ir roboto sąveika	14
1.1.4.1. Surinkimo linijų perprojektavimas	14
1.1.4.2. Demonstracinis robotų apmokymas	15
1.1.5. Pramoninė robotika	15
1.1.5.1. Prognozuojama analizė ir įrenginių papildymas	15
1.1.5.2. Prognozuojama techninė priežiūra	15
1.2. Virtualios realybės naudojimas šiuolaikinėje robotikoje	16
1.2.1. Operatorių mokymas	16
1.2.2. Nuotolinis valdymas.....	18
1.2.3. Dizainas ir vizualizacija	19
1.3. Aparatinė ir programinė įranga virtualiai realybei kurti.....	20
1.3.1. Aparatinė įranga	20
1.3.1.1. Daugiajutiminės robotizuotos sistemos su VR sąsaja projektavimas.....	20
1.3.2. Programinė įranga	22
1.3.2.1. Žaidimų variklio pritaikymas VR sąsajai kurti.....	22
1.4. Aparatinė ir programinė įranga skaitmeniniams dvyniams kurti	23
1.4.1. Aparatinė įranga	24
1.4.1.1. Realios sistemos duomenų nuskaitymas iš aparatinės įrangos.....	24
1.4.2. Programinė įranga	24
1.4.2.1. Robotizuotos sistemos skaitmeninio dvynio kūrimas naudojant ROS.....	25
2. Darbo eiga	27
2.1. Naudojamos įrangos aprašymas	27
2.1.1. „RobotStudio“ programinis įrankis ir ABB robotas.....	27
2.1.2. „Visual Studio“ programinis įrankis	28
2.1.3. „Unity“ programinis įrankis	28
2.1.4. „Oculus Quest 2“ virtualios realybės akiniai.....	29
2.2. VR programėlės kūrimas	30
2.2.1. Robotų tinklo paslaugos (angl. <i>Robot Web Services (RWS)</i>)	30
2.2.2. Roboto valdiklio konfigūravimas	32
2.2.3. „RAPID“ programos veikimas	33
2.2.4. VR akinių programėlės modeliavimas ir veikimas	34

3. VR programėlės tyrimas	40
3.1.1. Atsitiktinių taškų pozicionavimas	40
3.1.2. Galinio darbo įtaiso kalibravimas.....	43
3.1.3. Atsitiktinio atstumo poslinkis.....	45
3.1.4. Tyrimų apibendrinimas	47
Rezultatai ir išvados	48
Literatūros sąrašas	49

Paveikslų sąrašas

1 pav. Reali ir virtuali erdvės: (a) – reali darbo vieta, (b) – taškų debesis, (c) – virtuali erdvė, (d) – žmogaus ir roboto sąveikos virtuali erdvė [12].	17
2 pav. Daugiajutiminė sistema su regos, lietimio ir klausos grįžtamuju ryšiu [15].	22
3 pav. Sistemos modelyje ir tyrime naudotas manipulatorius ABB IRB 4600.	28
4 pav. „Oculus Quest 2“ VR akiniai su valdikliais.	29
5 pav. RWS galimų paslaugų ir išteklių schema [20].	31
6 pav. Valdiklio konfigūracijos nustatymų langas „RobotStudio“ aplinkoje.	32
7 pav. RmqType parametro keitimas.	32
8 pav. „AndroidManifest“ XML failo parametrai.	34
9 pav. Virtualaus valdiklio konfigūracijų failo duomenys.	35
10 pav. HTTP kliento funkcijos autentifikacijai realizuoti.	36
11 pav. Roboto valdiklio atsakymas į HTTP užklausą be autentifikacijos.	36
12 pav. „Oculus“ integracijos komponento sandara.	37
13 pav. Virtualios planšetės komponento sandara.	38
14 pav. Virtualios planšetės modelis „Unity“ aplinkoje.	39
15 pav. Virtualios planšetės vaizdas VR akinių programėlėje.	39
16 pav. Roboto galinio darbo įtaiso pozicionavimo atsitiktiniuose taškuose pavyzdys.	40
17 pav. 2 lentelėje aprašytų pozicionavimo taškų išsidėstymas. Raudonas žymeklis – Teach Pendant valdymas, žalias žymeklis – VR programėlės valdymas.	41
18 pav. Roboto galinio darbo įtaiso pozicionavimo vaizdas VR akiniuose.	43
19 pav. Roboto galinio darbo įtaiso kalibravimo langas.	43
20 pav. Roboto galinio darbo įtaiso kalibravimo rezultatų langas.	44
21 pav. Roboto galinio darbo įtaiso poslinkio matavimas.	45
22 pav. Roboto koordinatų ir pasisukimo kampų atvaizdavimas virtualioje aplinkoje.	46

Lentelių sąrašas

1 lentelė. Roboto valdiklio parametrai, pranešimams valdyti:.....	33
2 lentelė. Roboto galinio darbo įtaiso pozicionavimo atsitiktiniuose taškuose koordinatės.....	40
3 lentelė. Roboto galinio darbo įtaiso pozicionavimo atsitiktiniuose taškuose koordinatės nuokrypiai.	42
4 lentelė. Roboto galinio darbo įtaiso kalibravimo rezultatai.	44
5 lentelė. Roboto galinio darbo įtaiso poslinkių tyrimo rezultatai.	46

Įvadas

Nepaisant pažangos robotikos ir automatikos srityje, tradicinės ABB robotų manipuliatorių valdymo sąsajos gali turėti trūkumų, susijusių su naudotojo patirtimi, užduočių efektyvumu ir pritaikymu įvairioms pramoninėms aplinkoms. Esami valdymo metodai, dažnai priklausantys nuo sudėtingų sąsajų ir rankinių įvesties įtaisų, gali trukdyti operatoriaus darbui ir riboti intuityvios sąveikos su robotizuotomis sistemomis galimybes. VR technologija gali pasiūlyti įtraukiančias, intuityvias ir efektyvias robotizuotų sistemų valdymo priemones. Šiame darbe, tiriant virtualios realybės integravimo į ABB roboto manipuliatoriaus valdymo sistemą galimybes ir galimą naudą, siekiama spręsti minėtus iššūkius ir pagerinti robotizuotos sistemos operatoriaus darbą.

Darbo tikslas – sukurti ABB roboto manipuliatoriaus valdymo, naudojant virtualios realybės akinius, sistemą ir ištirti jos veikimo tikslumą.

Projekto įgyvendinimui sprendžiami uždaviniai:

1. apžvelgti esamas robotizuotas sistemas su virtualios realybės valdymo sąsaja;
2. išanalizuoti naudojamą įrangą;
3. sukonfigūruoti roboto valdiklį ir sukurti „RAPID“ valdymo programą;
4. sukurti VR akinių programėlę roboto rankai valdyti;
5. ištestuoti ir palyginti sukurtos programėlės galimybes roboto pozicionavimo užduotims atlikti naudojant VR akinius ir standartinį valdymo pultą.

1. Teorinė dalis

Robotai tapo neatsiejama šiuolaikinės visuomenės dalimi, atliekančia įvairias užduotis įvairiose srityse - nuo gamybos ir sveikatos priežiūros iki tyrinėjimo ir pramogų. Sparčiai tobulėjant technologijoms, tyrėjai ir kūrėjai nuolat ieško novatoriškų būdų, kaip padidinti robotų galimybes ir pagerinti jų sąveiką su žmonėmis. Pastaraisiais metais dvi technologijos - virtualioji realybė (VR) ir skaitmeniniai dvyniai - tapo galingomis priemonėmis, galinčiomis sukelti revoliuciją robotikos srityje.

Virtualioji realybė perkelia vartotojus į kompiuteriu sukurtą aplinką, stimuliuoja jų pojūčius ir leidžia bendrauti su virtualiais objektais ir subjektais. VR integravimas su robotika atveria naujas galimybes mokytis, testuoti ir valdyti robotus virtualioje aplinkoje. Ši technologija suteikia saugią ir kontroliuojamą aplinką, kurioje žmonės gali mokytis ir sąveikauti su robotais, sumažindama riziką, susijusią su fiziniais robotų eksperimentais. Naudodami VR, vartotojai gali įgyti vertingos praktinės patirties dirbant su robotais, nereikalaujant brangių fizinių prototipų ir nerizikuojant patirti nelaimingų atsitikimų. Be to, VR leidžia imituoti sudėtingus ir pavojingus scenarijus, kuriuos atkurti realiame pasaulyje būtų sudėtinga arba neįmanoma.

Skaitmeniniai dvyniai yra virtualios fizinių robotų ar robotų sistemų kopijos. Jie apima roboto dizainą, struktūrą, elgseną ir realaus laiko duomenis, pateikdami išsamią skaitmeninę fizinio subjekto kopiją. Skaitmeniniai dvyniai leidžia nuotoliniu būdu stebėti, optimizuoti ir prognozuoti robotų veikimą. Rinkdami duomenis iš fiziniuose robotuose įmontuotų jutiklių, skaitmeniniai dvyniai suteikia vertingų įžvalgų apie roboto elgseną, būseną ir efektyvumą. Ši informacija gali būti naudojama anomalijoms aptikti, techninės priežiūros poreikiams numatyti ir roboto veiklai optimizuoti, todėl pagerėja roboto našumas ir sumažėja prastovų laikas.

Šiame darbe nagrinėjamas virtualios realybės ir skaitmeninių dvynių taikymas šiuolaikinėje robotikoje, aparatūrinės ir programinės įrangos naudojimas šioms technologijoms realizuoti.

1.1. Skaitmeninių dvynių naudojimo sritys šiuolaikinėje robotikoje

Nuo 1960-ųjų pabaigos skaitmeninių dvynių technologijos naudojimas robotikoje vystėsi ir plėtėsi įvairiose srityse. Iš pradžių sukurtas kaip sudėtingas kibernetinis-fizikinis metodas kosminėje robotikoje, laikui bėgant skaitmeninis dvynys buvo iš naujo apibrėžtas ir pertvarkytas, kad atitiktų šiuolaikinio laikmečio poreikius, ypač realaus laiko robotų gamybos ekosistemose. [1] literatūros šaltinio atliktame tyrime buvo nustatytos penkios robotikos sritys, kuriose skaitmeninių dvynių technologijos pastaraisiais metais buvo įgyvendintos ir parodė didžiausią poveikį:

- kosmoso robotų technika;
- medicininė robotika;
- minkštoji robotika (angl. *soft robotics*);
- žmogaus ir roboto sąveika;
- pramoninė robotika.

1.1.1. Kosmoso robotų technika

Skaitmeniniai dvyniai kosmoso robotų technikoje gali padėti sistemų projektavimo, modeliavimo, veikimo ir eksploatacinės priežiūros srityse. Kurdami virtualias fizinių robotų kopijas, kosmoso

technikos inžinieriai įgyja įžvalgų apie įvairių procesų našumą, elgesį, veikimą. Skaitmeniniai dvyniai suteikia galimybę atlikti išsamius bandymus, stebėti ir valdyti sistemas nuotoliu realiuoju laiku, taip sumažinant rizikas, pagerinant misijų sėkmę ir tobulinant kosmoso tyrinėjimo galimybes.

1.1.1.1. Kosmoso technikos MRO (angl. *Maintenance, Repair and Operations*)

Šiuolaikinės kosminės robotikos srityje skaitmeninių dvynių technologijos pritaikymas pasirodė esąs reikšmingas konstruojant, remontuojant, atnaujinant ir prižiūrint kosminės erdvės komponentus, įskaitant erdvėlaivių korpusus. Ryškus to pavyzdys pateiktas [2] moksliniame darbe, kur specialiai kosminės MRO taikymams sukurta robotizuota šlifavimo sistema, kurioje integruotas skaitmeninis dvynys. Sistemoje naudojama 6-ių laisvės laipsnių roboto ranka, skirta tikslioms ventiliatoriaus menčių atnaujinimo/šlifavimo operacijoms atlikti. Naudojant skaitmeninį dvynį, automatizuotame procese galima analizuoti ir nustatyti svarbiausius šlifavimo kintamuosius, pavyzdžiui, reikiamą šlifavimo jėgą, šlifavimo įtaiso poziciją ir kampą. Į skaitmeninį dvynį buvo įtraukti apribojimai, kurių reikia laikytis šlifavimo proceso metu. Šie apribojimai padėjo rasti geriausius šlifavimo parametrus. Informacija apie šlifavimo parametrus buvo panaudota nustatant būtinas proceso sąlygas, kad būtų pasiektas tam tikras medžiagos pašalinimo greitis. Be to, šlifavimo parametrai buvo naudojami siekiant, kad paviršiaus temperatūra neviršytų verčių, galinčių sukelti terminę deformaciją.

Darbo rezultatai, gauti taikant fizika pagrįstą dalelių modelį skaitmeniniame dvynyje, buvo panašūs į realius eksperimentinius rezultatus. Buvo patvirtinta, kad dėl didesnio pjovimo gylio nuo modeliuojamo ventiliatoriaus mentės paviršiaus buvo pašalinta daugiau medžiagos. Šis pastebėjimas sutapo su vizualiniais ir kiekybiniais rezultatais, gautais iš fizinio ventiliatoriaus mentės paviršiaus. Taip pat, panašiai kaip ir fizinėje automatizavimo kameroje, imituojamai roboto rankai išdavus didelio pjovimo gylio komandą, susidarė didelės trinties jėgos ventiliatoriaus mentės ir šlifuoklio sąveikoje. Realiam pasaulyje tai lemtų variklio perkrovą. Panašus poveikis pastebėtas ir modeliuojant skaitmeninį dvynį. Skaitmeninis dvynys suteikė saugią, ekonomišką ir veiksmingą priemonę įvairioms šlifavimo režimo hipotezėms išbandyti prieš įgyvendinant optimalų rezultatą fizinėje roboto rankoje.

1.1.1.2. NASA programa „Artemis“

Kitas svarbus skaitmeninių dvynių naudojimo pavyzdys kosmoso robotų technikoje galėtų būti NASA programa „Artemis“. Šia plataus užmojo iniciatyva siekiama sugrąžinti žmones į Mėnulį ir sukurti giliojo kosmoso vartus, kurie taptų atspirties tašku tolesniems Marso ir kosmoso tyrimams. Pagrindinis „Artemis“ programos tikslas - naudojant sudėtingas robotų programas ir didelio tikslumo skaitmeninius dvynius, pasiekti aukštą savarankiškumo lygį [3]. Nors NASA dar nepaskelbė išsamių išvadų, verta paminėti programos svarbą, nes ji yra esminis skaitmeninių dvynių momentas kosminės robotikos srityje.

1.1.2. Medicininė robotika

Skaitmeniniai dvyniai medicinos robotikoje išplėtė ne tik chirurginių procedūrų galimybes, bet ir prisidėjo prie medicininių medžiagų gamybos ir rūšiavimo užduočių. Naudojant virtualias robotizuotų sistemų kopijas ir realaus laiko duomenų integravimą, skaitmeniniai dvyniai gali optimizuoti medicinos komponentų gamybą ir rūšiavimą, taip didinant tokių sistemų efektyvumą ir tikslumą.

1.1.2.1. Medicininių komponentų rūšiavimas

Per COVID-19 pandemiją, reaguojant į smarkiai išaugusią medicininių priemonių ir įrangos paklausą, pagreitį įgavo medicininės įrangos surinkimas robotais. 2020 m., atsižvelgiant į pandemijos iššūkius, buvo sukurtas roboto skaitmeninis dvynys, skirtas palengvinti bekontaktį vaistų ir būtiniausių reikmenų rūšiavimą/paskirstymą [4]. Šiame darbe skaitmeninio dvynio projektavimas atliktas naudojant FANUC ROBOGUIDE programinę įrangą ir pagrindinis skaitmeninio dvynio kūrimo tikslas, padėti atsakyti į esminius klausimus, kurie gali kilti organizacijai, norinčiai įdiegti aprašomą sistemą:

- a) Koku greičiu siūloma sistema gali įvykdyti užsakymus?
- b) Kokia yra optimaliausia darbo kelės konfigūracija pageidaujama našumui pasiekti?

Norint atsakyti į šiuos klausimus, reikia atlikti išsamią sistemos pralaidumo ir ciklo trukmės analizę.

Baigtas darbo kelės projektas buvo paleistas modeliavimo programine įranga, kad būtų surinkti ciklo trukmės duomenys. Kadangi imituojama darbo kelė yra skaitmeninis dvynys, sugeneruotas ciklo laikas atspindi tai, kas būtų pasiekta naudojant realų fizinį atitikmenį. Nustatyta, kad vieno užsakymo surinkimo ciklo trukmė yra 32,34 sekundės vienai dėžutei. Numatytas 1 užsakymo per minutę įvykdymo greitis, kai klientai turi pasiimti savo užsakymo dėžes ir išvykti. Tai reiškia, kad 20 valandų veikiantis robotas per dieną gali įvykdyti 1200 užsakymų. Dažnai minimas robotų privalumas - jų gebėjimas dirbti nepertraukiamai. Tačiau robotai turi ribas, leidžiančias užtikrinti nepertraukiamą darbą. Pagrindinės robotų gedimų priežastys yra variklio perkaitimas ir mechaninių pavarų gedimai. Šias problemas galima spręsti mažinant judėjimo greitį, mažinant naudingąją apkrovą (tai paprastai nėra tinkamas pasirinkimas) arba pridėdant pertraukiamą poilsio pertrauką, kad būtų palengvintas nepertraukiamas judėjimas. Pandemijos metu gali būti ilgesni didelio poreikio laikotarpiai, kai sistema turi veikti nepertraukiamai. Todėl labai svarbu suprasti roboto gebėjimą veikti be sutrikimų. Anksčiau nustatant šį gebėjimą reikėjo paleisti fizinį sistemos prototipą, kol jis pasiekdavo lūžio tašką. Tačiau naudojant modeliavimo programinę įrangą, aprašomame darbe, galima įvertinti roboto atsparumą darbui, remiantis tam tikru naudinguoju kroviniu ir veiklos apribojimais. Visas šias teorines problemas ir klausimus pavyko išspręsti naudojant projektuojamos sistemos skaitmeninį dvynį.

1.1.2.2. Medicininių komponentų surinkimas

Kitame pavyzdyje buvo sukurta optimizuota žmogaus ir roboto bendradarbiavimo sistema, skirta patenkinti didėjančią medicinos įrangos, konkrečiai - generatorių ir medicininių ventiliatorių, poreikį. Tyrimas parodė, kad optimizavus medicininės paskirties komponentų surinkimą, bendras sistemos efektyvumas padidėjo 25,09 %, o operatoriaus darbo krūvis sumažėjo 20,24 %. Be to, sistemoje integravus skaitmeninį dvynį gerokai sumažėjo klaidų skaičius - nuo 5,128 % prieš įdiegimą iki 0,513 % integravus skaitmeninį dvynį [5].

1.1.3. Minkštoji robotika

Minkštosios robotikos srityje pastaraisiais metais įvyko didelių pokyčių. Skaitmeninių dvynių taikymas minkštojoje robotikoje suvaidino lemiamą vaidmenį skatinant šią pažangą. Iš pradžių skaitmeninių dvynių taikymas minkštojoje robotikoje pirmiausia buvo orientuotas į virtualiosios realybės (VR) taikymus. Tačiau 2020 m. pastebimas ryškus poslinkis į pažangių minkštųjų robotų manipuliatorių valdymo metodų ir komponentų kūrimą ir plėtojimą [1].

1.1.3.1. Minkštojo roboto skaitmeninio dvynio projektavimas

Esminis iššūkis pateikiant skaitmeninę minkšto roboto kopiją yra tikslus jo formos atkūrimas virtualioje erdvėje [6]. Nors egzistuoja įvairūs metodai minkštųjų robotų skaitmeninius dvynius rekonstruoti, daugeliui jų trūksta tikslumo. Paprasti metodai, tokie kaip vieno lenkimo kampo ir kreivumo parametrų nustatymas, yra nepakankami, kad būtų galima atvaizduoti sudėtingus duomenis, gautus iš propriocepinių jutiklių, kaip išsamų skaitmeninį modelį, kaip tirta [6].

Atsižvelgiant į kompromisus tarp tikslumo, atvaizdavimo paprastumo ir integracijos su išplėstine realybe, tyrime sukurta infrastruktūra, pagrįsta skeleto animacijos metodu. Minkšto roboto manipulatoriaus duomenys buvo fiksuojami įterptais spalvų jutikliais, o RGBD duomenys, gauti iš spalvų jutiklių, buvo transformuoti į žymeklių koordinates naudojant neuroninį tinklą (angl. *Feedforward Neural Network, FFNN*) roboto operacinėje sistemoje (angl. *Robot Operating System, ROS*). Šiems taškams „Unity“ platformoje buvo pritaikyta papildoma interpoliacija ir takelažas, kad būtų gauta skaitmeninė kopija. Įvertinus sistemą paaiškėjo, kad skaitmeninio dvynio atvaizdavimas yra efektyvus ir tikslus, pasiekiant vidutiniškai 60 kadrų per sekundę (FPS) atvaizdavimo greitį, o uždelsimas - 16 milisekundžių (ms).

1.1.4. Žmogaus ir roboto sąveika

Skaitmeniniai dvyniai yra technologija, kuri transformuoja sąveiką tarp žmogaus ir mašinos per virtualias fizinių sistemų replikas. Jie leidžia stebėti, analizuoti ir valdyti sistemas realiuoju laiku, gerinant operatorių patirtį, keliant saugumo lygį, didinant produktyvumą ir suteikiant galimybę atlikti prognozuojamąją analizę.

1.1.4.1. Surinkimo linijų perprojektavimas

Skaitmeniniai dvyniai su dirbtinio intelekto pagalba gali būti efektyviai panaudoti žmogui ir robotui bendradarbiauti (angl. *Human-robot collaboration, HRC*) surinkimo linijose. Viena tyrime [7] buvo sukurta dinaminio perkonfigūravimo sistema, kurioje, remiantis iš 2D ir 3D jutiklių gautais duomenimis, panaudotas dirbtinis intelektas ir ROS (robotų operacinė sistema) surinkimo linijoms perprojektuoti. Sistemoje naudojami mobilūs dviejų rankų robotai, kurie juda gamyklos patalpose ir atlieka įvairias užduotis bendradarbiaudami su žmonėmis operatoriais. Skaitmeninis dvynys aprašomame darbe leidžia modeliuoti įvairius gamybos sistemos parametrus įvairiais lygmenimis, įskaitant surinkimo procesą, gamybos stotį ir liniją. Vienas iš pagrindinių šio metodo privalumų - galimybė dinamiškai atnaujinti skaitmeninį dvynį realiuoju laiku sintetinant duomenis iš daugelio 2D-3D jutiklių, taip pateikiant tikslią ir naujausią informaciją apie faktinį gamybos procesą. Modelis apima ir geometrinę informaciją, ir semantiką, taip sustiprindamas visapusišką gamybos sistemos atvaizdavimą. Skaitmeniniu dvyniu pagrįsta perprojektavimo sistema sukuria optimalias išdėstymo ir užduočių planų konfigūracijas. Sprendimas leidžia internetu perkonfigūruoti sistemą, dinamiškai perskirstant užduotis netikėtų įvykių metu ir realiuoju laiku koreguojant robotų elgseną, kad būtų užtikrintos trajektorijos be susidūrimų nestruktūrizuotoje aplinkoje.

Eksperimentiniai darbo rezultatai parodė, kad skaitmeninio dvynio integravimas į šią sistemą leido 25 % padidinti metinį gamybos našumą, 80 % sutrumpinti roboto elgsenos perprojektavimo laiką ir 69 % sutrumpinti stoties lygio perprojektavimo laiką.

1.1.4.2. Demonstracinis robotų apmokymas

Dar viename tyrime aprašomas robotizuotas medinių konstrukcijų surinkimas bendradarbiaujant žmogui ir robotui [8]. Šiame darbe skaitmeninis dvynys naudojamas demonstratyviai apmokyti robotus, taip išvengiant fizinės roboto ir žmogaus sąveikos. Darbas suskirstytas į keturis etapus: pirmame etape projektuojamas ir išsaugomas visos konstrukcijos 3D modelis. Į bendrą struktūrą įtraukiama informacija apie kiekvieno konkretaus medinio bloko panaudojimą konstrukcijoje. Šie duomenys susiejami su parametriniu konstrukcijos modeliu, kuris gali generuoti panašius duomenis įvairiems projektavimo atvejams. Naudojant geometrinius duomenis, planuojamos ir modeliuojamos roboto trajektorijos. Antrą etapą apima koordinuotų roboto judesių fiksavimas, pavyzdžiui, roboto vedimas pageidaujamu keliu naudojant transliacines ir rotacines trajektorijas skaitmeniniame dvinyje. Trečiame etape kiekvienas roboto veiksmas, pavyzdžiui, judėjimas iš taško į tašką arba atsuktuvo operacija, pateikiamas kaip paslauga su konkrečiais įėjimais ir išėjimais. Šiame etape skaitmeninė planavimo informacija iš 1 etapo ir fizinės trajektorijos iš 2 etapo sujungiamos, kad būtų sukurtas sudėtinių instrukcijų rinkinys, apibrėžiantis konkrečios medienos konstrukcijos surinkimo seką. Paskutiniame etape pasitelkiami fiziniai robotai užduotims atlikti pagal iš anksto nustatytą operacijų seką iš 3 etapo. Šiame darbe naudojami bendradarbiaujantys robotai su integruotais jėgos ir sukimo momento jutikliais. Surinkimo vykdymo metu išmatuoti duomenys renkami ir siunčiami atgal į skaitmeninį dvynį kaip grįžtamasis ryšys, kuris vėliau naudojamas surinkimo sėkmei įvertinti. Jei vykdymas sėkmingas, skaitmeninis dvynys suplanuoja kitą užduotį.

1.1.5. Pramoninė robotika

[9] atlikta išsami analizė atskleidė, kad skaitmeniniai dvyniai gali būti veiksmingai taikomi įvairių tipų pramonės įrenginių techninei priežiūrai, įskaitant reaktyviąją, normatyvinę, būkle pagrįstą, prognozuojamąją ir prevencinę techninę priežiūrą. Pastarieji du metodai - prognozavimo ir prevencinė priežiūra - buvo ypač sėkmingi. Nustatyta, kad šie metodai gali būti taikomi į robotus orientuotose sistemose.

1.1.5.1. Prognozuojama analizė ir įrenginių papildymas

2017 m. Vašalekas ir kiti pristatė skaitmeninio dvynio koncepciją, kurioje skaitmeninio dvynio optimizuotai prognozuojamai analizei ir įrenginių papildymui robotais valdomoje gamybos linijoje buvo naudojami genetiniai algoritmai. Šis metodas parodė teigiamus rezultatus, pasiekus 5,2 % sumažėjusį bendrą gamybos laiką. Tačiau buvo pastebėta, kad tarp skaitmeninio dvynio ir jo fizinio atitikmens buvo maždaug 1 sekundės vėlavimas. Šis vėlinimas buvo laikomas per dideliu, palyginti su dabartinėmis taikomosiomis programomis, kuriose dažnai toleruojamas gali būti tik milisekundžių trukmės vėlinimas [10].

1.1.5.2. Prognozuojama techninė priežiūra

Dar viename tyrime [11] pristatoma metodika, skirta pažangiam fizika pagrįstam modeliavimui įgyvendinti, siekiant palengvinti skaitmeninių dvynių koncepcijos taikymą prognozuojamos techninės priežiūros programose. Siekiant patvirtinti siūlomą metodiką, buvo sukurtas pramoninio roboto skaitmeninis modelis, pademonstruojant jo galimą taikymą prognozuojamos techninės priežiūros scenarijuose. Sistemos skaitmeninis dvynys sukurtas apjungus du programinius paketus: atvirojo kodo modeliavimo programą „OpenModelica“ ir „Matlab“. Kaip teigiama aprašomame darbe – „norint įgyvendinti skaitmeninio dvynio koncepciją, labai svarbu palyginti modeliavimo

rezultatus su duomenimis, gautais iš realios mašinos, kad būtų galima tikslinti skaitmeninį modelį realiuoju laiku“. Todėl, sukūrus ir imitavus skaitmeninį roboto rankos modelį „OpenModelica“ programoje, „Matlab“ naudojama modeliavimo rezultatams filtruoti, struktūrizuoti ir analizuoti. Į skaitmeninį modelį buvo integruoti virtualūs jutikliai, kad būtų galima surinkti duomenis, kurių negalėjo surinkti fiziniai jutikliai arba roboto valdiklis. Šiais virtualiais jutikliais buvo renkami padėties, kampinio greičio ir pagreičio duomenys ties pavarų dėžių išėjimo velenų, palyginti su atitinkamais jų įvesties signalais. Jutikliai buvo integruoti į kiekvieną skaitmeninio modelio pavarų dėžę. Duomenys iš virtualių jutiklių skaitmeniniame modelyje ir duomenys iš realaus roboto valdiklio buvo surinkti ir susisteminti, kad būtų galima nustatyti galimus realaus ir skaitmeninio robotų nuokrypius. Ši informacija buvo panaudota skaitmeninio dvynio koncepcijai įgalinti, atnaujinant roboto skaitmeninio modelio parametrus. Šiuo atveju derinimo mechanizmas buvo sutelktas į kiekvieno skaitmeninio roboto pavarų dėžės modeliavimo parametras, lyginant realius ir prognozuojamus sukimo momento signalus. Tikruosius sukimo momento signalus surinko roboto valdiklis, o prognozuojami signalai buvo generuojami įgyvendinant antros eilės paprastųjų diferencialinių lygčių sistemą, skirtą nuosekliojo manipulatoriaus dinaminei elgsenai, ir taikant virtualių jutiklių duomenis (padėties, greičio ir pagreičio).

Toliau buvo atliktas sistemos modeliavimas, o išėjimo signalai palyginti su realaus roboto išmatuotais signalais. Jei nuokrypis tarp realių ir prognozuojamų signalų viršijo iš anksto nustatytą ribą, modeliavimo parametrai buvo derinami, kad būtų pasiektas kuo tikslesnis modeliavimas. Modeliavimo, palyginimo ir derinimo etapas buvo kartojamas kelis kartus, kol buvo pasiektas norimas rezultatas. Atliktame eksperimentiniame tyrime parodytas prognozavimo analizės taikymas sukimo momento signalams, gautiems iš roboto pavarų dėžės. Per keturias iteracijas prognozuojamų ir faktinių duomenų neatitikimas sumažėjo nuo daugiau kaip 2,5 Nm iki 0-0,5 Nm.

1.2. Virtualios realybės naudojimas šiuolaikinėje robotikoje

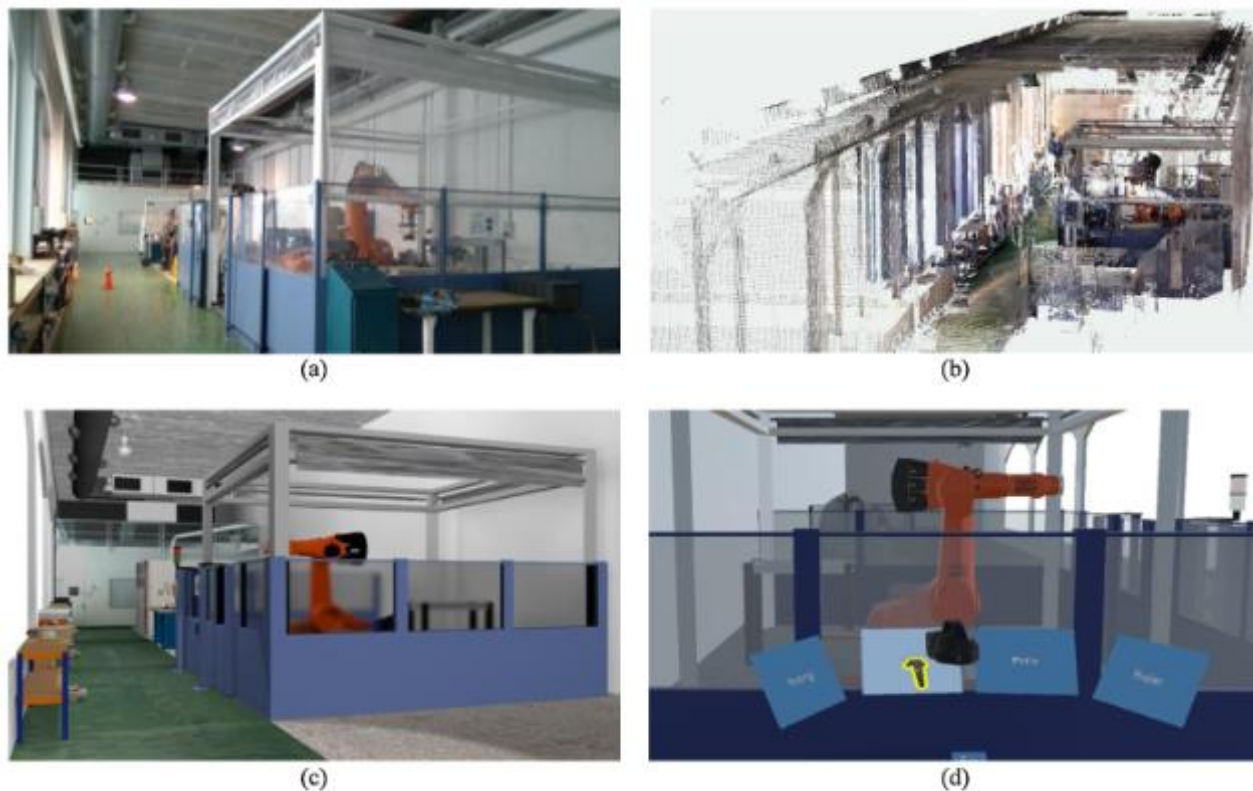
Virtualioji realybė (VR), kaip ir skaitmeniniai dvyniai, tapo perspektyvia robotų technikos srities technologija. Suteikdama įtraukiančią ir interaktyvią patirtį, VR leidžia naudotojams imituoti ir vizualizuoti sudėtingas robotų užduotis virtualioje aplinkoje. Ji leidžia realiai sąveikauti su virtualiais robotais, valdymo sąsajomis ir imituojama aplinka, suteikdama saugią ir ekonomiškai efektyvią platformą mokymams, įgūdžiams tobulinti ir eksperimentams. Dėl gebėjimo atkartoti realaus pasaulio scenarijus ir suteikti haptinį grįžtamąjį ryšį VR iš esmės keičia robotų projektavimo, programavimo ir valdymo būdus. Jos pritaikymo galimybės apima įvairias sritis, įskaitant pramonės automatizavimą, sveikatos priežiūros robotiką, bepilotės sistemas ir kt. Šiame skyriuje aptariama keletas pagrindinių robotikos aspektų, kuriose yra naudojama virtualioji realybė:

- a) mokymas ir modeliavimas;
- b) nuotolinis valdymas;
- c) dizainas ir vizualizacija.

1.2.1. Operatorių mokymas

VR yra saugi ir ekonomiškai robotų operatorių mokymo platforma. Ji leidžia naudotojams imituoti realaus pasaulio scenarijus ir sąveikauti su virtualiais robotais. Besimokantieji gali praktikuotis valdyti robotus, mokytis sudėtingų užduočių ir suprasti robotų elgseną nerizikuodami sugadinti įrangos ar sukelti nelaimingų atsitikimų. VR imitacijos taip pat leidžia išbandyti ir patvirtinti robotų algoritmus ir valdymo sistemas.

[12] nagrinėtame tyrime aprašoma virtualios realybės ir robotikos sinergija, ypač daug dėmesio skiriant komercinių žaidimų technologijoms naudoti kuriant įtraukiančią VR aplinką. Sistema apima sąsają, prijungtą prie roboto valdiklio, į kurią įtraukti matematiniai modeliai tiksliai valdyti virtualų robotą. Siūlomas unikalus ir ekonomiškai efektyvus metodas, skirtas mokytis, modeliuoti ir valdyti robotus realiuoju laiku. Eksperimentiniai tyrimo rezultatai rodo, kad įtraukianti virtualios realybės patirtis gali gerokai padidinti mokymo ir imitavimo procesų efektyvumą.



1 pav. Realioji ir virtualioji erdvės: (a) – realioji darbo vieta, (b) – taškų debesis, (c) – virtualioji erdvė, (d) – žmogaus ir roboto sąveikos virtualioji erdvė [12].

Nagrinėto darbo 5-ame skirsnyje yra aprašomas VR sąsajos kūrimo procesas. Realiojo robotų operatoriaus darbo vieta (1 pav. a)) iš pradžių buvo nuskaityta naudojant "FARO Focus3D X130 HDR" lazerinį skenerį. Gautas 3D taškų debesis (1 pav. b)), kuris toliau buvo apdorotas ir filtruotas naudojant „CloudCompare“ programinį paketą, o tada modeliuotas naudojant „Blender“ modeliavimo įrangą, kad būtų sukurta virtualioji aplinka (1 pav. c)). Palyginus 1 paveikslėlių a ir c dalis, matyti, kad 3D rekonstrukcija yra labai tiksli. Virtualioji aplinka tiksliai atspindi realią aplinką, kurioje užfiksuotos smulkiausios detalės. Galiausiai, kuriant žmogaus ir roboto sąveikos sąsają, buvo panaudota „Unity3D“ programinė įranga. Įdiegti virtualūs mygtukai, skirti naršyti po įvairius meniu, kaip parodyta 1 paveikslėlio d) dalyje.

Virtualiojoje aplinkoje operatorius gali imituoti įvairias padėtis, nagrinėti išskirtinumus, tikrinti pasiekiamumą, tirti galimus griebtuvų susidūrimus ir pan. Šis scenarijus buvo sukurtas siekiant įvertinti siūlomą sistemą, įskaitant įtraukiančią poveikį ir VR sąsają, su tikrais gamyklos operatoriais. Siūlomai metodikai patvirtinti buvo naudojamas robotas KUKA KR500-2, kuriame ryšiui tarp roboto valdiklio ir VR sistemos palaikyti naudojamas Modbus TCP protokolas. Iš įvairių komerciškai prieinamų akinių bandymams buvo pasirinkti „Oculus Rift“ ir „HTC Vive“.

Tyrimo patvirtinimuose ir bandymuose dalyvavo skirtingo profilio ir patirties lygio asmenų: robotų taikymo inžinieriai, robotų operatoriai, operatorių padėjėjai. Jų atsiliepimams surinkti buvo pateiktas klausimynas, apimantis įvairius robotų valdymo aspektus. Remiantis bandytojų atsiliepimais, virtualios realybės naudojimo metodas yra patogus, todėl sutrumpėja mokymo laikas. Naudotojams susipažįstant su aplinka, didėja mokymo atlikti realias užduotis efektyvumas. Taip pat, robotų programų ir trajektorijų testavimas VR aplinkoje sumažina riziką ir pagerina saugą pramonės įmonėse.

1.2.2. Nuotolinis valdymas

Virtualioji realybė gali pagerinti nuotolinį valdymą, kai žmogus operatorius valdo kitoje fizinėje erdvėje esantį robotą. Operatoriui pasineriant į virtualią aplinką, jis gali turėti geresnį buvimo ir gylio suvokimo pojūtį, o tai pagerina nuotolinio roboto valdymą ir manipuliavimą juo. VR sąsajos gali užtikrinti haptinį grįžtamąjį ryšį ir intuityvų valdymą, kad būtų lengviau atlikti tikslų ir įtraukiantį teleoperavimą.

Stabilus ir patogus vizualinis grįžtamasis ryšys yra labai svarbus teleoperacinėms robotinėms sistemoms, naudojamoms medicinoje. Tradiciniai kameromis grindžiami vaizdo transliacijos metodai turi tokių apribojimų kaip kameros skiriamoji geba, ribotas matymo kampas ir netvarkinga aplinka, dėl kurių gali padidėti žmogaus operatoriaus kognityvinė apkrova. Siekiant išspręsti šiuos iššūkius, [13] straipsnyje pristatoma teleoperacinė sistema, pagrįsta virtualia realybe. Sistemoje naudojamas regioninis konvoliucinis neuroninis tinklas (R-CNN) laboratoriniams prietaisams aptikti ir jų padėčiai nuotolinėje aplinkoje įvertinti. Eksperimentų rezultatai rodo, kad sukurta sistema leidžia sklandžiau valdyti robotą, todėl sutrumpėja užduočių vykdymo laikas manipuliuojant mėgintuvėliais. Taip pat, tyrimo dalyviai mano, kad virtualios realybės sistema reikalauja mažiau protinių (11 %) ir fizinių pastangų (16 %), palyginti su kameromis pagrįstu teleoperavimo metodu. Siūlomą technologiją galima pritaikyti atliekant laboratorinius tyrimus atokiose vietovėse, ypač dirbant su infekciniais ir nuodingais reagentais.

Aprašomą sistemą sudaro 6-ių laisvės laipsnių robotas manipulatorius UR3 su 2-jų pirštų griebtuvu (Robotiq 2F-85). Jame taip pat įmontuota 8 megapikselių skaitmeninė kamera, turinti 75 laipsnių matymo kampą, ir „Intel RealSense D435 RGB-D“ kamera, sumontuota ant galinio darbo įtaiso. Robotui valdyti operatorius naudoja stalinį įrenginį Omega 7. Kaip vaizdinio grįžtamojo ryšio su operatoriumi priemonė, virtuali aplinka sukurama naudojant „Unity Engine“. Šią aplinką sudaro skaitmeninė roboto kopija, realaus laiko vaizdo srautas iš 8 MP kameros, esančios ant galinio efektoriaus, ir skaitmeninis laboratorinio prietaiso dvynys. Laboratorinio prietaiso padėtis įvertinama naudojant specialiai šiam tikslui sukurtą kompiuterinės regos sistemą, kurioje naudojama „Intel RealSense“ gylio kamera.

Kad vartotojas geriau suprastų manipuliavimo procesą, darbe realiuoju laiku buvo atnaujinamos roboto ir valdomo objekto būsenos, naudojant skaitmeninio dvynio duomenis ir objekto pozicijos įvertinimą pagal CNN. Laboratorinių instrumentų segmentavimas atliktas mokant R-CNN modelį pagal surinktą duomenų rinkinį. Laboratorinės įrangos modeliai buvo iš anksto suprojektuoti CAD programine įranga ir eksportuoti kaip „Unity“ prefabrikatai. Todėl, gavus objekto centro koordinates RealSense koordinačių sistemoje ir transformavus jas į skaitmeninio dvynio koordinačių sistemą, objektas vizualizuojamas virtualioje aplinkoje.

Darbai įvertinti buvo atlikti eksperimentiniai tyrimai, kuriuos apėmė „paimk ir padėk“ užduotys su mėgintuvėliu. Kiekvienas tyrimo dalyvis turėjo tris bandymus atlikti užduotį. Dalyviams buvo nurodyta paimti tikslinį objektą ir padėti nurodytoje vietoje. Eksperimento pradžioje roboto galinis darbo įtaisas buvo pozicionuojamas 47 cm atstumu nuo tikslinio objekto, o kamera buvo sulygiuota su objekto centru. Operatoriui paspaudus pradžios mygtuką, robotas judėjo remdamasis manipuliacijomis. Kiekvienam dalyviui kiekvienu valdymo režimu išmatuoti kiekybiniai duomenys: užduoties atlikimo laikas, suėmimo tikslumas, roboto trajektorijos ilgis ir bandymų, reikalingų užduočiai atlikti, skaičius.

Eksperimentiniai rezultatai parodė, kad nuotolinis sistemos valdymas naudojant virtualią realybę pasižymėjo mažiausiu vidutiniu trajektorijos ilgiu (0,43 m) ir užduoties atlikimo laiku (87,89 s). Sukurta VR sąsaja leido operatoriui 23,2 % sutrumpinti roboto trajektorijos ilgį iki tikslinio objekto ir 31,3 % sutrumpinti užduoties atlikimo laiką, palyginti su kameromis pagrįstu teleoperavimo metodu. VR pagrįsta sistema taip pat pasižymėjo mažiausia vidutine 14,1 mm paklaida. Trajektorių rezultatai parodė, kad kameromis pagrįsto valdymo režimas reikalavo nereikalingų judesių dėl nepakankamo teleoperacinės zonos vaizdo. Priešingai, VR valdymo režimas pasižymėjo kryptingesniais judesiais tikslo link. Remiantis abiejų valdymo režimų naudotojų vertinimo rezultatai, VR pagrįsta sąsaja padidino naudotojo įsitraukimą 11 %, o našumą - 10 %.

1.2.3. Dizainas ir vizualizacija

VR leidžia inžinieriams ir dizaineriams kurti ir vizualizuoti robotų sistemas virtualioje erdvėje prieš pradedant fizinį konstravimą. Tai leidžia greitai kurti prototipus, įvertinti įvairias robotų konfigūracijas ir optimizuoti dizainą, atsižvelgiant į našumą ir ergonomiką. VR taip pat gali padėti vizualizuoti bei planuoti sudėtingus robotų judesius, trajektorijas ir jutiklių duomenis, kurti ir derinti valdymo algoritmus.

Daugelyje pramoninių projektų naudojami iš anksto paruošti moduliniai gaminiai vėliau yra montuojami mobiliuoju kranu nustatytoje vietoje. Sunkiasvorių mobiliųjų kranų eksploatavimas susijęs su didelėmis išlaidomis ir saugumo sumetimais, todėl būtina tiksliai planuoti sunkiasvorių elementų kėlimo darbus. Įprastiniai sunkiasvorių krovinių kėlimo maršrutų planavimo metodai perpildytose pramoninėse darbo vietose dažnai būna neveiksmingi, užima daug laiko ir yra netikslūs. Priešingai, kompiuteriniai imitaciniai modeliai ir vizualizacija yra vertinga tobulinimo priemonė. [14] moksliniame straipsnyje pristatoma virtualios realybės (VR) aplinka, leidžianti planuoti sunkiasvorių elementų kėlimo procesus. Tokio VR modelio pateikimas ne tik palengvina kritinių krovinių kėlimo planavimą, bet ir pasitarnauja kaip mokymo aplinka, skirta didinti saugos supratimą prieš pradedant realias kėlimo operacijas. Sukurtas VR modelis buvo sėkmingai įdiegtas veikiančioje naftos chemijos gamyklos statybos aikštelėje, kurioje naudojami moduliniai statybos metodai ir sunkiasvoriai mobilieji kranai.

Aprašomame tyrime naudoti „Oculus Rift“ virtualios realybės akiniai, „Xbox“ valdiklis ir „Oculus Rift Constellation“ optinis jutiklis. Autoriai atliko penkis pagrindinius veiksmus, kad sukurtų VR įrankį, palengvinantį sunkiųjų mobiliųjų kranų kėlimo planavimą. Šie veiksmai apėmė 3D krano ir statyb vietės modelio sukūrimą. Šešių laisvės laipsnių mobilaus krano modeliui sukurti naudotas „Unity3D“ žaidimų variklis. Kad naudotojams būtų teikiamos skirtingų vaizdų perspektyvos, į imituojamą modelį buvo integruotos įvairių tipų virtualios kameros. Siekiant pagerinti sprendimus priimančių asmenų ir suinteresuotųjų šalių supratimą ir bendravimą, buvo įtraukti realiu laiku

atliekami skaičiavimai ir informacijos pateikimas, pvz., pajėgumų naudojimo procentas, rizikos įvertinimai. Kaip teigia tyrimo autoriai – „tikroviškas sunkiojo keltuvo darbovietės modelis 3D virtualioje aplinkoje atskleidė iššūkius, kurie anksčiau buvo neprieinami tradicinių 2D planavimo procesų metu“.

1.3. Aparatinė ir programinė įranga virtualiai realybei kurti

Robotikoje kuriant virtualią aplinką, kurioje naudotojai gali vizualizuoti, valdyti ir sąveikauti su robotais, yra naudojamas aparatinės ir programinės įrangos komponentų derinys. Techninė įranga apima tokius svarbius komponentus kaip ant galvos montuojami ekranai (angl. *Head-Mounted Display (HMD)*), judesių sekimo sistemos, įvesties įrenginiai ir robotų sąsajos. Programinės įrangos dalis dažniausiai susideda iš virtualios realybės kūrimo platformų ir robotizuotų sistemų modeliavimo platformų.

1.3.1. Aparatinė įranga

HMD yra labai svarbi VR robotų techninės įrangos dalis. Ją sudaro ausinės/akiniai, kuriuos naudotojai užsideda ant galvos, kad pasinertų į virtualią aplinką. HMD paprastai turi didelės skiriamosios gebos ekranus kiekvienai akiai, todėl matomas stereoskopinis vaizdas. Juose taip pat gali būti įmontuotos ausinės arba garso lizdai, suteikiantys garsinį grįžtamąjį ryšį.

Kad būtų galima tiksliai sekti naudotojo judesius ir gestus, VR įrenginiuose dažnai naudojamos judesių sekimo sistemos. Šiose sistemose naudojamos įvairios technologijos, pavyzdžiui, infraraudonųjų spindulių kameros, lazeriniai jutikliai arba vidinis sekimas, kad būtų galima stebėti naudotojo galvos ir rankų padėtį ir orientaciją. Ši sekimo informacija vėliau naudojama naudotojo vaizdui virtualioje aplinkoje atnaujinti ir sąveikai su virtualiais objektais užtikrinti.

VR robotikoje reikalingi įvesties įrenginiai, kad naudotojas galėtų sąveikauti virtualioje erdvėje. Dažniausiai naudojami rankiniai valdikliai, kuriuose yra mygtukai, paleidikliai ir judesio jutikliai. Šie valdikliai leidžia naudotojams manipuluoti virtualiais objektais, valdyti robotų judesius arba atlikti tam tikrus kitus veiksmus. Kiti įvesties įrenginiai - tai duomenų pirštinės, galinčios fiksuoti pirštų judesius, arba taktilinio grįžtamojo ryšio įrenginiai, suteikiantys lytėjimo pojūčius.

Tam tikrose programose gali prireikti techninės įrangos sąsajų, kad VR sistemą būtų galima sujungti su fiziniais robotais. Šios sąsajos gali apimti robotui pritaikytus jutiklius arba pavaros įtaisy, kuriais fiksuojami duomenys arba atliekami roboto judesiai. Pavyzdžiui, VR sistema gali naudoti išorinius jutiklius roboto sąnarių kampams stebėti ir komandoms siųsti teleoperavimo ar valdymo tikslais.

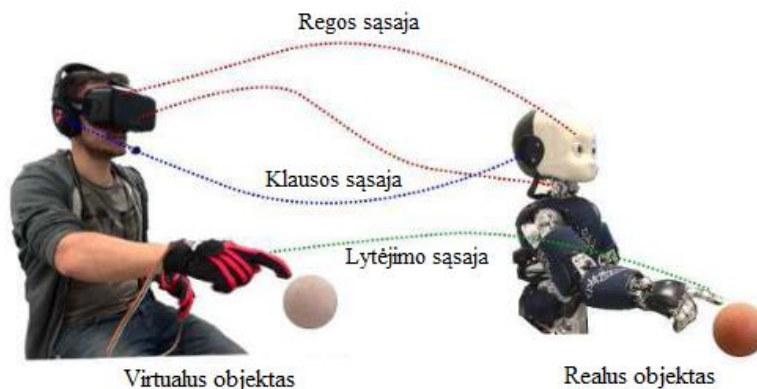
1.3.1.1. Daugiajutiminės robotizuotos sistemos su VR sąsaja projektavimas

[15] tyrime aprašytas humanoidinio roboto su virtualios realybės sąsaja „iCub“ naudojimas ir valdymas. Robotas sukurtas atsižvelgiant į biomimikų principus, tiksliai imituoja įvairias žmogaus funkcijas ir jutimo gebėjimus. Ši atvira platforma, turinti 53 laisvės laipsnius, leidžia atlikti sudėtingus ir vikrius judesius, todėl yra labai pažangi ir tinkama sistema kognityviniam vystymuisi. Pažymėtina, kad roboto rankos ir plaštakos yra biomimetiškai sukurtos atlikti natūralius ir įgudusius judesius, o jo šarnyrinė galva ir akys leidžia atlikti sklandžius ir tikslus galvos judesius. Turėdamas regos, lietimio ir klausos jutiklius bei kompiuterinius modelius, robotas gali bendrauti, tyrinėti ir suvokti aplinką panašiai kaip žmogus.

Kad naudotojas gautų regimąją grįžtamąją ryšį iš roboto valdymo, darbe naudoti „Oculus Rift“ virtualios realybės akiniai, prijungti prie abiejų humanoidinio roboto akių. Vaizdo modulis iš roboto akių gauna du vaizdo srautus, kurie vėliau perduodami į „Oculus Rift“ ekranus. Siekiant užtikrinti minimalų vėlavimą ir aukštos kokybės vaizdinę grįžtamąją ryšį, naudojamas „Motion Joint Photographic Experts Group“ (MJPEG) kodavimo ir dekodavimo modulis. Modulis gerokai sumažina duomenų kiekį ir duomenų srautams perduoti reikalingą pralaidumą, todėl vaizdas rodomas maždaug 25 Hz kadrų dažniu su minimaliomis skaičiavimo sąnaudomis. Į „Oculus Rift“ integruota kelių ašių galvos sekimo sistema, kuri fiksuoja naudotojo, dėvinčio akinius, posvyrio ašių duomenis. Šie duomenys naudojami roboto galvos judesiams valdyti, todėl naudotojai gali tyrinėti nuotolinę aplinką taip, tarsi joje fiziškai dalyvautų. Naudotojo galvos judesių duomenys atvaizduojami į atitinkamas roboto posvyrio ašis naudojant „iCub“ sukurtą Dekarto regos valdiklį. Ši dvikrypčio ryšio sąranka leidžia naudotojams ne tik matyti roboto akimis, bet ir valdyti roboto galvos judesius, kad galėtų vizualiai tyrinėti nuotolinę aplinką. Judesių valdymo modulyje įdiegtas proporcinis valdiklis, skirtas naudotojo galvos judesių (taikinio) ir dabartinės roboto galvos padėties paklaidai, kurią pateikia Dekarto regos modulis, nustatyti. Be to, judesių valdymo modulis atlieka pradinį kalibravimą, kad užfiksuotų roboto galvos judesių diapazoną, užkirsdamas kelią bet kokiems potencialiai žalingiems judesiams už roboto valdymo ribų.

Lytėjimo pojūtis taip pat yra vertingas informacijos šaltinis, kuris dažnai nepakankamai įvertinamas. Norint suteikti naudotojams taktilinę grįžtamąją ryšį, darbe buvo sukurta pora taktilinių pirštinių, kurios perduoda grįžtamąją ryšį iš humanoidinio roboto pirštų galiukų ir delnų. Robote įrengti pažangūs lytėjimo jutikliai, kurie, kartu su pirštinėmis, leidžia naudotojams fiziškai suvokti objektų kietumą ar kontaktą per vibracijas, atvaizduojamas pagal roboto rankų spaudimą. Lytėjimo pirštines sudaro miniatiūriniai „Precision Microdrives“ monetiniai vibraciniai varikliai. Šie varikliai, pritvirtinti prie pirštinių pirštų galų ir delnų, generuoja valdomą vibraciją. Fizinio prisilietimo grįžtamasis ryšys pagerina teleoperacijos patirtį, leisdamas naudotojams apčiuopti nutolusioje aplinkoje esančius objektus.

Iš pradžių tyrime naudoti roboto galvoje esantys mikrofoniškai garso grįžtamajam ryšiui gauti. Tačiau dėl didelio šių mikrofonų fiksuojamo triukšmo lygio jie pasirodė nepraktiški. Norint tai išspręsti, akustinis grįžtamasis ryšys įgyvendintas naudojant du daugiakrypčius „Hama Inc.“ LM-09 modelio mikrofonus. Šie mikrofoniškai, išdėstyti abiejose roboto pusėse, leidžia aptikti stereogarsą iš robotą supančios aplinkos. Mikrofonų užfiksuotas garsas perduodamas naudotojui per Creative HS800 garso ausines. Visa darbe suprojektuota daugiajutiminė sistema, jungianti regos, lietimą ir klausos modulius, pavaizduota 2 paveiksle.



2pav. Daugiajutiminė sistema su regos, lietimo ir klausos grįžtamuju ryšiu [15].

1.3.2. Programinė įranga

VR kūrimo platformos suteikia programinės įrangos įrankius ir sistemas, reikalingas VR programoms kurti. „Unity3D“ ir „Unreal Engine“ yra plačiai naudojamos platformos, siūlančios išsamias kūrimo aplinkas, įskaitant vaizdinius redaktorių, scenarijų kūrimo galimybes ir fizikos variklius. Šios platformos leidžia kūrėjams kurti interaktyvią VR patirtį, importuoti 3D modelius ir kurti tikroviškas virtualias aplinkas.

Robotikos modeliavimo programinė įranga integruoja VR galimybes ir leidžia naudotojams valdyti virtualius robotus imituojamose aplinkose. Tokiose platformose kaip „Gazebo“, V-REP (Virtual Robot Experimentation Platform) arba „Webots“ pateikiami fizikos varikliai ir realistinio modeliavimo funkcijos, skirtos robotų elgesiui imituoti. Dažnai jos siūlo API (mechanizmai, kurie leidžia dviem programinės įrangos komponentams bendrauti tarpusavyje naudojant apibrėžimų ir protokolų rinkinį) arba papildinius, skirtus prisijungti prie VR sistemų, kad naudotojai galėtų teleoperuoti arba išbandyti robotų valdymo algoritmus virtualioje erdvėje.

ROS (robotų operacinė sistema) yra lankstus įrankis, plačiai naudojamas kuriant robotizuotas sistemas. Joje pateikiamas bibliotekų, įrankių ir ryšių protokolų rinkinys, skirtas robotų taikomosios programoms kurti. ROS galima integruoti su VR sistemomis, taip užtikrinant virtualios aplinkos ir roboto ryšį bei koordinavimą. Ji leidžia kūrėjams dalytis duomenimis, valdyti robotų judesius ir gauti jutiklių informaciją.

Siekiant pagerinti naudotojų sąveiką ir vizualizaciją VR aplinkoje, galima naudoti įvairias bibliotekas ir SDK (angl. *software development kits*). Šios bibliotekos tvarko konkrečias VR funkcijas, pavyzdžiui, įvesties įrenginių valdymą, 3D grafikos atvaizdavimą arba fizikos modeliavimo įgyvendinimą. Pavyzdžiui, SteamVR, Oculus SDK, OpenVR arba OSVR (Open Source Virtual Reality).

1.3.2.1. Žaidimų variklio pritaikymas VR sąsajai kurti

[16] straipsnyje išsamiai aprašomi pagrindiniai etapai, susiję su atviro ir lengvai pritaikomo žmogaus ir roboto bendradarbiavimo modeliavimo metodo kūrimu gamybos inžinerijos srityje, pasitelkiant virtualiąją realybę. Šis metodas remiasi naujausiais žaidimų pramonės programinės įrangos kūrimo pasiekimais ir naudoja komerciškai prieinamą aparatinę įrangą. Tokiu siūlomą metodu įveikiami gamybos mašinų gamintojų teikiamos VR programinės įrangos apribojimai. Siekiant įrodyti metodo

įgyvendinamumą, pateikiamas praktinis virtualios aplinkos modeliavimo pavyzdys naudojant „Unity“ žaidimų variklį.

Aprašomo projekto tikslas - atkurti ABB IRB 120 robotą manipuliatorių virtualiosios realybės (VR) aplinkoje, kad operatorius galėtų valdyti jo judesius dėvėdamas HMD (ant galvos montuojamą ekraną). Šiam tikslui pasirinktas HMD - HTC Vive TM.

Modeliuojant naudojamas virtualus ABB IRB 120 robotas manipulatorius, valdomas naudojant į „Unity“ integruotą algoritmą. Šis algoritmas sujungia genetinius algoritmus ir dalelių optimizavimą, atvirkštinės kinematikos uždaviniui spręsti. Jis leidžia animuoti virtualias rankas, kurių judesiai natūraliai primena žmogaus judesius, pritaikant bet kokį sąnarių skaičių. Roboto manipulatoriaus galinio efektoriaus padėtis ir orientacija susiejama su erdvine informacija, kurią fiksuoja operatoriaus turimas valdiklis realiaame pasaulyje. Operatoriui judinant ranką, galinio efektoriaus padėtis perskaičiuojama naudojant algoritmą, todėl judesiai yra sklandūs ir natūralūs.

Operatoriaus rankos judesiai nėra tiesiogiai atvaizduojami į roboto manipulatoriaus judesius pagal atskirus sąnarius. Vietoj to operatorius, manipuliudamas valdikliu, nurodo norimą roboto manipulatoriaus galinio įtaiso padėtį. Kinematikos algoritmas išsprendžia įvairius klausimus, įskaitant santykinės galinio įtaiso padėties įvertinimą pagal operatoriaus rankos poziciją, roboto pasiekiamumo pritaikymą pagal operatoriaus rankos padėtį ir roboto judesių išlyginimą, kad būtų galima pasiekti padėtis, kurių operatoriaus ranka negali lengvai pasiekti. Kai atvirkštinės kinematikos sistemai nepavyksta pasiekti tikslinės padėties, naudotojui pateikiamas taktilinis grįžtamasis ryšys - vibracija.

Bendra VR sąveika su robotu manipulatoriumi yra suskirstyta į tris lygius: pozavimą, įrašymą ir žaidimą. Pozavimo režimu operatorius laisvai veda robotą manipuliatorių norima trajektorija. Įrašymo režimu visi judesiai įrašomi kaip trajektorija. Žaidimo režimu robotas nepriklausomai nuo operatoriaus judesių laikosi įrašytos trajektorijos.

Tyrimo išvadose pabrėžiama, kad žaidimų varikliai suteikia galimybę naudotojams atvirai pasiekti įvairias funkcijas, pavyzdžiui, CAD modelius, medžiagas, tekstūras, skirtingus scenarijus, atvaizdavimą, garso efektus ir animacijas. Nors priėjimas prie CAD modelių tarp gamybos programinės įrangos naudotojų yra įprastas dalykas, dalijimasis ištekliais modeliavimo programinėje įrangoje skatina atvirą pramoninių programų bendruomenę, kuri naudoja bendrus įrankius su pritaikyta elgsena toje pačioje modeliavimo programinėje įrangoje.

Pateiktas metodas susiduria su dideliais iššūkiais, kai reikia visiškai suderinti žaidimų programinę įrangą su gamybos inžinerijos modeliavimui reikalingais programinės įrangos standartais. Tačiau pademonstruotas taikymas ir nurodyti privalumai skatina toliau tyrinėti šią sritį.

1.4. Aparatinė ir programinė įranga skaitmeniniams dvyniams kurti

Skaitmeniniams dvyniams robotų technikoje, kaip ir projektuojant virtualią realybę, reikia derinti aparatinės ir programinės įrangos komponentus. Šie komponentai leidžia realiuoju laiku sinchronizuoti, keisti duomenimis ir valdyti fizinį robotą ir jo skaitmeninį atitikmenį.

1.4.1. Aparatinė įranga

Kalbant apie aparatinę įrangą, robotai yra fiziniai subjektai, kuriuose įrengti jutikliai ir vykdikliai, fiksuojantys realiuoju laiku gaunamus aplinkos duomenis bei realizuojantys skaitmeninio dvynio komandas.

Jutikliai atlieka svarbų vaidmenį fiksuojant duomenis iš fizinio roboto ir jo aplinkos. Kameros teikia vaizdinę informaciją, gylio jutikliai leidžia suvokti 3D vaizdą, jėgos ir sukimo momento jutikliai matuoja kontaktines jėgas, enkoderiai stebi sąnarių padėtis, o lazeriniai jutikliai leidžia detaliam kartografuoti aplinką. Šie jutikliai realiuoju laiku teikia grįžtamąjį ryšį skaitmeniniam dvyniui, todėl galima tiksliai modeliuoti ir valdyti realias sistemas.

Vykdikliai yra atsakingi už skaitmeninio dvynio komandų pavertimą fiziniiais judesiais. Priklausomai nuo roboto konstrukcijos ir pritaikymo, tai gali būti elektros varikliai, servopavaros, hidraulinės ar pneumatinės sistemos, griebtuvai ar kiti mechanizmai. Pavaros leidžia skaitmeniniam dvyniui imituoti fizinius roboto veiksmus.

Kai kuriais atvejais gali būti naudojami daiktų interneto įrenginiai papildomiems duomenims apie robotą ir jo veikimo sąlygas rinkti. Temperatūros jutikliai, drėgmės jutikliai, vibracijos jutikliai gali suteikti vertingos informacijos apie roboto veikimą ir aplinką.

1.4.1.1. Realios sistemos duomenų nuskaitymas iš aparatinės įrangos

[17] straipsnyje siūlomas automatizuotas metodas, skirtas kėbulo gamybos sistemos skaitmeniniam dvyniui sukurti, naudojant turimus išteklius, gaminio duomenis ir proceso informaciją iš fizinės sistemos. Kibernetinė fizikinė gamybos sistema apima daugybę aparatinės informacijos apie kiekvieną elektrinį ir mechatroninį prietaisą, kartu ir programinės įrangos informaciją bei realaus laiko duomenis. Šiuos duomenų šaltinius galima panaudoti kuriant skaitmeninį dvynį gamybos planavimo tikslais. Straipsnyje nagrinėjamos įvairios metodikos realios sistemos duomenims nuskaityti ir pritaikyti skaitmeniniame dvynyje. Šie metodai apima aparatinės įrangos konfigūracijos analizę, programinės įrangos analizę ir realaus laiko informacijos rinkimą.

Aprašomoje gamybos sistemoje programuojamieji loginiai valdikliai (PLC) ir robotai tarnauja kaip pagrindiniai įrenginiai. PLC valdo robotų, kitų įrenginių, jutiklių ir pavarų darbą, o kiekvienas robotas atsakingas už individualias užduotis.

Vienas iš būdų gauti sistemos duomenims - atlikti tinklo nuskaitymą. Kitas metodas apima roboto ir PLC įrangos konfigūracijų naudojimą gauti išsamią informaciją apie visus elektroninius ir mechatroninius įrenginius. Originalios įrangos gamintojai naudoja atsarginių kopijų serverius, kuriuose saugomos PLC ir robotų programinės ir aparatinės įrangos konfigūracijos. Atsarginių kopijų serveriai gali būti naudojami PLC ir robotų programinei įrangai analizuoti. Analizuojant programinę įrangą taip pat galima sužinoti papildomos informacijos apie mechaninę įrangą: PLC programinėje įrangoje galima nustatyti stočių ir gamybos linijų ciklą trukmę, robotų programinė įranga suteikia informacijos apie roboto grandžių išsidėstymą ir apkrovos lygius.

1.4.2. Programinė įranga

Kalbant apie programinę įrangą, naudojami modeliavimo įrankiai, vizualizavimo priemonės, robotikos sistemos ir bibliotekos, duomenų analizės ir mašininio mokymosi metodai. Šie programinės

įrangos komponentai palengvina tikslių virtualių kopijų kūrimą, algoritmų ir valdymo strategijų kūrimą bei testavimą ir iš fizinio roboto surinktų duomenų analizę.

Imitavimo programinė įranga leidžia sukurti virtualią roboto ir jo aplinkos kopiją. Šios priemonės suteikia fizikos variklius, atvaizdavimo galimybes ir programavimo sąsajas roboto dinamikai, sąveikai ir elgsenai apibrėžti. Inžinieriai gali kurti, išbandyti ir patvirtinti algoritmus ir valdymo strategijas imitacinėje aplinkoje prieš jas įdiegdami fiziniame robote.

Kompiuterizuoto projektavimo programinė įranga leidžia sukurti tikslius roboto ir jo komponentų 3D modelius. Vizualizavimo įrankiai užtikrina tikrovišką virtualios aplinkos atvaizdavimą, todėl lengviau suprasti ir analizuoti skaitmeninio dvynio elgseną.

Robotinės sistemos kaip ROS (robotų operacinė sistema), turi įrankių, bibliotekų ir ryšių protokolų rinkinį, skirtą robotų programoms kurti. Jos suteikia standartizuotas sąsajas robotams valdyti, keistis duomenimis ir integruoti skirtingus programinės įrangos komponentus.

Duomenų analizės ir mašininio mokymosi metodai naudojami duomenims, surinktiems iš fizinio roboto ir jo skaitmeninio dvynio, apdoroti ir analizuoti. Tai gali būti įžvalgų išgavimas, dėsningumų nustatymas, našumo optimizavimas arba prognostinės techninės priežiūros užtikrinimas. Tokios bibliotekos kaip TensorFlow, PyTorch ar scikit-learn suteikia įrankių šiems algoritmams įgyvendinti.

1.4.2.1. Robotizuotos sistemos skaitmeninio dvynio kūrimas naudojant ROS

[18] tyrime siūlomas programinės įrangos architektūros projektas, kuriame diskrečiųjų įvykių procesų imitatoriai integruojami su robotų operacine sistema (ROS), leidžiančia sklandžiai keistis informacija tarp įvairių gamyklos komponentų. Ši integracija palengvina greitą skaitmeninių dvynių kūrimą robotizuotoms gamybos sistemoms. Pasiūlymas taip pat leidžia nesudėtingai integruoti skaitmeninį dvynį su autonomine sprendimų priėmimo sistema.

Robotų ir automatizuotos įrangos gamintojai šiais laikais paprastai pateikia savo gaminių kompiuterinius imitatorius, kuriais remiantis kuriami kiekvienos atitinkamos įrangos ar subjekto skaitmeniniai dvyniai. Tačiau daugelis šių atskirų sprendimų nesuteikia visapusiškos perspektyvos pramoninės gamybos sistemos požiūriu. Dėl gamybos procese dalyvaujančių subjektų sudėtingumo ir įvairovės imitatoriams, skirtiems visiškai modeliuoti gamybos sistemą, sunku tiksliai atspindėti visą sistemą. Šie imitatoriai dažnai supaprastina sudedamųjų subjektų dinamiką, dėl to gali nukentėti visos pramoninės gamybos sistemos skaitmeninio dvynio kūrimas. Šiuo atžvilgiu programinės įrangos architektūros, pagrįstos robotų operacine sistema (angl. *Robot Operating System, ROS*), kūrimas tyrimo autoriams pasirodė esąs perspektyvus variantas skaitmeniniam dvyniui realizuoti išmaniosiose gamybos sistemose.

ROS yra sistema, sudaryta iš heterogeniškų ir nepriklausomų procesų, vadinamų mazgais, kurie gali veikti vienu metu ir tarpusavyje keistis pranešimais. Ryšys tarp mazgų vyksta per temas, kai mazgas gali skelbti pranešimus arba prenumeruoti tam tikrą temą. Tokia lanksti architektūra leidžia skirtingomis programavimo kalbomis sukurtus mazgus inkapsuliuoti į ROS, o tai palengvina sistemos plėtrą ir prototipų kūrimą. Ji leidžia sklandžiai įtraukti naujus mazgus, kurie valdo skirtingus įrenginius, pavyzdžiui, jutiklius ir vykdiklius, net jei jie turi patentuotą programinę įrangą ir skirtingas charakteristikas. Dėl šio universalumo ROS yra ypač vertinga atliekant mokslinius tyrimus, kuriems reikia greitai integruoti įvairius programinės ir aparatinės įrangos komponentus.

Skaitmeniniams dvyniams kurti labai svarbų vaidmenį atlieka ryšių protokolų standartizavimas. Atsižvelgiant į tai, kad skaitmeninį dvynį sudaro imitaciniai modeliai, sujungti su realiais duomenimis, juos naudinga kurti su galimybe sąveikauti su realiais subjektais naudojant ROS pagrįstą programinės įrangos architektūrą. Tai leidžia visus duomenis generuojančius ir jais besidalijančius subjektus (pvz., robotus, operatorius, stakles, sprendimų priėmimo sistemas) tarpusavyje sujungti per konkrečius ROS mazgus, palengvinant keitimąsi informacija realiuoju laiku. Šie realaus laiko duomenys gali būti naudojami skaitmeninių dvynių modeliams atnaujinti.

Tyrime siūlomoje architektūroje ROS laikoma vienu komponentu, kurį sudaro du mazgai: ROScore ir Plant Simulation. Mazgą ROScore sudaro ROS procesų ir programų rinkinys, leidžiantis palaikyti ryšį tarp heterogeninių tinklo mazgų. Plant Simulation - įrenginio modeliavimo mazgas yra procesas, atsakingas už imituojamo gamybos įrenginio veikimą. Jis gali bendrauti su kitais mazgais per temas kaip prenumeratorių arba leidėjas. Pavyzdžiui, jei gamybos proceso užduočių parametrai gaunami atlikus modeliavimo tyrimą neprisijungus, jie gali būti tiesiogiai paskelbti atitinkamiems realios gamyklos subjektams per atitinkamą temą. Panašiai, gamyklos modeliavimo mazgas gali prenumeruoti realaus subjekto, kuris daro įtaką konkrečios gamyklos modeliavimo procesui, pateiktą duomenų temą. Ši grįžtamojo ryšio tarp fizinių ir virtualių sistemų schema sudaro pagrindą kuriant pramoninės sistemos skaitmeninį dvynį.

Kaip nurodomas aprašyto tyrimo išvadose, pagrindinis darbo tikslas buvo - programinės įrangos architektūros, pagrįstos ROS, kuri leidžia greitai kurti skaitmeninius dvynius, sukūrimas. Sukurta architektūra integruojama su gamybos procesų imitatoriais ir sprendimų priėmimo sistemomis, palengvinant tiesioginį įvairių aparatinės ir programinės įrangos komponentų tarpusavio ryšį ir keitimąsi duomenimis lanksčiuose robotizuotuose gamybos procesuose.

2. Darbo eiga

Virtualios realybės technologija šiuolaikiniame pasaulyje sparčiai tobulėja ir gali turėti daugybę pritaikymo būdų robotikos srityje. Aprašomame darbe VR naudojama ABB roboto manipuliatoriui valdyti, nenaudojant standartinės ABB robotų operatoriaus sąsajos. VR galimybėms tirti ir palyginti sukūriau virtualios realybės (VR) programėlę, skirtą valdyti ABB robotui. Šioje darbo dalyje detalai apžvelgiama programėlei kurti naudota programinė ir aparatinė įranga, įrangos konfigūravimai ir veikimo algoritmų kūrimas.

2.1. Naudojamos įrangos aprašymas

Naudotos programinės įrangos sąrašą sudaro kompiuterinės programos, skirtos virtualiam roboto manipuliatoriui ir valdikliui atvaizduoti, programiniam kodui rašyti, sukurtai programėlei testuoti, virtualiai realybei kurti. Naudotą fizinę įrangą sudaro virtualios realybės akiniai ir ABB robotas. Pilnas šio darbo įrangos sąrašas: „RobotStudio 2023“, „Visual Studio 2022“ ir „Unity 2022“ programiniai įrankiai, virtualios realybės akiniai „Oculus Quest 2“ ir ABB IRB 4600 robotas su IRC5 valdikliu.

2.1.1. „RobotStudio“ programinis įrankis ir ABB robotas

„RobotStudio“ yra robotizuotų sistemų modeliavimo ir programavimo programinė įranga, naudojama pramoninių robotų programoms optimizuoti ir kurti. Ją sukūrė pirmaujanti robotikos bendrovė ABB. Šis įrankis leidžia vartotojams sukurti virtualią aplinką, skirtą robotams modeliuoti, programuoti ir išbandyti, prieš faktiškai diegiant robotą realioje aplinkoje. Keletas pagrindinių „RobotStudio“ savybių ir funkcijų:

- a) vartotojai gali kurti robotizuotų sistemų 3D modelius, įskaitant patį robotą, ruošinius ir kitą įrangą, imituoti ir vizualizuoti roboto judesius, tikrinti, ar nėra susidūrimų su darbo vieta, atlikti reikiamus pakeitimus be jokio fizinio kontakto su robotu;
- b) suteikia patogią programavimo aplinką, kurioje naudotojai gali rašyti, redaguoti ir derinti robotų programas. Įrankis dažniausiai programuojamas naudojant originalią „RobotStudio“ programavimo kalbą „RAPID“, tačiau taip pat palaiko ir tokias populiarias kalbas kaip C# ar „Python“;
- c) vartotojai gali kurti, keisti ir derinti robotų programas nedalyvaujant tikram robotui. Taip sutaupoma gamybos laiko ir sumažėja fizinio buvimo roboto kameroje poreikis. Užprogramuotas užduotis galima perkelti į tikrą robotą;
- d) „RobotStudio“ apima optimizavimo įrankius, padedančius naudotojams pagerinti roboto veikimą ir našumą: ciklo trukmės analizė, judesių optimizavimas ir susidūrimų aptikimas;
- e) „RobotStudio“ sukurta taip, kad ją būtų galima lengvai integruoti su kita programine įranga ir sistemomis. Įrankis palaiko įvairius CAD formatus, skirtus importuoti ir eksportuoti robotizuotų sistemų projektus, leidžia sklandžiai palaikyti ryšį su išoriniais prietaisais ir įranga;

„RobotStudio“ įrankį šiame darbe naudojau virtualiai roboto manipuliatoriaus sistemai atvaizduoti, programuoti ir testuoti. Virtualų robotizuotos sistemos modelį sudaro ABB IRB 4600 manipuliatorius (3 pav.) ir roboto valdiklis IRC5.

ABB IRB 4600 manipulatorius yra pritaikytas įprastai pramoninei veiklai. Pagrindinės specifikacijos: manipulatorius turi 6 grandis, manipulatoriaus svoris – 435kg, maksimalus keliamas svoris – 60kg, pasiekiamumas – 2.55m. Tolimesnėse darbo dalyse kuriama sistema bus tirama naudojant realų ABB IRB 4600 roboto manipuliatorių.

Naudotas IRC5 manipulatoriaus valdiklis su programinės įrangos versija: „RobotWare“ 6.15.04.00.



3 pav. Sistemos modelyje ir tyrime naudotas manipulatorius ABB IRB 4600.

2.1.2. „Visual Studio“ programinis įrankis

„Visual Studio“ yra integruota programų kūrimo aplinka (IDE), sukurta „Microsoft“. Joje programuotojams teikiamas išsamus įrankių ir paslaugų rinkinys, skirtas įvairių tipų programoms, įskaitant darbalaukio, žiniatinklio, mobiliąsias, debesų, žaidimų ir kitas, kurti. „Visual Studio“ palaiko kelias programavimo kalbas, pavyzdžiui, C#, C++, „Visual Basic“, „Python“ ir „JavaScript“, ir siūlo tokias funkcijas kaip kodo redagavimas, derinimas, versijų kontrolės integracija ir didelė plėtinių bei šablonų biblioteka.

Šiame darbe „Visual Studio“ įrankį naudoju išorinėms roboto manipulatoriaus valdymo programoms kurti. Programų kūrimas atliktas programuojant C# programavimo kalba.

2.1.3. „Unity“ programinis įrankis

„Unity“ yra populiarus ir plačiai naudojamas žaidimų variklis, kurį sukūrė „Unity Technologies“. Įrankis suteikia galimybę kūrėjams kurti aukštos kokybės 2D ir 3D žaidimus, imitacijas ir kitas programas. „Unity“ dažniausiai programuojamas C# programavimo kalba, naudoja komponentais pagrįstą modeliavimą, turi vizualinį redaktorių ir palaiko daug aplikacijų programavimo sąsajų. Naudojant šį įrankį galima kurti programas įvairioms platformoms: „Windows“, „MacOS“, „iOS“, „Android“, „Xbox“, „PlayStation“ ir t.t. Be to, „Unity“ taip pat turi daugybę įrankių ir funkcijų,

įskaitant fizikos modeliavimą, animaciją, garsą, todėl tai yra universalus ir išsamus žaidimų kūrimo variklis.

Šiame darbe „Unity“ įrankį naudoju virtualios realybės akinių programėlei, skirtai išoriniam roboto manipulatoriaus valdymui, kurti.

2.1.4. „Oculus Quest 2“ virtualios realybės akiniai

„Oculus Quest 2“ (4 pav.) yra „Facebook Technologies“ sukurti virtualios realybės (VR) akiniai su dviem valdymo pultais, išleisti 2020 m. spalio mėn. Quest 2 VR akinių veikimas yra „savarankiškas“, t. y. jiems veikti nereikia kompiuterio ar žaidimų konsolės.

Šiame darbe VR akinius naudoju roboto manipuliatoriui valdyti HTTP protokolu.

Akinių valdikliai turi įprastiems žaidimų konsolių valdikliams būdingus mygtukus, kurie yra jautrūs lietimui, taip pat infraraudonosios šviesos diodais, akcelerometrais bei giroskopais pagrįstą žmogaus rankų ir pirštų judesių sekimą, aplinkoje esančių kliūčių fiksavimą.



4 pav. „Oculus Quest 2“ VR akiniai su valdikliais.

Pagrindinės VR akinių specifikacijos:

ekranas: IPS LCD, 1832x1920 skiriamoji geba, 90Hz ekrano atnaujinimo dažnis, apytiksliai 90 laipsnių matymo laukas;

programinė įranga: „Oculus Mobile“, pagrįsta „Android 10“ platforma, palaiko kompiuterinius VR žaidimus per „Oculus link“ ir trečiųjų šalių belaidžius sprendimus;

sekimas: palaikomas 6 laisvės laipsnių galvos ir rankų sekimas naudojant integruotą „Oculus Insight“ technologiją, 4 priekinės kameros vizualiai sekti valdiklius, taip pat girokopai ir akcelerometrai judesiems ir aplinkai fiksuoti;

garsas: įmontuoti stereogarsiakalbiai ir mikrofonas, 3,5 mm garso lizdas, 3D garso palaikymas;

ryšiai: „Wi-Fi“, „Bluetooth 5.0 LE“, C tipo USB prievadas su „USB Host“ palaikymu.

Nors pagrindinė šių VR akinių paskirtis yra virtualios realybės atvaizdavimas, kadangi akiniai turi 4 priekines kameras, jie gali būti naudojami ir papildytai realybei atvaizduoti. Akiniuose naudojamos infraraudonųjų spindulių kameros, matomas vaizdas yra nespaltotas, skiriamoji geba maža ir vaizdo kokybė prasta („Oculus“ viešai neatskleidžia šių kamerų specifikacijų). Kadangi kamerų tikslas yra tik sekti naudotojo judesius ir aplinką, aukštos kokybės vaizdas nėra reikalingas. Tačiau dėl galimybės naudoti akinius papildytai realybei, nemažai kūrėjų ir vartotojų naudojami šia papildoma funkcija. Aprašomame darbe „Oculus Quest 2“ VR akiniai taip pat naudojami būtent papildytai realybei atvaizduoti. Tuo siekiama pabrėžti naudojamos įrangos universalumą ir įsitikinti, ar roboto manipuliatoriui valdyti papildytoje realybėje reikalingas aukštos kokybės vaizdas.

2.2. VR programėlės kūrimas

Šioje darbo dalyje aprašomas VR akinių programėlės kūrimas roboto manipuliatoriui valdyti. Galutiniame rezultate siekiama palyginti roboto manipulatoriaus valdymą naudojant standartinį valdymo pultą ir sukurtą išorinę VR akinių programėlę.

VR akinių programėlės kūrimas buvo atliekamas naudojant „Unity“ žaidimų kūrimo variklį. Ši aplinka pasirinkta dėl plačios įrankių ir funkcijų bibliotekos, bei patogios grafinės sąsajos VR aplinkai kurti. VR akinių programėlės komunikacijai su roboto valdikliu realizuoti naudojama aplikacijų programavimo sąsaja, vadinama Robotų Tinklo Paslaugomis (angl. *Robot Web Services (RWS)*). RWS yra laisvai ir nemokamai prieinamas ABB įrankis. Šis įrankis naudoja HTTP komunikacijos protokolą.

2.2.1. Robotų tinklo paslaugos (angl. Robot Web Services (RWS))

RWS įrankis sukurtas pagal REST (angl. *REpresentational State Transfer*) architektūrinį stilių [19]. REST yra plačiai naudojamas architektūrinis stilius kuriant tinklo taikomųjų programų sąsajas. RWS kaip taikomąjį protokolą naudoja HTTP. HTTP pagrindinės sudedamosios dalys yra URL ir metodai. URL nurodo tinklalapį (pavyzdžiui, www.google.com) arba I/O signalą (pavyzdžiui, 127.0.0.1/rw/iosystem/signals/DO1). REST sistemoje URL identifikuoja išteklių. Iš roboto valdiklio siunčiamų taikomosios programos duomenų atvaizdavimas gali būti XML arba JSON formatu.

URL gali turėti užklauso parametru, kurie žymimi simboliu '?'. Pavyzdžiui, dauguma RWS išteklių palaiko šiuos užklauso parametrus:

- json=1; grąžinamas JSON atvaizdavimas. Pagal numatytuosius nustatymus visada pateikiamas XML formatas;
- debug=1; grąžinama daugiau informacijos, padedančios derinti programas.

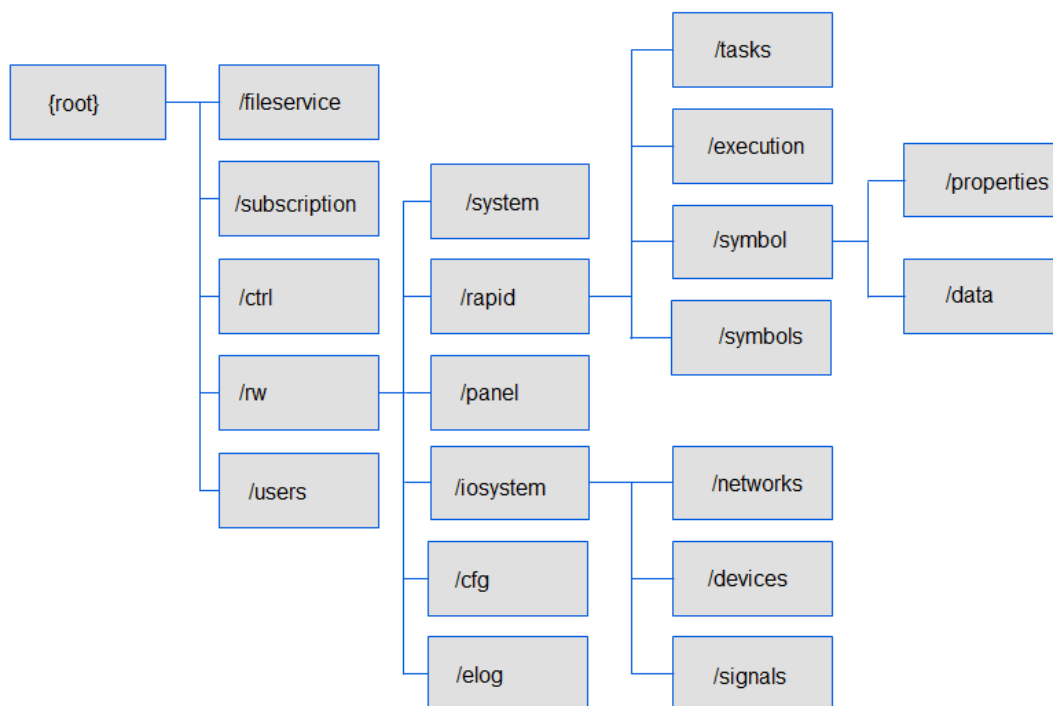
URL (127.0.0.1/rw/iosystem?json=1) prašoma pateikti šį išteklių JSON formatu. HTTP metodas apibrėžia, kokia užduotis turi būti vykdoma su ištekliu. Kiekvienas išteklius gali palaikyti vieną ar daugiau HTTP metodų. HTTP turi keletą iš anksto apibrėžtų metodų. Svarbiausi yra šie:

- GET: išteklių gavimas;
- PUT: sukurti arba atnaujinti išteklių;
- POST: atnaujinti išteklių;
- DELETE: išteklių ištrynimasis.

GET nekeičia ištekliaus būsenos, o DELETE, PUT ir POST keičia ištekliaus būseną. RWS pateikia žiniatinklio API rinkinį, kurį gali naudoti bet kuris HTTP žinantis klientas, naudojantis bet kurią programavimo kalbą. API grąžina duomenis XML arba JSON pavidalu, kuriuos galima analizuoti naudojant standartinius XML/JSON analizatorius. Paprastas būdas patikrinti prieinamą roboto žiniatinklio paslaugą - naudoti žiniatinklio naršyklę ir adreso lauke įvesti roboto valdiklio URL. Norint naudotis REST API, kliento programa turėtų pateikti HTTP užklausą ir analizuoti atsakymą.

Iš klientų nereikalaujama apklausti ištekliaus būsenos pokyčių, būsenos pokyčiai klientui siunčiami kaip įvykiai. RWS palaiko žiniatinklio lizdų protokolą (angl. *WebSockets*). Klientai pirmiausia turi užsiprenumeruoti pokyčius, o tada pokyčiai per Websockets siunčiami abonentams.

RWS sudaro kelios paslaugos, o kiekviena paslauga gali turėti papildomų paslaugų arba vieną ar daugiau ištekliaus. RWS galimų paslaugų ir ištekliaus schema vaizduojama 5 pav.



5 pav. RWS galimų paslaugų ir ištekliaus schema [20].

Failų paslaugos (angl. *File Service (/fileservice)*): suteikia nuotolinę prieigą prie failų ir katalogų. Tvarko failų ir katalogų perkėlimą, kūrimą, šalinimą ir pervardijimą.

Prenumeratos paslaugos (angl. *Subscription Service (/subscription)*): tvarko ištekliaus prenumeratą ir siunčia įvykius, kai prenumeruojami ištekliai atnaujinami. Prenumeratos paslauga įvykiams naudoja „WebSockets“.

Valdiklio paslaugos (angl. *Controller Service (/ctrl)*): tvarko roboto valdiklio bendrąsias funkcijas, pavyzdžiui, prieigą prie valdiklio vidinio laikrodžio, valdiklio identifikavimą, atlieka pakartotinį paleidimą ir t.t.

Vartotojų paslaugos (angl. *Users Service (/users)*): tvarko prijungtų klientų registraciją.

„RobotWare“ paslaugos (angl. *RobotWare Service (/rw)*): tvarko „RobotWare“ paslaugas, tokias kaip I/O, „RAPID“, įvykių žurnalas (angl. *Event Log*), konfigūracijas ir kt.

2.2.2. Roboto valdiklio konfigūravimas

Išorinei VR programėlei veikti reikalinga atlikti tam tikras roboto valdiklio konfigūracijas. Pirmiausia, valdiklio konfigūracijos nustatymų lange reikia įtraukti „PC Interface“, „FlexPendant Interface“ ir „Multitasking“ parinktis (6 pav.). Šie nustatymai yra būtini, kad būtų galima komunikuoti su roboto valdikliu naudojant RWS klientą.

HOME/user.sys		Configuration - Communication	Configuration - Controller X
Type	Name	Description	
Auto Condition Reset	Calaxis	Axis Calibration	
Automatic Loading of Modules	Calpend	Pendulum Calibration	
Cyclic Bool Settings	GtpuSDK	617-1 FlexPendant Interface	
Event Routine	MOTCOM	Motor Commutation	
General Rapid	multask	623-1 Multitasking	
ModPos Settings	PCINB	616-1 PC Interface	
Operator Safety	RemoteService	Remote Service	
Options	SIS	Service Info System	
Path Return Region			
Run Mode Settings			
Safety Runchain			
Task			

6 pav. Valdiklio konfigūracijos nustatymų langas „RobotStudio“ aplinkoje.

Roboto valdiklis turi keletą parametrų, kurie įtakoja pranešimų tarp VR ir „RAPID“ programėlių veikimą (1 lentelė). Sistemos parametras RmqType turi būti nustatytas į Remote, kad tarp „RAPID“ ir VR programėlių būtų galima perduoti pranešimus. Šis nustatymas pakeistas roboto valdymo pulte „FlexPendant“ atsidarius Control Panel -> Configuration -> Controller -> Task -> T_ROB1 (7 pav.). RmqMode parametras gali būti keičiamas priklausomai nuo pageidaujamo programėlės veikimo režimo (pertraukimo režimas arba sinchroninis). Kiti 1 lentelėje aprašomi parametrai keičiami „RobotStudio“ konfigūracijos failuose, tačiau rekomenduojama palikti numatytas vertes.

Control Panel - Configuration - Controller - Task - T_ROB1	
Name:	T_ROB1
Tap a parameter twice in order to modify it.	
Parameter Name	Value
 Task	T_ROB1
Main Entry	main
Check Unsolved References	1
Motion Task	Yes
Use Mechanical Unit Group	
RMQ Type	Remote

7 pav. RmqType parametro keitimas.

1 lentelė. Roboto valdiklio parametrai, pranešimams valdyti:

Parametras	Apibūdinimas
RmqType	• None (Nei vienas) – „RAPID“ pranešimų funkcionavimas išjungtas šioje „RAPID“ užduotyje. Tai yra numatytoji reikšmė. • Internal (Vidinis) – „RAPID“ pranešimų funkcionavimas įjungtas vietiniam naudojimui valdiklyje. • Remote (Nuotolinis) – „RAPID“ pranešimų funkcionavimas įjungtas vietiniam naudojimui ir kompiuterio bei „FlexPendant“ programėlėms.
RmqMode	• Pertraukimo režimas – pranešimą galima gauti prijungus skaitymo procedūrą prie nurodyto tipo pranešimo arba naudojant siuntimo-laukimo funkciją. Visi pranešimai, kurie nėra atsakymas į aktyvią siuntimo-laukimo instrukciją arba kurių tipas nėra prijungtas prie skaitymo procedūros, bus atmesti. Tai numatytasis režimas. • Sinchroninis režimas – visi pranešimai bus įrašyti į eilę ir gali būti gaunami tik naudojant naują skaitymo-laukimo instrukciją. Pranešimai nebus atmetami, nebent eilė bus pilna. Šiuo režimu negalima naudoti siuntimo-laukimo instrukcijos.
RmqMaxMsgSize	Didžiausias pranešimo duomenų dydis baitais. Numatytoji vertė yra 350. Šios reikšmės negalima keisti „RobotStudio“ arba „FlexPendant“ įrankyje.
RmqMaxNoOfMsg	Didžiausias pranešimų skaičius eilėje. Numatytoji reikšmė yra 5. Šios reikšmės negalima keisti „RobotStudio“ arba „FlexPendant“ įrankyje.

Atlikus aukščiau aprašytas konfigūracijas, kūriau „RAPID“ programinį kodą, kuriame realizuotas roboto manipulatoriaus judėjimas.

2.2.3. „RAPID“ programos veikimas

„RAPID“ programinis kodas susideda iš trijų užduočių.

Pirmoje užduotyje sukurtos funkcijos, kurios pradeda arba stabdo roboto manipulatoriaus judėjimą. Realizuotas tiek atskirų grandžių (galima judinti kiekvieną grandį atskirai), tiek linijinis judėjimas (robotas juda Dekarto koordinačių sistemoje išlaikydamas galinio darbo įtaiso pasisukimą). Funkcijų veikimą valdo skirtingi kintamieji, nurodantys kokį judėjimą (grandžių ar linijinį), kurios roboto grandies ir kuria kryptimi pradėti. Judėjimo stabdymas realizuotas naudojant „RAPID“ TRAP procedūrą, skirtą apdoroti pertraukimams. TRAP procedūros „RAPID“ programiniame kode susiejamos su konkrečiu pertraukimu ir, jei tas pertraukimas įvyksta, automatiškai įvykdomas TRAP procedūrai priskirtas kodas. Programoje niekada negalima tiesiogiai iškviesti TRAP procedūros. Šiuo atveju, roboto judėjimas stabdomas pasikeitus tam tikram kintamajam arba dingus ryšiui su išorine roboto valdymo programa (įvykus pertraukimui). Išorinė VR akinių programa keičia minėtų kintamųjų dydžius ir taip reguliuoja kurią „RAPID“ programinio kodo dalį vykdyti arba kokį roboto judėjimą pradėti/stabdyti.

Antra užduotis susideda iš dviejų dalių. Pirmoje dalyje atliekamas ryšio su išorine valdymo programa tikrinimas. Išorinė programa prisijungus prie roboto valdiklio nustatytu pastoviu laiko žingsniu

siunčia roboto valdikliui signalą (pakeičia kintamąjį), o valdiklio „RAPID“ programoje skaičiuojamas laikas nuo paskutinio signalo priėmimo. Nustatytą laiko tarpą negavus signalo yra stabdomas „RAPID“ veikimas taip užtikrinant išorinės VR akinių programėlės saugumą. Antra dalis kiekvieną užduoties vykdymo ciklą perskaičiuoja roboto grandžių kampus ir galinio įtaiso koordinatas, kad šiuos duomenis realiu laiku būtų galima atvaizduoti išorinėje VR akinių programėlėje ir panaudoti kituose skaičiavimuose.

Trečioje „RAPID“ programos užduotyje realizuotas ciklas, skaičiuojantis tolimiausias koordinatas, kurias roboto galinis įtaisas gali pasiekti. Taip išvengiama valdiklio klaidų, kai bandoma pozicionuoti robotą jam nepasiekiamame taške.

2.2.4. VR akinių programėlės modeliavimas ir veikimas

VR akinių programėlė kurtą naudojant „Unity“ programinį paketą. Pirmiausia, atliekant „Unity“ projekto konfigūraciją, buvo nurodyta platforma (šiuo atveju „Oculus“ VR akiniai veikia „Android“ platformoje), kuriai bus kuriama programėlė ir pasirinkta VR akinių procesoriaus architektūra (ARM64). Kiti konfigūracijos parametrai sudaro VR aplinkos atvaizdavimo nustatymus, pavyzdžiui: spalvų erdvė – linijinė, spalvų gama – sRGB, ir pan.

VR programėlės veikimui reikalinga atlikti dar kelis žingsnius, susijusius su tinklo prieigos leidimais. Pirmiausia atlikau „AndroidManifest“ XML failo redagavimą. Šiame faile saugomi svarbūs „Android“ programėlės metaduomenys, apimantys versijos palaikymą, aparatinės įrangos funkcijų palaikymą, leidimus ir kitus parametrus. Įprastai šis failas yra konfigūruojamas „Unity“ paketo nustatymuose, tačiau kai kurie parametrai gali būti pakeisti tik rankiniu būdu. Šiuo atveju rankiniu būdu buvo pridėti „Android“ programėlės leidimai prisijungti prie tinklo, reikalingi HTTP komunikacijai veikti (8 pav.).

```
<?xml version="1.0" encoding="utf-8"?>
<manifest
  xmlns:android="http://schemas.android.com/apk/res/android"
  package="com.unity3d.player"
  xmlns:tools="http://schemas.android.com/tools">
  <uses-permission android:name="android.permission.INTERNET"/>
  <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
  <application>
    <activity android:name="com.unity3d.player.UnityPlayerActivity"
      android:theme="@style/UnityThemeSelector">
      <intent-filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
      </intent-filter>
      <meta-data android:name="unityplayer.UnityActivity" android:value="true" />
    </activity>
  </application>
</manifest>
```

8 pav. „AndroidManifest“ XML failo parametrai.

Antras žingsnis VR programėlei veikti – roboto valdiklio konfigūracijos XML failo redagavimas. Šio failo parametrai gali būti redaguojami taip pat tik rankiniu būdu. Parametrai vaizduojami 9 pav. Faile nurodomi IP adresai, kuriems leidžiama prisijungti prie virtualaus roboto valdiklio. Iš viso šiame faile galima įtraukti iki dešimties priimamų IP adresų.

```
<?xml version="1.0" encoding="UTF-8" ?>
<VCConfiguration>
  <RemoteVCConfiguration PublicationEnabled="true" />
  <hosts>
    <host ip="192.168.8.100"/>
    <host ip="192.168.8.101"/>
    <host ip="192.168.8.102"/>
    <host ip="192.168.8.103"/>
    <host ip="192.168.8.104"/>
    <host ip="192.168.8.105"/>
    <host ip="192.168.8.106"/>
    <host ip="192.168.8.107"/>
  </hosts>
</VCConfiguration>
```

9 pav. Virtualaus valdiklio konfigūracijų failo duomenys.

Galiausiai, svarbu aprašyti HTTP kliento autentifikaciją. ABB roboto valdiklis palaiko tik vieną autentifikacijos metodą – „Digest“. Įprasti autentifikacijos metodai turi kelis parametrus: prisijungimo vardas, slaptažodis, serverio adresas ir naudoja lengvai konvertuojamą Base64 kodavimą, kuris nėra saugus. „Digest“ nuo įprastų metodų skiriasi tuo, kad turi daugiau parametrų, reikalingų klientui prisijungti prie serverio: prisijungimo vardas, slaptažodis, serverio adresas, „nonce“ ir „realm“ parametrai. „Digest“ autentifikacija prieš siunčiant vartotojo vardą ir slaptažodį tinklu, duomenims taiko maišos (angl. *hash*) funkciją. Techniniu požiūriu, „Digest“ autentifikacija yra MD5 kriptografinio šifravimo taikymas, kai, siekiant išvengti duomenų atkūrimo atakų, naudojamos „nonce“ reikšmės.

„Unity“ programinis įrankis HTTP klientui kurti naudoja savo metodų biblioteką vadinamą „UnityWebRequest“, kuri neturi „Digest“ autentifikacijos palaikymo. Dėl šios priežasties, kliento autentifikacijos funkcijas reikėjo sukurti rankiniu būdu. Sukurtų funkcijų programinis kodas vaizduojamas 10 pav. Paveiksle raudonai pažymėtos eilutės atspindi HTTP kliento prisijungimo vardo, slaptažodžio ir serverio adreso užšifravimą tarp „nonce“ ir „realm“ parametrų reikšmių. „Realm“ parametro reikšmė ABB roboto valdiklyje yra pastovi, o „nonce“ – kiekvieną kartą perkrovus valdiklį pasikeičia. Parametrų reikšmės galima gauti siunčiant roboto valdikliui HTTP užklausą be autentifikacijos. Tokiu atveju yra gaunamas valdiklio atsakymo pranešimas su klaidos kodu „401 Unauthorized“ ir įvairiais komunikacijos parametrais, tarp kurių yra ir „realm“ bei „nonce“ reikšmės. 11 pav. pateikiamas tokio HTTP serverio atsakymo pavyzdys. Šiuo atveju pranešimas gautas siunčiant roboto valdikliui HTTP pranešimą su prisijungimo užklausa nenurodant autentifikacijos ir kitų parametrų (<http://localhost/>). Pranešimui siųsti naudotas programinis įrankis „Postman“.

```

string GenerateDigestHeader(string username, string password, string method, string uri)
{
    string nonce = "MzExMjhhkZDYzYzBkZGUwNzpzYWxpZHVzZXJzQHJvYmFwaS5hYmI6MThmN2M4ZWUxODM6NA==";
    string realm = "validusers@robapi.abb";

    string ha1 = CalculateMD5Hash(username + ":" + realm + ":" + password);
    string ha2 = CalculateMD5Hash(method + ":" + uri);
    string response = CalculateMD5Hash(ha1 + ":" + nonce + ":" + ha2);

    string authHeader = "Digest username=\"\" + username + "\", realm=\"\" + realm + "\", nonce=\"\"
    + nonce + "\", uri=\"\" + uri + "\", response=\"\" + response + "\"";

    return authHeader;
}

string CalculateMD5Hash(string input)
{
    using (MD5 md5 = MD5.Create())
    {
        byte[] inputBytes = Encoding.UTF8.GetBytes(input);
        byte[] hashBytes = md5.ComputeHash(inputBytes);

        StringBuilder sb = new StringBuilder();
        for (int i = 0; i < hashBytes.Length; i++)
        {
            sb.Append(hashBytes[i].ToString("x2"));
        }
        return sb.ToString();
    }
}

```

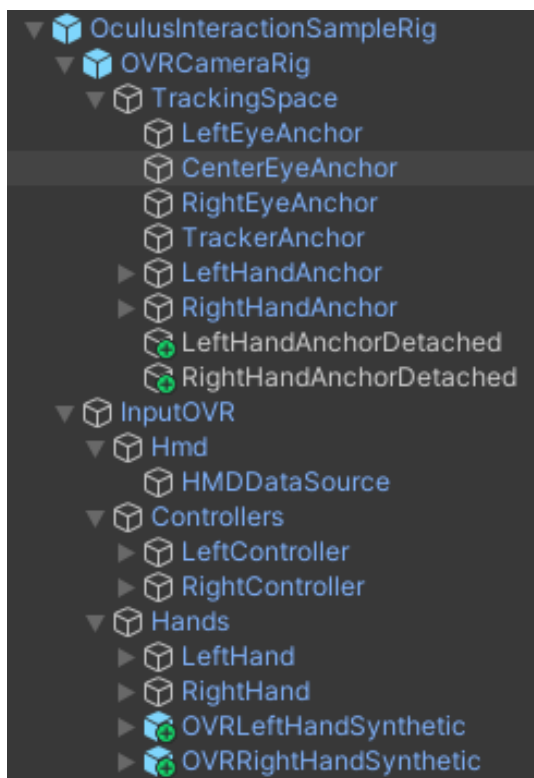
10 pav. HTTP kliento funkcijos autentifikacijai realizuoti.

X-Frame-Options	①	SAMEORIGIN
Content-Type	①	text/plain
X-Content-Type-Options	①	nosniff
Date	①	Wed, 15 May 2024 15:29:06 GMT
Cache-Control	①	no-cache
Content-Length	①	29
X-XSS-Protection	①	1; mode=block
Connection	①	Keep-Alive
WWW-Authenticate	①	Digest realm="validusers@robapi.abb", domain="/", qop="auth", nonce="MzExMjhhkZDYzYzBkZGUwNzpzYWxpZHVzZXJzQHJvYmFwaS5hYmI6MThmN2M4ZWUxODM6NA==", opaque="799d5", algorithm="MD5", stale="FALSE"
Accept-Ranges	①	

11 pav. Roboto valdiklio atsakymas į HTTP užklausą be autentifikacijos.

VR programėlės grafinėi sąsajai kurti „Unity“ pakete buvo naudojamas „Oculus“ integracijos SDK. Panašiai kaip ir RWS, šis įrankis turi tiksliniams poreikiams skirtą įrankių, bibliotekų ir dokumentų rinkinį. VR programėlėse „Oculus“ integracijos SDK apima funkcijas kaip galvos sekimas, rankų sekimas, erdvinis garsas, įvesties atvaizdavimas, platformos integracija ir kiti komponentai. Įrankis palaiko populiarius žaidimų variklius, tokius kaip „Unity“ ir „Unreal Engine“.

Naudojant „Oculus“ integracijos SDK „Unity“ projekte buvo pridėtas „Oculus“ integracijos komponentas (12 pav.) (angl. *Oculus Interaction Sample Rig*), kurį sudaro keli pagrindiniai įrenginiai: kameros įrenginys (angl. *OVR Camera Rig*) ir įvesties įrenginys (angl. *Input OVR*).

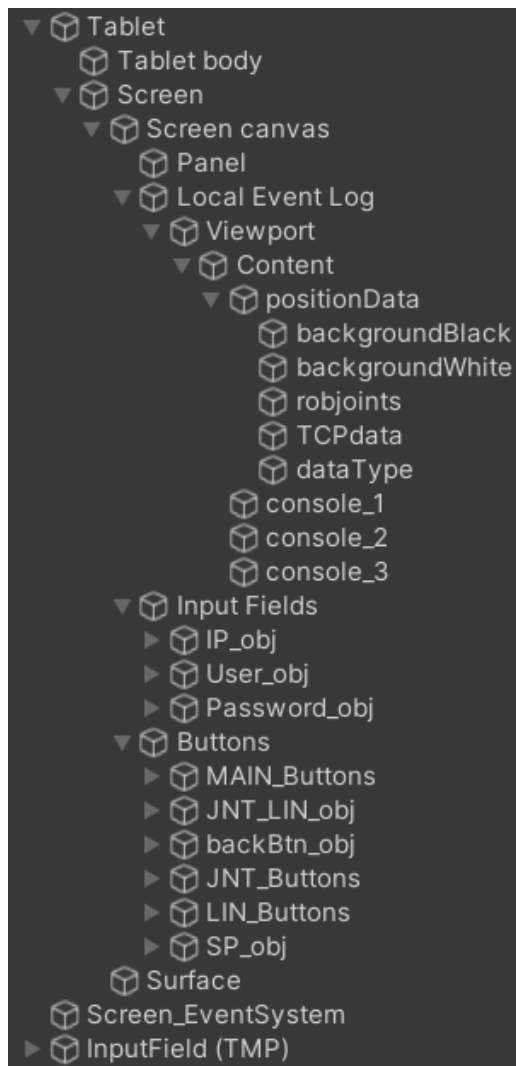


12 pav. „Oculus“ integracijos komponento sandara.

Kameros įrenginį sudaro komponentai, skirti vartotojo galvai, rankoms ir VR akinių valdikliams sekti, aplinkai atvaizduoti. Įrenginio nustatymuose buvo įjungtas perėjimo (angl. *Passthrough*) parametras. Šis parametras aktyvuoja „Oculus“ VR akinių išorines kameras, kurios fiksuoja ir akiniuose atvaizduoja realios aplinkos vaizdą. Tokiu būdu VR akiniai gali būti naudojami papildytai realybei. Kitus kameros įrenginyje esančius komponentus sudaro grafiniai modeliai, objektų atpažinimo algoritmai bei įvairios animacijos. Jų pagalba akiniai atpažįsta vartotojo rankas, rankų judesius, akinių valdiklius, galvos judesius. Tai leidžia vartotojams virtualiai sąveikauti su aplinka bei sumodeliuotais virtualiais objektais.

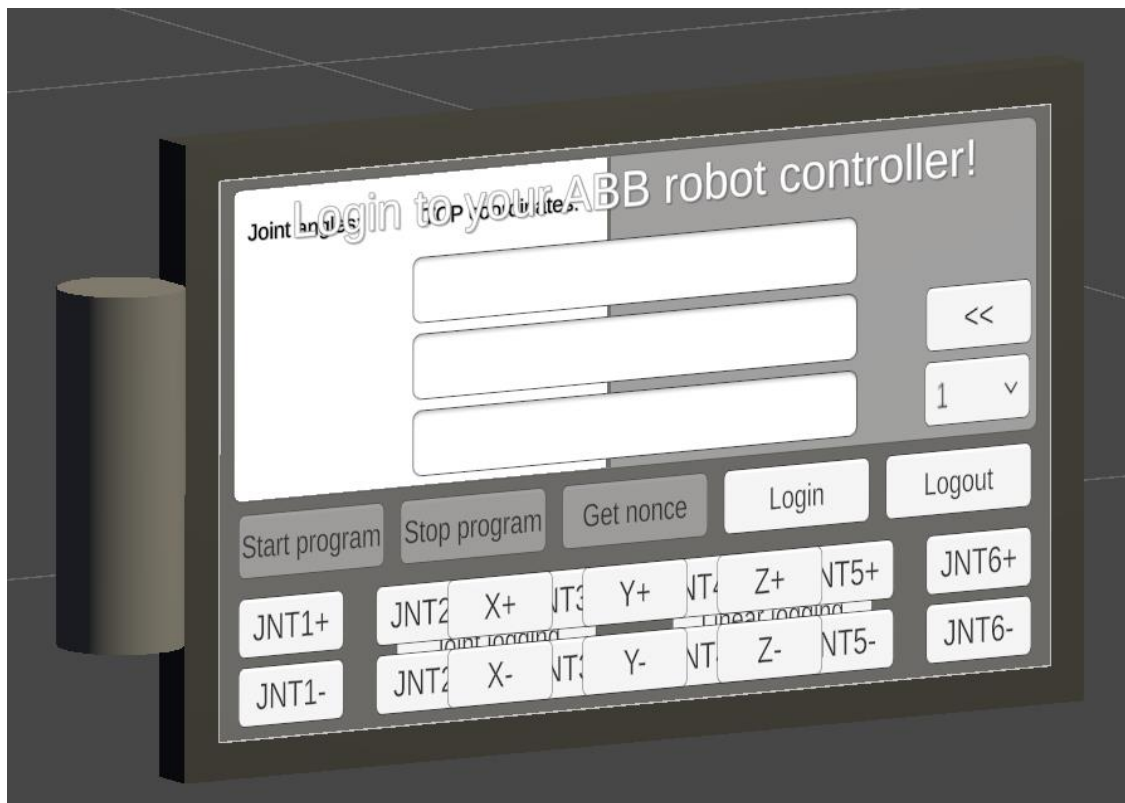
Vartotojo sąveika su virtualia aplinka yra realizuojama įvesties įrenginyje. Tai programinės įrangos komponentas, leidžiantis kūrėjams tvarkyti ir apdoroti įvesties įvykius iš „Oculus“ valdiklių ir kitų palaikomų įvesties įrenginių, kaip, pavyzdžiui, vartotojo rankų. Įvesties įrenginyje pateikiama sąsaja, leidžianti sąveikauti su įvesties sistema ir pasiekti bei atpažinti įvesties duomenis, pavyzdžiui, mygtukų paspaudimus, lietimui jautraus ekrano pozicijas, valdiklių judesius, rankų ar valdiklių gestus, kuriuos atlieka vartotojas ir pan.

Atlikus „Oculus“ integracijos komponento konfigūravimą buvo kuriama virtuali planšetė, kurios pagalba bus atliekama komunikacija su roboto valdikliu. Planšetės komponento sandara vaizduojama 13 pav.

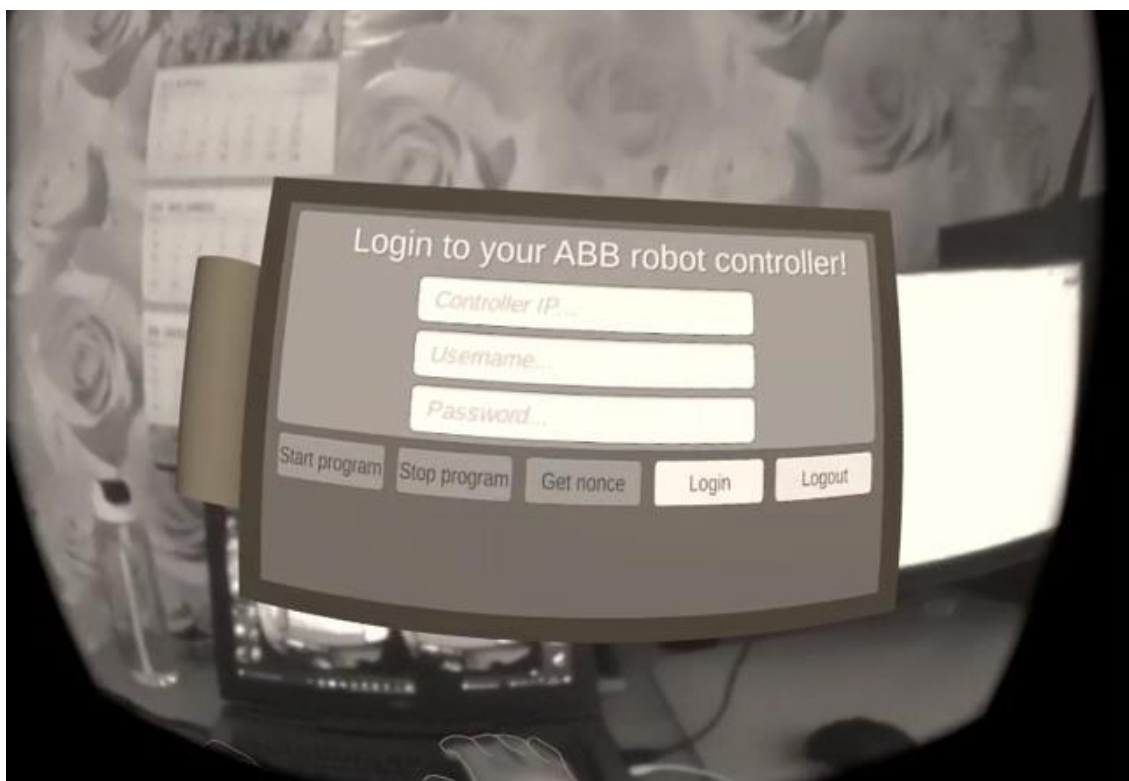


13 pav. Virtualios planšetės komponento sandara.

Virtualią planšetę sudaro ekranas, kuriame atvaizduojami HTTP užklausų atsakymai, roboto valdiklio ar kita pageidaujama informacija. Planšetė taip pat turi virtualius mygtukus, kurių paspaudimais roboto valdikliui siunčiami HTTP pranešimai ir taip realizuojamas roboto manipulatoriaus valdymas: prisijungiama prie valdiklio, paleidžiama valdiklio „RAPID“ programa, „RAPID“ programoje įrašomi pageidaujami kintamieji ir pan. Planšetės modelis vaizduojamas 14 pav. ir 15 pav.



14 pav. Virtualios planšetės modelis „Unity“ aplinkoje.



15 pav. Virtualios planšetės vaizdas VR akinių programėlėje.

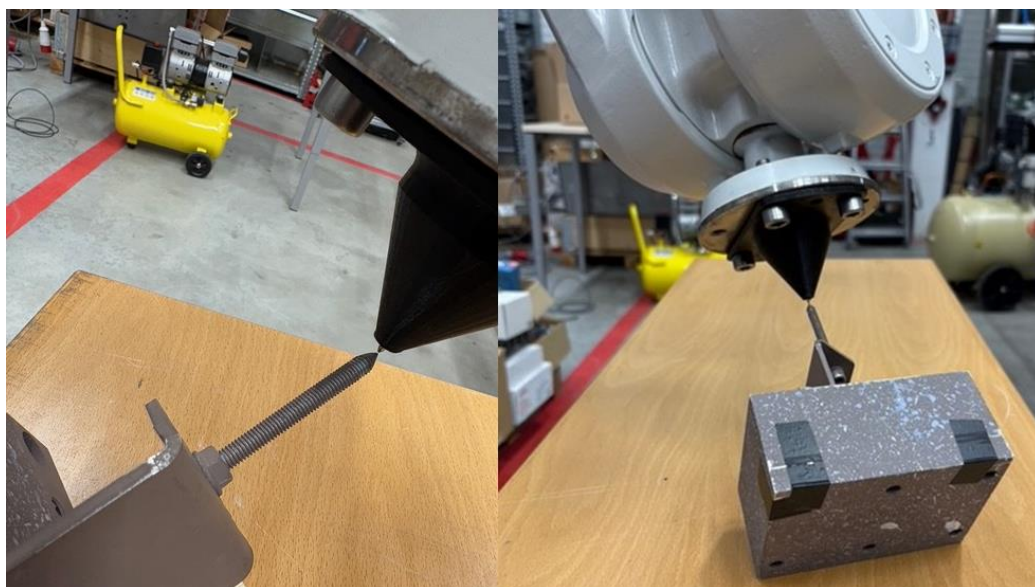
Virtuali planšetė sukonfigūruota taip, kad galėtų sąveikauti su VR akinių valdikliais ir vartotojo rankomis. Vartotojas rankomis, sugriebimo (angl. *Grab*) judesiu, arba valdikliais gali paimti planšetę ir nešioti kaip realioje aplinkoje, planšetės mygtukai yra jautrūs valdiklių mygtukų ir žmogaus rankos pirštų paspaudimams.

3. VR programėlės tyrimas

Sukurtos programėlės veikimas buvo testuotas atliekant pozicionavimo tyrimus. Vykdomos įvairios pozicionavimo užduotys valdant roboto ranką standartiniu valdymo pultu (Teach Pendant) bei VR akinių programėle. Užduotys susideda iš: atsitiktinių taškų pozicionavimo, galinio darbo įtaiso kalibravimo bei atsitiktinio atstumo poslinkio. Galiausiai lyginami VR akinių programėlės ir standartinio valdymo pulto pozicionavimo rezultatai, skaičiuojami galinio darbo įtaiso koordinatčių nuokrypiai, kalibravimo tikslumas. Kaip atskaitos taškas imamos roboto koordinatės valdant standartiniu valdymo pultu.

3.1.1. Atsitiktinių taškų pozicionavimas

Atsitiktinių taškų pozicionavimo tyrimas atliktas pozicionuojant roboto galinį darbo įtaisą skirtinguose taškuose kaip parodyta 16 pav. Kiekviename pasirinktame taške galinis įtaisas pozicionuojamas valdant robotą standartiniu valdymo pultu ir naudojant VR akinių programėlę. Tyrime naudotas linijinis roboto valdymas (angl. Linear jogging).



16 pav. Roboto galinio darbo įtaiso pozicionavimo atsitiktiniuose taškuose pavyzdys.

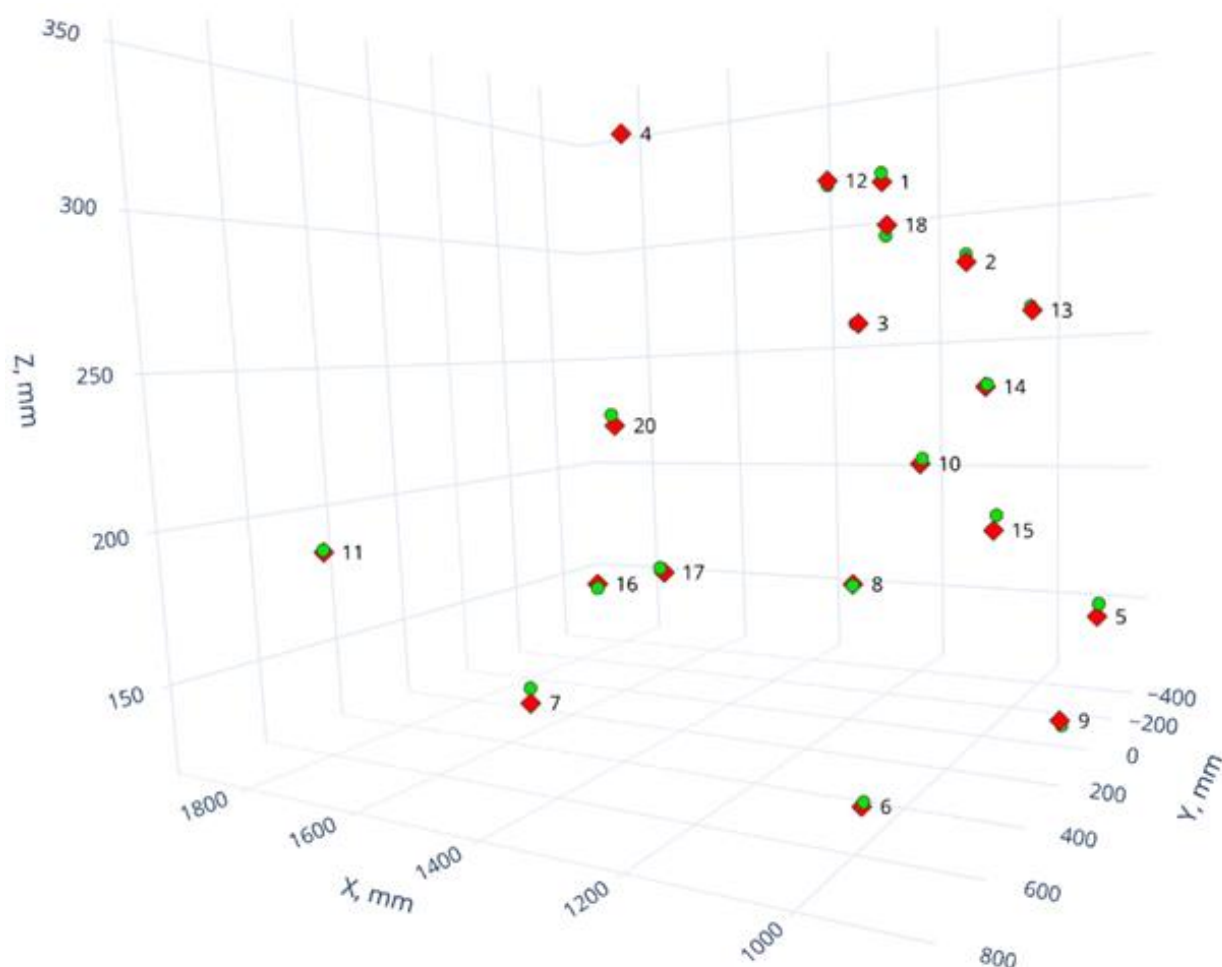
Šiame tyrime buvo vertinamos tik roboto galinio darbo įtaiso koordinatės, neįtraukiant galinio įtaiso pasisukimo erdvėje. Iš viso paimta 20 tokių pozicionavimo taškų. 2 lentelėje aprašytos kiekvieno išmatuoto taško koordinatės X, Y ir Z milimetrais.

2 lentelė. Roboto galinio darbo įtaiso pozicionavimo atsitiktiniuose taškuose koordinatės.

Taško numeris	Teach Pendant valdymas			VR programėlės valdymas		
	X, mm	Y, mm	Z, mm	X, mm	Y, mm	Z, mm
1.	1144.97	-32.39	302.46	1146.28	-33.26	305.36
2.	1070.31	-218.81	276.83	1071.44	-221.53	279.49
3.	976.94	625.42	253.55	980.95	623.35	253.57
4.	1644.79	-164.81	336.75	1646.12	-166.88	337.23
5.	930.32	-524.15	142.87	926.08	-513.18	148.39
6.	936.15	772.47	146.32	933.07	776.46	147.75
7.	1331.53	770.75	159.31	1330.43	771.49	163.38
8.	1327.12	-426.62	152.96	1327.03	-423.68	152.41

Taško numeris	Teach Pendant valdymas			VR programėlės valdymas		
	X, mm	Y, mm	Z, mm	X, mm	Y, mm	Z, mm
9.	891.57	12.81	129.54	889.75	9.22	127.88
10.	1028.12	208.91	213.62	1026.02	206.73	215.14
11.	1911.84	456.24	184.37	1912.34	457.61	185.24
12.	1267.56	-151.67	308.31	1268.91	-154.13	306.83
13.	1001.23	-356.13	259.66	1004.29	-358.66	261.22
14.	987.54	21.08	234.86	985.81	20.11	235.57
15.	913.83	305.65	197.11	910.24	306.98	201.36
16.	1562.32	157.22	168.28	1561.11	159.75	166.85
17.	1422.15	194.62	175.52	1426.84	200.14	177.13
18.	1152.11	-79.21	289.37	1153.68	-75.98	285.86
19.	1394.09	562.55	364.25	1394.83	566.19	364.97
20.	1481.46	227.37	226.21	1486.12	228.83	229.89

17 pav. vaizduojamas 2 lentelėje aprašytų koordinatų grafikas 3D erdvėje. Grafiką galima peržiūrėti interaktyviai ir įvertinti kiekvieną individualų koordinatų tašką paspaudus šią nuorodą: <https://chart-studio.plotly.com/~rokas850/2.embed>.



17 pav. 2 lentelėje aprašytų pozicionavimo taškų išsidėstymas. Raudonas žymeklis – Teach Pendant valdymas, žalias žymeklis – VR programėlės valdymas.

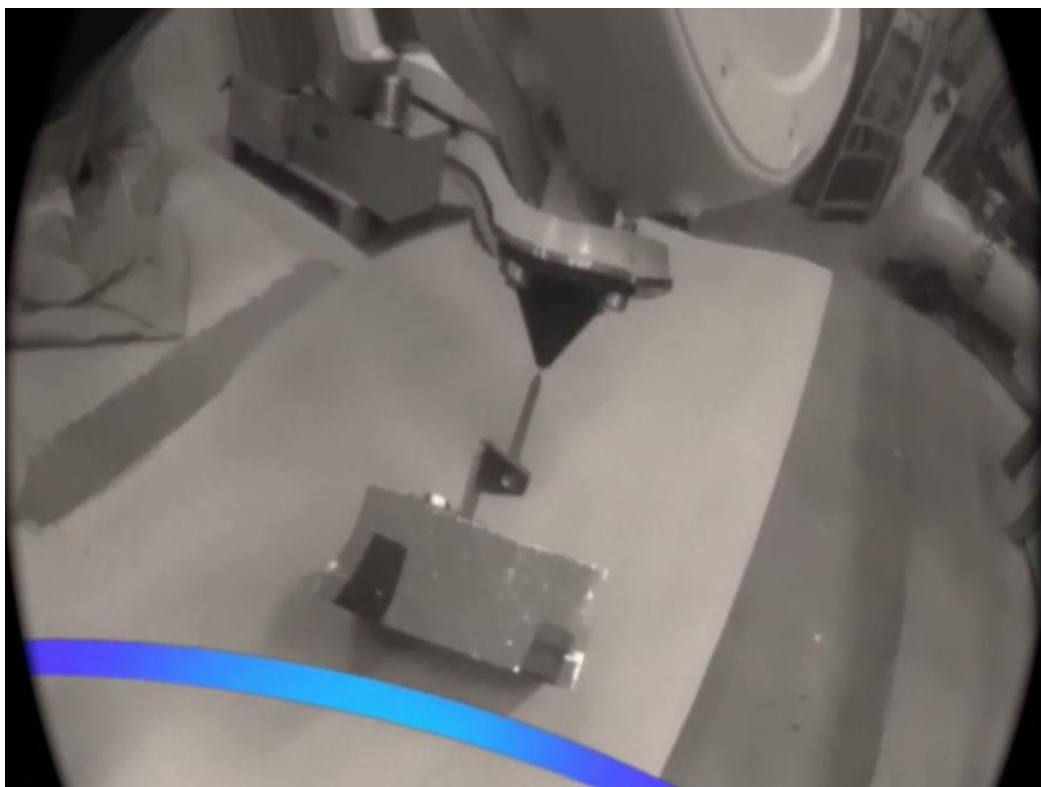
Naudojant gautus koordinačių duomenis paskaičiuoti kiekvieno taško nuokrypiai, nuokrypių vektorių ilgiai ir nuokrypių vidurkiai, aprašyti 3 lentelėje.

3 lentelė. Roboto galinio darbo įtaiso pozicionavimo atsitiktiniuose taškuose koordinačių nuokrypiai.

Taško numeris	X, mm	Y, mm	Z, mm	Nuokrypio vektoriaus ilgis, mm
1.	1.31	0.87	2.9	3.30
2.	1.13	2.72	2.66	3.97
3.	4.01	2.07	0.02	4.51
4.	1.33	2.07	0.48	2.51
5.	4.24	10.97	5.52	12.99
6.	3.08	3.99	1.43	5.24
7.	1.1	0.74	4.07	4.28
8.	0.09	2.94	0.55	2.99
9.	1.82	3.59	1.66	4.35
10.	2.1	2.18	1.52	3.39
11.	0.5	1.37	0.87	1.70
12.	1.35	2.46	1.48	3.17
13.	3.06	2.53	1.56	4.27
14.	1.73	0.97	0.71	2.11
15.	3.59	1.33	4.25	5.72
16.	1.21	2.53	1.43	3.15
17.	4.69	5.52	1.61	7.42
18.	1.57	3.23	3.51	5.02
19.	0.74	3.64	0.72	3.78
20.	4.66	1.46	3.68	6.11
Nuokrypių vidurkis	2.1655	2.859	2.0315	4.50
Didžiausias nuokrypis	4.69	10.97	5.52	12.99
Mažiausias nuokrypis	0.09	0.74	0.02	1.70

3 lentelėje matomi roboto galinio darbo įtaiso koordinačių skirtumai, kai robotas buvo pozicionuojamas tame pačiame taške naudojant standartinį valdymo pultą ir naudojant sukurta VR akinių programėlę. Didžiausio nuokrypio vektoriaus ilgis yra 12.99mm, mažiausio – 1.7mm, o nuokrypių vektorių ilgių vidurkis – 4.5mm. Pagrindinė priežastis, dėl kurios atsiranda aprašyti koordinačių skirtumai: VR akinių išorinių kamerų vaizdo kokybė. Valdant robotą standartiniu valdymo pultu, kiekvienas pozicionavimo taškas yra matomas aiškiai ir įvertinamas tiesiogiai žmogaus akies, be jokių tarpinių vaizdo perdavimo priemonių, panašiai kaip 16 pav. Tuo tarpu valdant robotą VR akinių programėle, kiekvienas pozicionavimo taškas matomas VR akiniuose (18 pav.), o vaizdą perduoda akinių išorinės kameros. Dėl nespaltvoto ir mažos skiriamosios gebos vaizdo, pozicionuojamas taškas matomas neaiškiai, todėl atsiranda netikslumai.

Šiuo atveju nebuvo atsižvelgta į objektų dydį, spalvą ar VR akinių modelį. Tikėtina, kad naudojant skirtingų dydžių ar spalvų objektus, kitokius virtualios ar papildytos realybės akinius, turinčius kokybiškesnes kameras, būtų galima išgauti tikslesnius rezultatus.



18 pav. Roboto galinio darbo įtaiso pozicionavimo vaizdas VR akiniuose.

3.1.2. Galinio darbo įtaiso kalibravimas

Roboto galinio darbo įtaiso kalibravimas reikalingas tam, kad robotas tiksliai galėtų valdyti įrankio judėjimą ir žinotų jo vietą erdvėje. Kalibravimui buvo pasirinktas TCP metodas. TCP (angl. Tool Center Point) yra taškas, žymintis roboto galinio darbo įtaiso vietą ir orientaciją roboto koordinatinių sistemoje. Kalibravimas atliekamas pasirinkus vieną pastovų tašką roboto koordinatinių sistemoje ir tame taške skirtingomis roboto rankos orientacijomis pozicionuojant galinį įtaisą bei pozicijas išsaugant kalibravimo lange (19 pav.).

Tool Frame Definition

Tool: **testTool**

Select a method, modify the positions and tap OK.

Method: **TCP (default orient.)** No. of points: **4**

Point	Status
Point 1	-
Point 2	-
Point 3	-
Point 4	-

Positions **Modify Position** **OK** **Cancel**

19 pav. Roboto galinio darbo įtaiso kalibravimo langas.

Dažniausiai naudojamas 4 arba 6 skirtingų roboto rankos orientacijų pozicionavimas. Kalibravimo tikslumui užtikrinti svarbu, kad kiekvienoje skirtingoje roboto orientacijoje kiekviena roboto grandis

turėtų kitokią pasisukimo kampą, todėl šiame tyrime buvo naudotas atskirų roboto grandžių valdymas (angl. Joint jogging). Atlikus kalibravimą, ABB valdiklis nustato galinio darbo įtaiso poslinkius X, Y ir Z ašyse ir tuo pačiu automatiškai apskaičiuoja kalibravimo tikslumą (20 pav.).

Calculation Result

Tool: testToolTP

Tap OK to confirm the result or Cancel to redefine the source data.

2 to 7 of 7

Max Error

0.4226618 mm

Min Error

0.2304419 mm

Mean Error

0.3672783 mm

X:

0.4585333 mm

Y:

-0.1404447 mm

Z:

54.70494 mm

OK

Cancel

20 pav. Roboto galinio darbo įtaiso kalibravimo rezultatų langas.

Tyrime naudoti 4 ir 6 roboto rankos orientacijų metodai, kiekvienas kalibravimo metodas atliktas po 10 kartų. Identiškomis sąlygomis tuose pačiuose taškuose kalibravimas atliktas valdant robotą standartiniu valdymo pultu ir VR akinių programėle. 4 lentelėje aprašyti gauti kalibravimo rezultatai.

4 lentelė. Roboto galinio darbo įtaiso kalibravimo rezultatai.

Kalibravimo metodas	Teach Pendant valdymas			Didžiausia paklaida, mm	Mažiausia paklaida, mm	Paklaidos vidurkis, mm
	Didžiausia paklaida, mm	Mažiausia paklaida, mm	Paklaidos vidurkis, mm			
TCP 4 roboto orientacijos	0.422	0.231	0.367	3.548	2.321	3.191
	0.398	0.199	0.298	3.429	2.331	2.881
	0.651	0.218	0.434	4.262	3.082	3.672
	0.433	0.209	0.321	3.666	1.997	2.831
	0.379	0.206	0.292	3.582	2.638	3.112
	0.402	0.217	0.309	4.128	3.036	3.582
	0.495	0.255	0.375	4.488	1.944	3.216
	0.599	0.283	0.441	4.854	2.177	3.515
	0.463	0.265	0.364	4.125	2.307	3.216
	0.481	0.226	0.353	3.877	2.161	3.019
Bendras vidurkis	0.472	0.231	0.355	3.995	2.399	3.223
TCP 6 roboto orientacijos	0.316	0.101	0.247	2.917	1.285	2.101
	0.304	0.184	0.244	3.917	1.227	2.572
	0.244	0.167	0.205	3.605	2.111	2.858
	0.254	0.107	0.181	2.966	2.457	2.711
	0.267	0.147	0.207	2.651	1.588	2.119
	0.343	0.121	0.232	2.386	1.332	1.859

	Teach Pendant valdymas			VR programėlės valdymas		
	Didžiausia paklaida, mm	Mažiausia paklaida, mm	Paklaidos vidurkis, mm	Didžiausia paklaida, mm	Mažiausia paklaida, mm	Paklaidos vidurkis, mm
	0.279	0.181	0.229	3.981	2.407	3.194
	0.306	0.163	0.234	2.077	1.272	1.674
	0.327	0.176	0.251	3.583	2.024	2.803
	0.269	0.109	0.189	2.549	2.183	2.366
Bendras vidurkis	0.291	0.145	0.222	3.063	1.788	2.425

Analizuojant 4 lentelę pastebima, kad TCP 6 roboto rankos orientacijų metodas visais atvejais buvo tikslesnis už TCP 4 roboto rankos orientacijų metodą (tiek valdant robotą standartiniu valdymo pultu, tiek VR akinių programėle gautos mažesnės paklaidos). Lyginant skirtingus roboto rankos valdymo būdus, standartinio valdymo pulto naudojimas abiem kalibravimo metodais buvo tikslesnis už VR akinių programėlės naudojimą. Valdant robotą Teach Pendant pultu TCP 4 kalibravimo metodo didžiausia paklaida – 0.651mm, mažiausia paklaida – 0.199mm, bendras paklaidų vidurkis – 0.355mm, tuo tarpu valdant VR akinių programėle didžiausia paklaida – 4.854mm, mažiausia paklaida – 1.944mm, bendras paklaidų vidurkis – 3.223mm. VR akinių programėlės bendras kalibravimo paklaidų vidurkis didesnis už Teach Pendant vidurkį per 2.868mm. Valdant robotą Teach Pendant pultu TCP 6 kalibravimo metodo didžiausia paklaida – 0.343mm, mažiausia paklaida – 0.101mm, bendras paklaidų vidurkis – 0.222mm, tuo tarpu valdant VR akinių programėle didžiausia paklaida – 3.981mm, mažiausia paklaida – 1.227mm, bendras paklaidų vidurkis – 2.425mm. Šiuo atveju VR akinių programėlės bendras kalibravimo paklaidų vidurkis didesnis už Teach Pendant vidurkį per 2.203mm.

Šiame tyrime buvo naudotas tik vienas roboto galinis darbo įtaisas, todėl nebuvo atsižvelgta į jo geometrinės formos sudėtingumą. VR akinių programėlės naudojimas galinio įtaiso kalibravimui turėjo vidutiniškai 2-3mm didesnę paklaidą nei naudojant standartinį valdymo pultą. Didelio tikslumo reikalaujančiose užduotyse šis skirtumas gali turėti reikšmingą įtaką.

3.1.3. Atsitiktinio atstumo poslinkis.

Atsitiktinio atstumo poslinkio tyrimas atliktas naudojant linijinį roboto rankos valdymą. Tyrimo užduotis pastumti roboto galinį darbo įtaisą vienoje koordinačių ašyje pasirinktą atstumą. Viso atlikta 10 eksperimentų su skirtingais atstumais. Galinio įtaiso poslinkiai išmatuoti matavimo juosta (rulete) (21 pav.), kurios matavimo paklaida nebuvo vertinama.



21 pav. Roboto galinio darbo įtaiso poslinkio matavimas.

Išmatuotų poslinkių duomenys aprašyti 5 lentelėje.

5 lentelė. Roboto galinio darbo įtaiso poslinkių tyrimo rezultatai.

Pasirinktas atstumas, mm	Roboto galinio darbo įtaiso poslinkis, mm	
	Teach Pendant valdymas	VR programėlės valdymas
100	100	100
200	200	200
300	300	300
400	400	400
500	500	500
600	600	600
700	700	700
800	800	800
900	900	900
1000	1000	1000

Šiuo atveju gauti identiški rezultatai robotą valdant tiek standartiniu valdymo pultu, tiek VR akinių programėle. Roboto galinio darbo įtaiso nueitas atstumas buvo stebimas vidinių „RAPID“ programos kintamųjų pagalba, t.y. „RAPID“ programoje galima realiu laiku stebėti galinio įtaiso koordinates, o atėmus pradinio taško koordinatę iš galinio taško koordinatės žinomas įtaiso poslinkis. Galinio įtaiso koordinačių atvaizdavimas, įskaitant ir galinio įtaiso bei kiekvienos roboto grandies pasisukimo kampą, buvo realizuotas ir VR akinių programėlėje (22 pav.), todėl valdant robotą abiem būdais buvo galima realiu laiku stebėti jo koordinates ir tuo pačiu galinio įtaiso poslinkį.



22 pav. Roboto koordinačių ir pasisukimo kampų atvaizdavimas virtualioje aplinkoje.

Taigi šio tyrimo atveju, kai yra žinomas robotui nueiti reikalingas atstumas, valdant robotą abiem būdais galima išgauti identišką tikslumą.

3.1.4. Tyrimų apibendrinimas

ABB roboto valdymo, naudojant virtualios realybės akinius, tikslumas buvo vertintas atliekant įvairias pozicionavimo užduotis ir gauti duomenys lyginti su roboto valdymu naudojant standartinį valdymo pultą. Dėl galimybės beveik realiu laiku stebėti roboto koordinates ir pasisukimo kampus VR akinių programėlėje (duomenų atnaujinimo dažnis priklauso nuo programų ciklo bei HTTP pranešimų laiko), vienodu tikslumu buvo galima stumdyti galinį įtaisą koordinatinių ašyse norimą atstumą, tiek valdant VR programėle, tiek standartiniu valdymo pultu. Atliekant pozicionavimo atsitiktiniuose taškuose tyrimą, roboto valdymas VR programėle turėjo vidutiniškai 4.5mm koordinatinių vektoriaus nuokrypį lyginant su valdymu naudojant standartinį valdymo pultą. Galinio darbo įtaiso kalibravimo užduotyse VR programėlės valdymas turėjo vidutiniškai 2-3mm didesnę kalibravimo paklaidą nei valdant robotą standartiniu valdymo pultu. Iš gautų rezultatų galima spręsti, kad didelio tikslumo reikalaujančioms užduotims (smulkios elektronikos ar optinių elementų surinkimas, chirurginė robotika, laboratoriniai tyrimai, preciziškas metalo apdirbimas) atlikti šis ABB roboto valdymo sprendimas nebūtų tinkamas. Tačiau mažesnio tikslumo užduotims su didesniu gabaritų objektais atlikti (objektų pakavimas, rūšiavimas), toks roboto valdymas galėtų būti naudingas. Pavyzdžiui norint nuotoliniu būdu atlikti robotizuotų sistemų stebėjimą ar derinimą. Šiuo atveju VR suteiktą galimybę virtualioje aplinkoje matyti ir valdyti robotizuotą sistemą fiziškai nebūnant šalia jos. Tai ypač pasiteisintų darbui pavojingose aplinkose (radiacija, cheminės medžiagos, sprogimo rizika).

Atliekant aprašytus tyrimus buvo išmatuotas ir vidutinis HTTP pranešimų vėlavimo laikas, kuris siekė 20/30ms. Valdant robotą VR programėle šis vėlavimas fiziškai nebuvo jaučiamas ir nebuvo atsižvelgta į jo įtaką tikslumo rezultatams. Šiuo atveju darbas atliktas tik sujungus ABB roboto valdiklį ir VR akinius į tą patį tinklą (vietinis tinklas, angl. Local Area Network (LAN)). Kadangi įprastai vietiniame tinkle esančių įrenginių komunikacija HTTP protokolu veikia sklandžiau ir greičiau nei komunikuojant globaliame tinkle (angl. Wide Area Network (WAN)), toliau plėtojant darbą reikėtų atsižvelgti ir į valdymo tikslumo priklausomybę nuo HTTP pranešimų vėlavimo laiko ar tinklo tipo.

Šiame darbe buvo apsiribota galimybėmis VR programėle prisijungti prie ABB roboto valdiklio, paleisti/stabdyti valdiklio „RAPID“ programą bei valdyti robotą linijiniu būdu ar atskiras jo grandis. Tačiau VR programėlę galima tobulinti, kad būtų galima kurti sudėtingesnes robotizuotas užduotis, kaip judėjimo ciklų kūrimas, įvairių galinių įtaisų valdymas (griebtuvai, suvirinimo, dažymo aparatai, apdirbimo įrankiai), įėjimo/išėjimo signalų ar jutiklių nuskaitymas ir valdymas...

Rezultatai ir išvados

1. Projekte išanalizuotos skirtingos robotinės sistemos su virtualios realybės valdymo sąsaja ir esami sprendimai panašioms sistemoms kurti.
2. Išanalizuoti pagrindiniai naudojamos įrangos parametrai, išmokta naudotis ABB robotų aplikacijų programavimo sąsajomis, kurti ABB roboto išorines valdymo sistemas.
3. Atliktas ABB roboto konfigūravimas bei sukurta „RAPID“ valdymo programa išorinei virtualios realybės akinių programėlei veikti.
4. Sukurta virtualios realybės akinių programėlė prisijungti prie ABB roboto valdiklio ir jam valdyti. VR programėlėje komunikacijai su robotu valdikliu naudotas HTTP protokolas. Naudojant šią programėlę galima prisijungti prie roboto valdiklio, paleisti arba stabdyti valdiklio „RAPID“ programą, nuskaityti ir keisti „RAPID“ programos kintamuosius, taip realizuojant roboto grandžių arba linijinį valdymą. Programėlėje taip pat realizuotas komunikacijos pranešimų, roboto grandžių pasisukimo kampo ir galinio darbo įtaiso koordinačių bei pasisukimo kampo atvaizdavimas.
5. Programėlės veikimas testuotas ir tirtas naudojant ABB IRB 4600 robotą. Tyrime ištirtos programėlės galimybės ir valdymo tikslumas roboto pozicionavimo užduotims atlikti. Programėlėje realizuotas roboto galinio darbo įtaiso koordinačių atvaizdavimas padėjo tiksliai stumdyti galinį įtaisą koordinačių ašyse norimą atstumą. Kai robotą reikia tiksliai pozicionuoti atsitiktiniuose taškuose, roboto valdymas naudojant VR akinių programėlę turėjo vidutiniškai 4.5mm nuokrypį lyginant su valdymu naudojant standartinį valdymo pultą. Robotu galinio darbo įtaiso kalibravimui naudojant 4 manipulatoriaus orientacijų metodą, kalibravimo paklaida valdant robotą VR programėle buvo didesnė vidutiniškai 2.868mm nei valdant robotą standartiniu valdymo pultu. 6 manipulatoriaus orientacijų metodui, VR programėlės paklaida buvo didesnė už standartinio valdymo pulto vidutiniškai 2.203mm. Galima daryti išvadą, kad didelio tikslumo reikalaujančiose užduotyse šis ABB roboto valdymo būdas nėra tinkamas sprendimas.

Literatūros sąrašas

1. A. Mazumder, M.F. Sahed, Z. Tasneem, P. Das, F.R. Badal, M.F. Ali, M.H. Ahamed, S.H. Abhi, S.K. Sarker, S.K. Das, M.M. Hasan, M.M. Islam, M.R. Islam, Department of Mechatronics Engineering, Rajshahi University of Engineering & Technology, Rajshahi 6204, Bangladesh, „Towards next generation digital twin in robotics: Trends, scopes, challenges, and future“. 2023.
Žiūrėta – [2023-03-15]. Prieiga per: <https://doi.org/10.1016/j.heliyon.2023.e13359>
2. John Oyekan, Michael Farnsworth, Windo Hutabarat, David Miller, Ashutosh Tiwari, Applying a 6 dof robotic arm and digital twin to automate fan-blade reconditioning for aerospace maintenance, repair, and overhaul, *Sensors* 20 (16) (2020).
Žiūrėta – [2023-03-15]. Prieiga per: <https://doi.org/10.3390/s20164637>
3. Fernando Figueroa, Lauren Underwood, Jonathan Morris, Mark Walker, Rane Brown, Risk-reduction autonomy implementation to enable NASA Artemis missions, in: 2022 IEEE Aerospace Conference (AERO), pp. 1–13, 2022.
Žiūrėta – [2023-03-15]. Prieiga per: <https://doi.org/10.1109/AERO53065.2022.9843815>
4. Ajit Sharma, Philip Zanotti, Laxmi P. Musunur, Drive through robotics: robotic automation for last mile distribution of food and essentials during pandemics, *IEEE Access* 8 (2020) 127190–127219.
Žiūrėta – [2023-03-17]. Prieiga per: <https://doi.org/10.1109/ACCESS.2020.3007064>
5. Qibing Lv, Rong Zhang, Xuemin Sun, Yuqian Lu, Jinsong Bao, A digital twin-driven human-robot collaborative assembly approach in the wake of covid-19, *J. Manuf. Syst.* 60 (2021) 837–851.
Žiūrėta – [2023-03-17]. Prieiga per: <https://doi.org/10.1016/j.jmsy.2021.02.011>
6. Elvis IA Borges, Jonas SI Rieder, Doris Aschenbrenner, Rob BN Scharff, Framework for armature-based 3d shape reconstruction of sensorized soft robots in extended reality, *Front. Robot. AI* 9 (2022).
Žiūrėta – [2023-03-17]. Prieiga per: <https://doi.org/10.3389/frobt.2022.810328>
7. Niki Kousi, Christos Gkournelos, Sotiris Aivaliotis, Konstantinos Lotsaris, Angelos Christos Bavelos, Panagiotis Baris, George Michalos, Sotiris Makris, Digital twin for designing and reconfiguring human–robot collaborative assembly lines, *Appl. Sci.* 11 (10) (2021) 4620.
Žiūrėta – [2023-03-17]. Prieiga per: <https://doi.org/10.3390/app11104620>
8. Aljaz Kramberger, Anja Kunic, Iñigo Iturrate, Christoffer Sloth, Roberto Naboni, Christian Schlette, Robotic assembly of timber structures in a human-robot collaboration setup, *Front. Robot. AI* 8 (2021).
Žiūrėta – [2023-03-22]. Prieiga per: <https://doi.org/10.3389/frobt.2021.768038>
9. Itxaro Errandonea, Sergio Beltrán, Saioa Arrizabalaga, Digital twin for maintenance: a literature review, *Comput. Ind.* 123 (2020) 103316.
Žiūrėta – [2023-03-22]. Prieiga per: <https://doi.org/10.1016/j.compind.2020.103316>
10. Ján Vachálek, Lukáš Bartalský, Oliver Rovný, Dana Šišmišová, Martin Morhác, Milan Lokšík, The digital twin of an industrial production line within the industry 4.0 concept, in: 2017 21st International Conference on Process Control (PC), IEEE, 2017, pp. 258–262.
Žiūrėta – [2023-03-25]. Prieiga per: <https://doi.org/10.1109/PC.2017.7976223>
11. P. Aivaliotis, K. Georgoulas, Z. Arkouli, S. Makris, Methodology for enabling digital twin using advanced physics-based modelling in predictive maintenance, *Proc. CIRP* 81 (2019) 417–422.

- Žiūrėta – [2023-03-25]. Prieiga per: <https://doi.org/10.1016/j.procir.2019.03.072>
12. Luis Pérez, Eduardo Diez, Rubén Usamentiaga, Daniel F. Garcíab, „Industrial robot control and operator training using virtual reality interfaces“, *Computers in Industry* 109 (2019) 114–120.
- Žiūrėta – [2023-03-25]. Prieiga per: <https://doi.org/10.1016/j.compind.2019.05.001>
13. Polina Ponomareva, Daria Trinitatova, Aleksey Fedoseev, Ivan Kalinov, and Dzmitry Tsetserukou, „GraspLook: a VR-based Telemanipulation System with R-CNN-driven Augmentation of Virtual Environment“, *arXiv:2110.12518v1 [cs.RO]* 24 Oct 2021.
- Žiūrėta – [2023-04-09]. Prieiga per: <https://arxiv.org/pdf/2110.12518.pdf>
14. Navid Kayhani, Hosein Taghaddos, Mojtaba Noghabae, and Ulrich (Rick) Hermann, „Utilization of Virtual Reality Visualizations on Heavy Mobile Crane Planning for Modular Construction“, *35th International Symposium on Automation and Robotics in Construction (ISARC 2018)*.
- Žiūrėta – [2023-04-09]. Prieiga per: <https://doi.org/10.48550/arXiv.1901.06248>
15. Martinez-Hernandez, U., Boorman, L. orcid.org/0000-0001-5189-0232 and Prescott, T. orcid.org/0000-0003-4927-5390 (2017) „Multisensory Wearable Interface for Immersion and Telepresence in Robotics“. *IEEE Sensors Journal*, 17 (8). pp. 2534-2541. ISSN 1530-437X.
- Žiūrėta – [2023-04-09]. Prieiga per: <https://doi.org/10.1109/JSEN.2017.2669038>
16. Andrea de Giorgio, Mario Romero, Mauro Onori, Lihui Wang, „Human-machine collaboration in virtual reality for adaptive production engineering“, *Procedia Manufacturing* 11 (2017) 1279 – 1287.
- Žiūrėta – [2023-04-19]. Prieiga per: <https://doi.org/10.1016/j.promfg.2017.07.255>
17. Florian Biesinger, Davis Meikea, Benedikt Kraß, Michael Weyrich, „A digital twin for production planning based on cyber-physical systems: A Case Study for a Cyber-Physical System-Based Creation of a Digital Twin“, *Procedia CIRP* 79 (2019) 355–360.
- Žiūrėta – [2023-04-19]. Prieiga per: <https://doi.org/10.1016/j.procir.2019.02.087>
18. Carolina Saavedra Sueldo, Ivo Perez Colo, Mariano De Paula, Sebastián A. Villar, Gerardo G. Acosta, „ROS-based architecture for fast digital twin development of smart manufacturing robotized systems“, *Annals of Operations Research* (2023) 322:75–99.
- Žiūrėta – [2023-04-19]. Prieiga per: <https://doi.org/10.1007/s10479-022-04759-4>
19. ABB Robotics, „Application manual. PC SDK“. Žiūrėta – [2023-11-07]. Prieiga per: <https://library.e.abb.com/public/124d6b59313ed85fc125793400410c5b/3HAC036957-en.pdf>.
20. ABB Robotics, „Application Manual - Robot Web Services“. Žiūrėta – [2023-11-09]. Prieiga per: <https://developercenter.robotstudio.com/api/rwsApi/>.
21. 2023 Meta, „Interaction SDK Overview“. Žiūrėta – [2023-11-09]. Prieiga per: <https://developer.oculus.com/documentation/unity/unity-isdk-interaction-sdk-overview/>.
22. 2023 ABB, „RobotStudio® Suite“. Žiūrėta – [2023-11-13]. Prieiga per: <https://new.abb.com/products/robotics/robotstudio>.
23. 2020 Circuit Stream, „How to Setup Oculus Quest 2 for Development in Unity“. Žiūrėta – [2023-11-25]. Prieiga per: <https://circuitstream.com/blog/oculus-quest-2-setup>.
24. 2023 Meta, „Get Started with Meta Quest Development in Unity“. Žiūrėta – [2023-11-25]. Prieiga per: <https://developer.oculus.com/documentation/unity/unity-gs-overview/>.