Mendelova univerzita v Brně
Provozně ekonomická fakulta

Řízení polohy robotického ramene s využitím detekce gest

Diplomová práce

Vedoucí práce:
Ing. Vít Ondroušek, Ph.D.

Bc. Miloslav Schmied

Brno 2017

Tento list bude nahrazen zadáním práce.

Čestné prohlášení

Prohlašuji, že jsem tuto práci: **Řízení polohy robotického ramene s využitím detekce gest**
vypracoval samostatně a veškeré použité prameny a informace jsou uvedeny v seznamu použité literatury. Souhlasím, aby moje práce byla zveřejněna v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách ve znění pozdějších předpisů, a v souladu s platnou *Směrnicí o zveřejňování vysokoškolských závěrečných prací.*
Jsem si vědom, že se na moji práci vztahuje zákon č. 121/2000 Sb., autorský zákon, a že Mendelova univerzita v Brně má právo na uzavření licenční smlouvy a užití této práce jako školního díla podle § 60 odst. 1 Autorského zákona.
Dále se zavazuji, že před sepsáním licenční smlouvy o využití díla jinou osobou (subjektem) si vyžádám písemné stanovisko univerzity o tom, že předmětná licenční smlouva není v rozporu s oprávněnými zájmy univerzity, a zavazuji se uhradit případný příspěvek na úhradu nákladů spojených se vznikem díla, a to až do jejich skutečné výše.

v Brně dne 16. května 2017 .....

Mé poděkování patří Ing. Vítu Ondrouškovi, Ph.D. za cenné rady a metodické vedení této práce. Dále děkuji Ing. Davidu Procházkovi, Ph.D. za poskytnutí odborné konzultace. Poděkování také patří mé rodině za podporu během celého studia.

Abstract

Schmied, M. *Controlling robotic arm with gesture detection.* Diploma thesis. Brno, 2017.

The thesis deals with the design and implementation of user interface for robotic manipulator control by means of gesture detection of the operator. The used manipulator is the Katana 6M180 with five degrees of freedom. Gestures are detected using an innovated sensor system which includes Microsoft Kinect version 2 and Leap Motion sensors. The first part provides an overview of existing solutions and used technologies. The second part describes the design of the user interface, its implementation and testing.

Key words

Katana HD300s, Katana 6M180, Kinect v2, Leap Motion, gesture detection, manipulator, user interface

Abstrakt

Schmied, M. *Řízení polohy robotického ramene s využitím detekce gest.* Diplomová práce. Brno, 2017.

Tato práce se zabývá návrhem a implementací uživatelského rozhraní pro práci s robotickým manipulátorem pomocí detekce gest operátora. Použitým manipulátorem je Katana 6M180 s pěti stupni volnosti. Gesta jsou detekována pomocí inovované senzorické soustavy, která obsahuje senzory Microsoft Kinect verze 2 a Leap Motion. V první části je uveden přehled existujících řešení a použitých technologií. Ve druhé části je popsán návrh uživatelského rozhraní, jeho implementace a testování.

Klíčová slova

Katana HD300s, Katana 6M180, Kinect v2, Leap Motion, detekce gest, manipulátor, uživatelské rozhraní

Obsah

1 Úvod a cíl práce **17**
1.1 Úvod . . . 17
1.2 Cíl práce . . . 17

2 Současný stav **18**
2.1 Bezdotykové řízení polohy koncového efektoru manipulátoru . . . 18
2.2 Řízení manipulátoru pomocí hybridních snímačů . . . 18
2.3 Rozhraní mozek-stroj pro řízení manipulátoru . . . 19
2.4 Nositelné rozhraní pro řízení robotického manipulátoru . . . 19
2.5 Učení průmyslových robotických manipulátorů . . . 20
2.6 Rozhraní pro práci s duálními manipulátory . . . 21
2.7 Rozhraní využívající senzor Leap Motion . . . 22
2.8 Řízení manipulátoru pomocí hlasu a dotykových gest . . . 23
2.9 Rozhraní založené na datové rukavici . . . 24
2.10 Zhodnocení existujících řešení . . . 25

3 Použité technologie **27**
3.1 Kinect V2 . . . 27
3.2 Senzor Leap Motion . . . 28
3.3 Katana 6M180 . . . 29

4 Metodika **32**

5 Realizace **33**
5.1 Analýza požadavků . . . 33
5.2 Návrh ovládání gesty . . . 35
5.3 Návrh řídicí aplikace . . . 36
5.4 Návrh grafického uživatelského rozhraní . . . 38
5.5 Implementace . . . 39
5.6 Popis aplikace . . . 42
5.7 Kontrola síly stisku uchopovače . . . 50

6 Testování **52**
6.1 První fáze . . . 52
6.2 Druhá fáze . . . 53
6.3 Vyhodnocení provedených testů . . . 54

7 Diskuze **55**

8 Shrnutí **56**

9 Závěr **57**

10 Literatura **59**

Přílohy **62**

A CD s aplikací **63**

B Diagram tříd **64**

C Screenshoty GUI **66**

D Fotografie vytvořeného rozhraní **72**

Seznam obrázků

Obrázek 1: Uživatelské rozhraní aplikace (Kolaja, 2015) 18

Obrázek 2: Snímání inerciální měřící jednotky (Du, Zhang, 2015) 19

Obrázek 3: Nositelné rozhraní pro řízení manipulátoru (Fernandes, Frank, Kapila, 2014) 20

Obrázek 4: Navržená rozhraní (Capi, Hossain, Kaneko, Bode, 2015) 21

Obrázek 5: Barevné snímky operátora, hloubkové mapy a virtuální prostředí (Du, Zhang, 2013) 22

Obrázek 6: Struktura navrženého rozhraní (Du, Zhang, Liu, 2016) 23

Obrázek 7: Dotyková gesta (Oka, Matsushima, 2016) 24

Obrázek 8: Datová rukavice (Paredes-Madrid, Gonzalez de Santos, 2013) 24

Obrázek 9: Microsoft Kinect verze 2 (Wikipedia, 2017) 27

Obrázek 10: Senzor Leap Motion (Wikipedia, 2017) 28

Obrázek 11: Snímání rukou (Nepal's Online Shopping Portal, 2015) 29

Obrázek 12: Manipulátor Katana 6M180 (Kolaja, 2015, s. 14) 30

Obrázek 13: Uchopovač typu S03.02 (Neuronics AG, 2006, s. 20) 30

Obrázek 14: Umístění senzorů na uchopovači (Janda, 2015, s. 11) 31

Obrázek 15: Use Case diagram 34

Obrázek 16: Gesto zatlačení (Simple-ism, 2014) 36

Obrázek 17: Stavový diagram aplikace 36

Obrázek 18: Schéma řídicího řetězce 37

Obrázek 19: Funkcionalita navržených tříd 38

Obrázek 20: Drátový model GUI 39

Obrázek 21: Visual Gesture Builder 41

Obrázek 22: Kalibrace manipulátoru (screenshot GUI) 43

Obrázek 23: Zastavená manipulace (výřezy) 44

Obrázek 24: Změna ruky pro manipulaci (výřezy) 45

Obrázek 25: Spuštění manipulace (screenshot GUI) 46

Obrázek 26: Probíhající manipulace (výřezy) 47

Obrázek 27: Otevření / zavření uchopovače 48

Obrázek 28: Pozastavená manipulace (výřezy) 49

Obrázek 29: Indikace umístění ruky do přesné pozice (screenshot GUI) 50

Obrázek 30: Sevření předmětů z různých materiálů 51

Obrázek B.1: Diagram tříd 64

Obrázek B.2: Diagram statických tříd 65

Obrázek C.1: Kalibrace manipulátoru 67

Obrázek C.2: Zastavená manipulace 68

Obrázek C.3: Změna ruky pro manipulaci 69

Obrázek C.4: Probíhající manipulace manipulace 70

Obrázek C.5: Pozastavená manipulace 71

Obrázek D.1: Fotografie vytvořeného uživatelského rozhraní 72

Seznam tabulek

Tabulka 1: Technické parametry senzoru Kinect v2 (Microsoft, 2017d) 27

Tabulka 2: Technické parametry manipulátoru Katana 6M180 (Neuronics AG, 2006) 31

Tabulka 3: Výsledky první fáze testování detekce gest 52

Tabulka 4: Úspěšnost první fáze testů 52

Tabulka 5: Výsledky druhé fáze testování detekce gest 53

Tabulka 6: Úspěšnost druhé fáze testů 54

1 Úvod a cíl práce

1.1 Úvod

Nasazování průmyslových robotů a manipulátorů v praxi je dnes čím dál tím častější jev. Robotické manipulátory dokáží pracovat v rizikových prostředích, která jsou pro lidi nebezpečná. Dokáží repetitivně plnit úkoly s vysokými nároky na přesnost a rychlost, které by pro člověka byly nesplnitelné. Vznikají tak robotizovaná pracoviště, ve kterých hrozí člověku nebezpečí ze strany robota. Přístup je zpravidla k takovémuto pracovišti omezený, případně je mu zcela zamezeno mechanickými zábranami (bariéry, klece, zóny, …).

Dalším trendem je aplikace „inteligentních“ robotických manipulátorů, které kooperují s člověkem. Aby bylo zabráněno úrazu, robot musí analyzovat pohyb člověka. Kooperativní manipulátor v blízkosti člověka zastaví, zpomalí pohyb nebo omezí výkon a sílu. Moderní manipulátory umožňují i ruční vedení.

Konvenční způsob řízení manipulátoru je pomocí přenosného programovacího panelu. Operátor manipulátoru plánuje trajektorii definováním pozic, kterými má procházet. Současně může definovat akce, které má vykonat. Výhodou je vysoká přesnost manipulace. Nevýhodou je, že pohyb mezi pozicemi na trajektorii nemusí být operátorovi znám. Dalším způsobem plánování trajektorie je ruční vedení manipulátoru, který je poté pouze opakován.

Výše zmíněné způsoby řízení vyžadují od operátora expertní znalost pro práci s manipulátorem. Snahou této práce je usnadnit operátorovi práci pomocí přirozenějšího uživatelského rozhraní, které bude jednoduché použít a které nebude klást na operátora tak velké požadavky.

1.2 Cíl práce

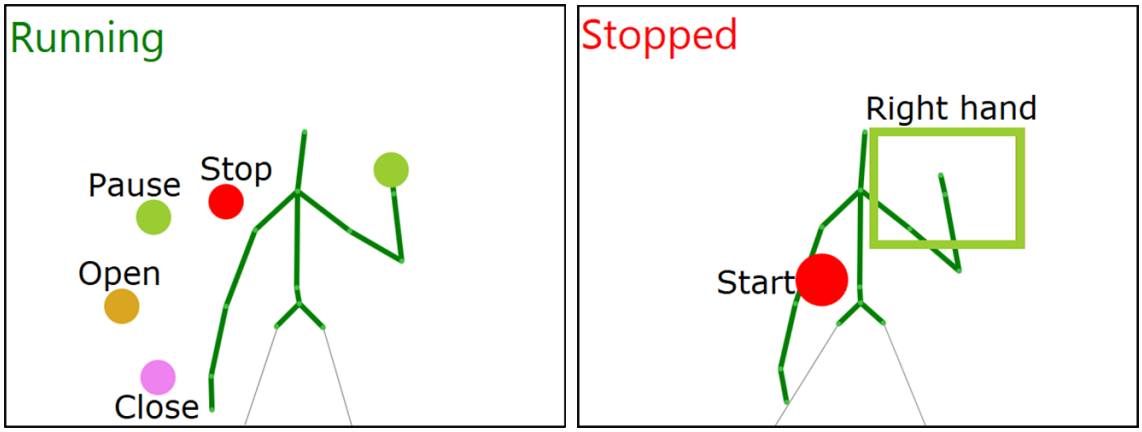
Cílem této práce je návrh, implementace a testoání uživatelského rozhraní, které umožní řídit polohu koncového efektoru robotického stacionárního manipulátoru Katana 6M180 s pěti stupni volnosti pomocí detekce gest operátora. K detekci gest bude použita inovovaná senzorická soustava, která se skládá ze senzoru Kinect verze 2 pro snímání postavy operátora a senzoru Leap Motion pro snímání jeho rukou a prstů.

2 Současný stav

V této kapitole je obsaženo shrnutí vědeckých článků nebo publikací zabývajících se problematikou nekonvenčního uživatelského rozhraní pro řízení robotického ramene. Existující řešení jsou poté zhodnocena.

2.1 Bezdotykové řízení polohy koncového efektoru manipulátoru

Jedná se o současné řešení, které je použito pro ovládání manipulátoru Katana 6M180 na Mendelově univerzitě v Brně. Uživatel je snímán senzorem Kinect první verze. Grafické uživatelské rozhraní pro práci s manipulátorem je tvořeno oknem, ve kterém uživatel vidí svou „kostru“, virtuální menu a textové pole, do kterého je vypisován stav manipulace (obr. 1a). Jako kurzor pro virtuální menu slouží uživatelova levá ruka. Pro spuštění manipulace musí uživatel umístit svou pravou ruku do označeného čtverce a současně levou ruku umístit na virtuální tlačítko *Start* (obr. 1b). Pro manipulaci je použita jeho pravá ruka, jejíž pohyb je pří manipulaci koncovým efektorem kopírován (Kolaja, 2015).

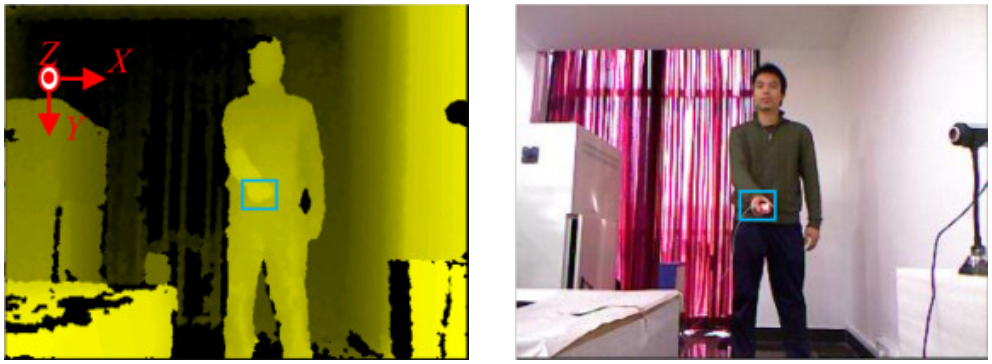


(a) Spuštěná manipulace (b) Zastavená manipulace

Obrázek 1: Uživatelské rozhraní aplikace (Kolaja, 2015)

2.2 Řízení manipulátoru pomocí hybridních snímačů

Článek se zabývá řízením manipulátoru pomocí 3D kamery a inerciální měřící jednotky. Pro ovládání koncového efektoru musí uživatel držet inerciální měřící jednotku v pravé ruce a zároveň být snímán kamerou. Inerciální měřící jednotka snímá orientaci ruky a kamera její polohu (obr. 2). Kálmánův filtr je použit pro odhad polohy ruky. Pro odhad orientace ruky a úhlových rychlostí je použit filtr částic a filtr s konečnou impulzní odezvou. Zavírání a otevírání uchopovače je řešeno zastavením pohybu ruky po dobu 3 sekund. Do budoucna však autoři plánují k ovládání uchopovače využít rozpoznávání hlasu (Du, Zhang, 2015).



(a) Hloubková mapa (b) Barevný obraz uživatele

Obrázek 2: Snímání inerciální měřící jednotky (Du, Zhang, 2015)

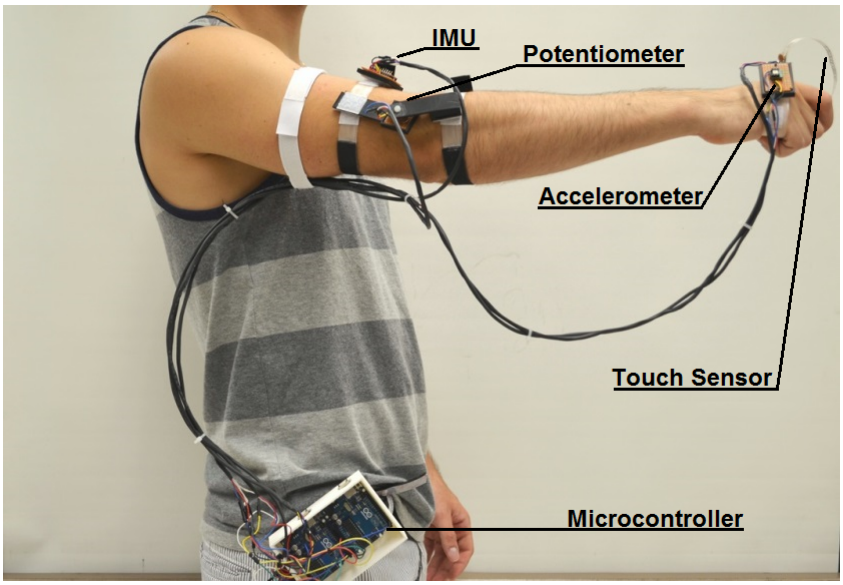
2.3 Rozhraní mozek-stroj pro řízení manipulátoru

Zcela odlišný přístup je popsán v článku, který se zabývá experimentálním řízením manipulátoru s 6 stupni volnosti pomocí neinvazivního snímání EEG signálů. Pro klasifikaci EEG signálů je použita neuronová síť s topologií small-world. Systém identifikuje 5 stavů mozkové aktivity (představa levé ruky, pravé ruky, nohy, pohyb jazykem a ostatní). Pohyb manipulátoru je řízen příkazy, které se skládají ze sekvence dvou klasifikovaných EEG signálů. Například pro kladný pohyb po ose X je potřeba si představit levou ruku a poté nohu (Li, Hong, Zhang, 2013).

2.4 Nositelné rozhraní pro řízení robotického manipulátoru

Tento článek se zabývá popisem a vývojem nositelného rozhraní, které využívá přirozený pohyb uživatelovy paže k řízení manipulátoru se 4 stupni volnosti. Nositelné rozhraní (obr, 3) se skládá z několika senzorů, které jsou umístěny na plastových dílech vyrobených 3D tiskem. Na ruku jsou potom připevněny pomocí suchého zipu. Mezi použitými senzory je inerciální měřící jednotka, potenciometr, akcelerometr a dotykový senzor, který sloužil k ovládání uchopovače. Jako mikrokontroler je použit Arduino Uno.

Nositelné rozhraní bylo experimentálně porovnáno s dalšími rozhraními na úloze zdvižení a položení předmětu. V porovnávání byla aplikace pro tablet, která umožňovala ovládat manipulátor naklápěním a pomocí tlačítek na displeji a herní ovladač. Výsledný čas potřebný k zvládnutí testovací úlohy byl výrazně kratší s použitím nositelného rozhraní. Jeho nevýhodou však bylo snímání třesoucích se rukou (Fernandes, Frank, Kapila, 2014).



Obrázek 3: Nositelné rozhraní pro řízení manipulátoru (Fernandes, Frank, Kapila, 2014)

2.5 Učení průmyslových robotických manipulátorů

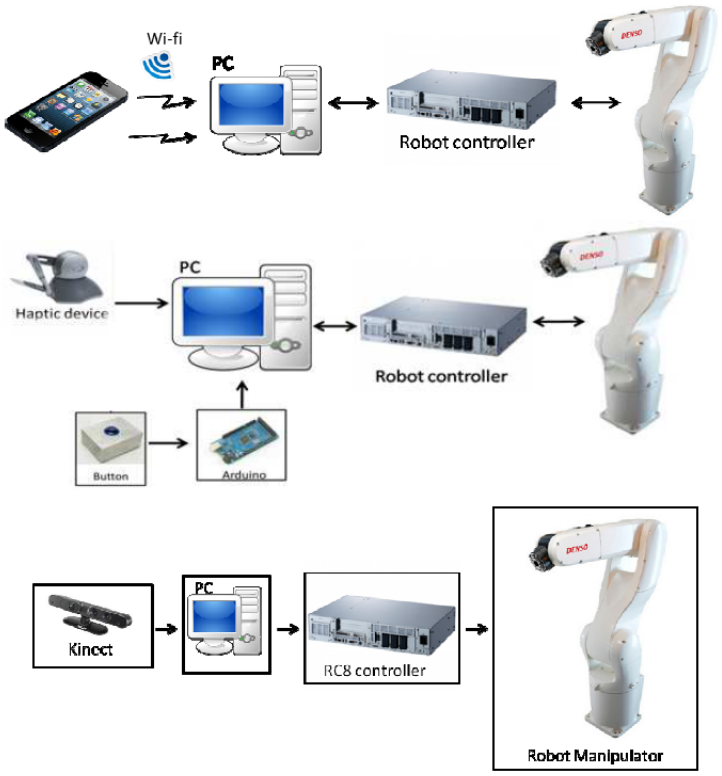
Článek se zabývá měřením výkonnosti 3 navržených učících rozhraní manipulátoru, která nevyžadují expertní znalosti. V článku je použit manipulátor Puma s 6 stupni volnosti. Navržená rozhraní jsou znázorněna na obrázku 4.

První rozhraní je založené na iPhonu, který bezdrátově komunikuje s řídicím počítačem přes LAN. Jeho nakláněním se určuje směr a rychlost pohybu manipulátoru. K ovládání uchopovače musí uživatel stisknout klávesu „K“ na klávesnici řídicího počítače. Klávesa „S“ slouží ke zpomalení pohybu a klávesa „Q“ k jeho ukončení.

Druhé rozhraní je haptické, využívá zařízení Phantom Omni od firmy Sensable Technologies. Jde o zařízení s 6 stupni volnosti, které poskytuje haptickou odezvu ve 3 rozměrech. Úhly jeho vazeb jsou přepočítány na úhly vazeb manipulátoru. Uchopovač je ovládán stiskem tlačítka.

Třetí rozhraní je založené na sensoru Kinect. Manipulátor kopíruje pohyb ruky. Zvednutím levé ruky uživatel zpomalí rychlost manipulátoru pro zvýšení přesnosti a snížení chybovosti. Uchopovač je ovládán zavíráním a otevíráním dlaně.

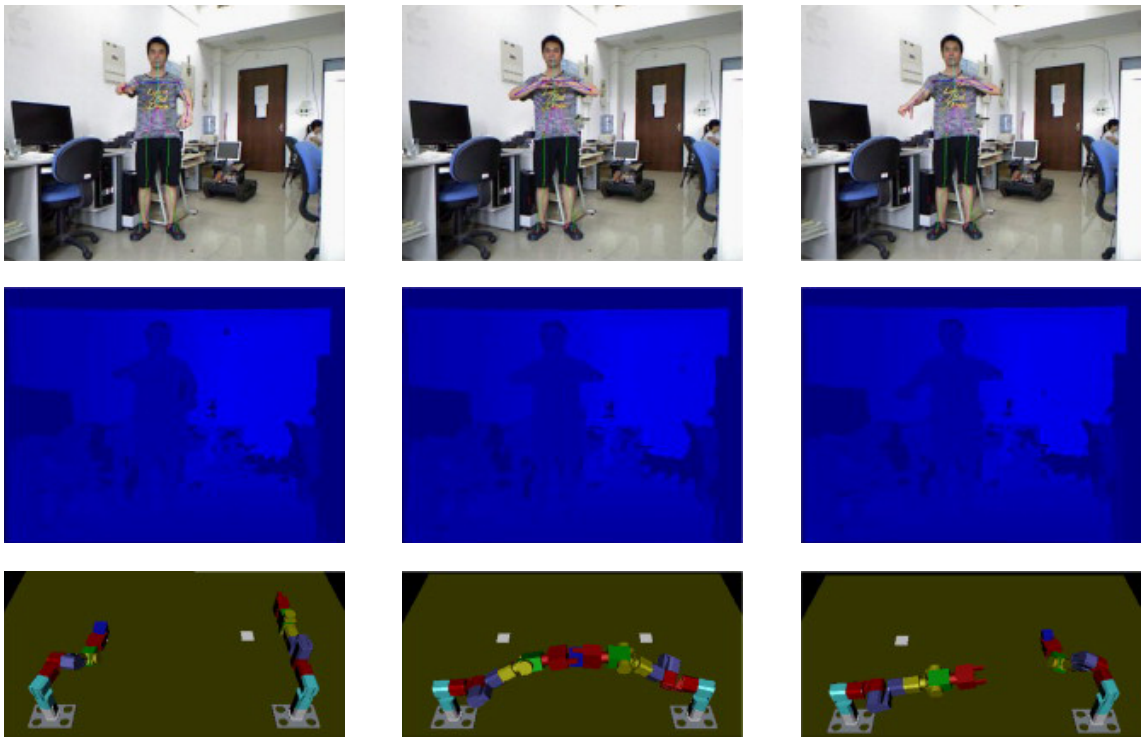
Tato rozhraní byla porovnána na úloze zvednutí a položení předmětu z hlediska času potřebného k provedení a přesnosti provedení. Haptické rozhraní se ukázalo být nejlepším (Capi, Hossain, Kaneko, Bode, 2015).



Obrázek 4: Navržená rozhraní (Capi, Hossain, Kaneko, Bode, 2015)

2.6 Rozhraní pro práci s duálními manipulátory

Článek popisuje řízení dvou manipulátorů s 6 stupni volnosti ve virtuálním prostředí. K jejich řízení jsou použita data ze senzoru Kinect první verze. Každý z manipulátorů odpovídá jedné ruce uživatele. Grafické uživatelské rozhraní aplikace zobrazuje 3D scénu s modely manipulátorů a jedním předmětem, kterým lze manipulovat. Pro inicializaci manipulace musí uživatel napnout obě paže. Uživatelské rozhraní je znázorněno na obrázku 5 (Du, Zhang, 2013).

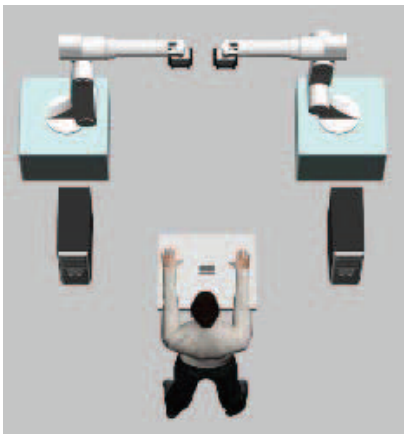


Obrázek 5: Barevné snímky operátora, hloubkové mapy a virtuální prostředí (Du, Zhang, 2013)

2.7 Rozhraní využívající senzor Leap Motion

V článku je navrženo a otestováno rozhraní pro práci s dvěma manipulátory s šesti stupni volnosti. Řízení manipulátorů probíhá na základě dat ze senzoru Leap Motion, který snímá obě ruce operátora. Manipulátory pak zrcadlí pohyb rukou operátora v reálném čase. Pro odhad pozice rukou je použit intervalový Kálmánův filtr a vylepšený filtr částic. Pro eliminaci třesu rukou je použita adaptivní multiprostorová transformace.

Struktura navrženého rozhraní je znázorněna na obrázku 6. Aby se operátor během manipulace nedostal rukou mimo dosah senzoru Leap Motion, může sevřít dlaň a zrcadlení pozice ruky bude přerušeno. Poté může ruku přemístit zpět do zorného pole senzoru a dlaň rozevřít, čímž opět aktivuje zrcadlení pozice ruky.



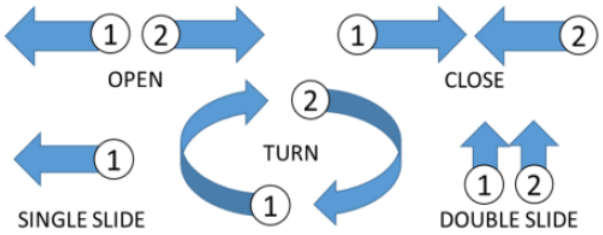
Obrázek 6: Struktura navrženého rozhraní (Du, Zhang, Liu, 2016)

2.8 Řízení manipulátoru pomocí hlasu a dotykových gest

V konferenčním příspěvku je navrženo rozhraní pro práci s manipulátorem se sedmi stupni volnosti. Ovládání je realizováno rozpoznáváním hlasových povelů a pomocí dotykové plochy. Hlasové povely slouží pro zvolení módu ovládání: translační nebo rotační pohyb. V módu translace operátor ovládá pozici manipulátoru v trojrozměrném prostoru. Mód rotace slouží pro ovládání natočení uchopovače. Lze také vytvářet nové hlasové povely za běhu, při kterých se uloží aktuální pozice manipulátoru. Nově vytvořený povel pak slouží pro přesunutí manipulátoru na uloženou pozici.

Pozici manipulátoru ovládá operátor pomocí gest na dotykové obrazovce. Definovaná gesta jsou znázorněna na obrázku 7. Posouváním prstu po dotykové ploše ovládá pohyb v horizontální rovině. Posouváním dvěma prsty najednou hýbe manipulátorem v rovině vertikální. Dále lze gesty otevírat a zavírat uchopovač nebo ho otáčet okolo své osy.

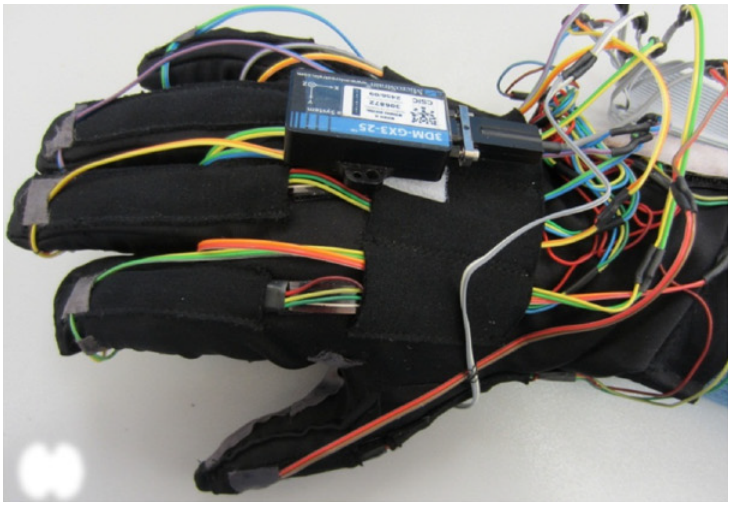
Počet bodů, kterými se operátor dotýká obrazovky, umožňuje provádět další akce. Pokud se při ovládání pozice dotkne na jednom dalším bodu, dojde ke zpomalení rychlosti manipulátoru. Dotknutím na dalších dvou bodech dojde k pozastavení manipulace a dotknutí na dalších třech bodech manipulaci ukončí (Oka, Matsushima, 2016).



Obrázek 7: Dotyková gesta (Oka, Matsushima, 2016)

2.9 Rozhraní založené na datové rukavici

Článek se zabývá návrhem rozhraní pro ovládání manipulátoru s šesti stupni volnosti. Rozhraní se skládá z datové rukavice, která byla doplněna o 11 senzorů síly, z inerciální měřící jednotky a z kontroléru, který umožňuje řídit manipulátor v reálném čase. Rukavice umožňuje komplexní měření sil působených rukou. Pozice ruky v prostoru je měřena inerciální měřící jednotkou. Kontrolér pro zpracování dat ze senzorů je umístěn v batohu, který musí operátor nést (Paredes-Madrid, Gonzalez de Santos, 2013).



Obrázek 8: Datová rukavice (Paredes-Madrid, Gonzalez de Santos, 2013)

2.10 Zhodnocení existujících řešení

Uživatelské rozhraní navržené v diplomové práci Bezdotykové řízení polohy koncového efektoru manipulátoru (Kolaja, 2015) neumožňuje volbu ruky pro manipulaci. Z vlastní zkušenosti vím, že uživatel při práci s manipulátorem musí střídavě sledovat manipulátor a monitor, na kterém je vykreslováno virtuální menu pro ovládání aplikace.

V článku Řízení manipulátoru pomocí hybridních snímačů (Du, Zhang, 2015) je uživatel také bohužel omezen na ovládání manipulace pouze pravou rukou. Bohužel zde není popsáno, jakým způsobem uživatel manipulaci spustí a zastaví. Ovládání uchopovače zastavením pohybu ruky po dobu 3 sekund mi připadá značně neergonomické. Použití rozpoznávání hlasu, které autoři zmiňují, by podle mě bylo velkým přínosem.

Práce Rozhraní mozek-stroj pro řízení manipulátoru (Li, Hong, Zhang, 2013) se zabývá ovládáním manipulátoru pomocí snímání mozkové aktivity, což je zcela odlišný přístup.

V konferenčním příspěvku Nositelné rozhraní pro řízení robotického manipulátoru (Fernandes, Frank, Kapila, 2014) je pohyb uživatelovy ruky snímán fyzicky připojenými senzory, což může být na úkor pohodlí. Bohužel zde není popsáno spuštění a zastavení manipulace.

Článek Učení průmyslových robotických manipulátorů pomocí (Capi, Hossain, Kaneko, Bode, 2015) popisuje návrh hned 3 rozhraní, která mají sloužit pro učení manipulátoru. Jako nejzdařilejší se mi zdá rozhraní s využitím senzoru Kinect, protože využívá přirozený pohyb uživatelovy ruky včetně otevření / sevření dlaně k ovládání uchopovače.

Rozhraní pro práci s duálními manipulátory (Du, Zhang, 2013) pro řízení manipulace také používá senzor Kinect. Manipulace probíhá pouze ve virtuálním prostředí. Spuštění manipulace provede uživatel napnutím paží, otevření / zavření gripperu a ukončení manipulace v článku není popsáno.

Článek Rozhraní využívající senzor Leap Motion (Du, Zhang, Liu, 2016) využívá zmíněný senzor pro detekci obou rukou operátora. To operátorovi umožňuje ovládat bezdotykově dva manipulátory zrcadlením pozice jeho rukou v reálném čase. Způsob spuštění nebo zastavení manipulace není v článku popsán. Zdařilý se mi jeví způsob, jakým může operátor přemisťovat ruce v zorném poli senzoru, pokud by se dostal na jeho okraj.

Příspěvek Řízení manipulátoru pomocí hlasu a dotykových gest (Oka, Matsushima, 2016) popisuje rozhraní, ve kterém je manipulátor ovládán hlasovými povely a pomocí dotykového panelu. Hlasovými povely lze přepínat mezi ovládacími módy a samotné ovládání polohy je řešeno dotykovými gesty. Poměrně velké množství dotykových gest může snižovat uživatelskou přívětivost. Na druhou stranu, operátor může manipulátor přesně ovládat.

V článku Rozhraní založené na datové rukavici (Paredes-Madrid, Gonzalez de Santos, 2013) je navrženo nositelné rozhraní, které využívá datovou rukavici pro

měření síly působené rukou a inerciální měřící jednotku pro snímání polohy ruky. Rukavice je určena pouze pro pravou ruku a operátor tak nemá možnost zvolit si ruku pro ovládání manipulace. Operátor je nucen nést batoh, ve kterém je kontrolér pro zpracování dat ze senzorů. Komplexní měření síly působené rukou je vykoupeno sníženou ergonomií řešení. Způsob spuštění nebo ukončení manipulace článek nepopisuje.

Je vidět, že existuje mnoho přístupů pro vytvoření rozhraní mezi člověkem a manipulátorem. Mezi existujícími řešeními má největší zastoupení senzor Kinect. Po něm následují nositelná rozhraní. V nalezených řešeních není kladen příliš velký důraz na uživatelskou přívětivost rozhraní pro práci s manipulátorem, a tak prostor pro nová řešení je v tomto ohledu stále značný.

3 Použité technologie

Tato kapitola se zabývá popisem technologií použitých v práci. Jedná se zejména o popis senzorů použitých pro snímání polohy lidského těla a jejich možností pro dosažení cíle práce. Dále je zde uveden popis použitého robotického manipulátoru.

3.1 Kinect V2

Kinect (obr. 9) je senzor od společnosti Microsoft, který se skládá z RGB kamery, hloubkového senzoru a pole mikrofonů. Umožňuje snímání pohybu těla ve 3D prostoru, rozpoznání obličeje a rozpoznání hlasu. Druhá verze tohoto senzoru vylepšuje přesnost snímání těla. Nově je možno snímat až 6 skeletonů současně a u každého z nich je snímáno 25 kloubů. Barevná kamera poskytuje obraz v rozlišení 1080p. Vylepšeno je také snímání hloubky a aktivní infračervený senzor umožňuje nezávislost na světelných podmínkách (Wikipedia, 2017).

Obrázek 9: Microsoft Kinect verze 2 (Wikipedia, 2017)

Tabulka 1: Technické parametry senzoru Kinect v2 (Microsoft, 2017d)

Technické parametry	**Kinect v2**
Rozlišení hloubkového senzoru	512 × 424
Frekvence hloubkového senzoru	30 Hz
Rozlišení barevné kamery	1920 × 1080
Frekvence barevné kamery	30 Hz / 15 Hz[1]
Zorné pole	70° × 60°
Dosah snímání	0,5–4,5 m

[1]Při slabém osvětlení je použita frekvence 15 Hz.

SDK pro druhou verzi Kinectu přináší řadu změn a vylepšení zejména díky změnám v hardwaru. Vylepšen byl i nástroj Kinect Studio, který slouží k nahrávání a přehrávání klipů. Nově přibyl nástroj Visual Gesture Builder, který umožňuje definovat vlastní gesta metodou strojového učení (Microsoft, 2017a).

Pro ovládání manipulátoru by se z SDK nechalo použít rozpoznávání hlasu nebo zmíněné definování gest. Pří snímání hlasu umožňuje SDK zjistit i úhel zdroje zvuku. Pro samotné rozpoznávání hlasu je možné použít Microsoft Speech Platform SDK (Microsoft, 2017e). Kinect SDK umožňuje zjistit i natočení uživatelova obličeje. To by se nechalo použít např. pro kontrolu, zda uživatel skutečně sleduje manipulátor během práce s ním z důvodu bezpečnosti (Microsoft, 2017c).

3.2 Senzor Leap Motion

Senzor Leap Motion je malé periferní USB zařízení (obr. 10) určené pro snímání pohybu rukou a prstů. Navrženo je tak, aby leželo na desce stolu otočeno směrem vzhůru, případně bylo umístěno na přední straně headsetu pro virtuální realitu. Využívá dvě monochromatické infračervené kamery a tři infračervené LED (Wikipedia, 2017).

Obrázek 10: Senzor Leap Motion (Wikipedia, 2017)

Senzor snímá prostor přibližně v úhlu 150°. Efektivní dosah senzoru sahá od výšky 82,5 do 317,5 mm (Leap Motion, 2015a).

SDK verze 2.3 umožňuje rozpoznat a sledovat ruce, prsty a nástroje podobné prstům (např. ukazovátko) (Leap Motion, 2015a). Snímání rukou je znázorněno na obrázku 11. Za účelem ovládání manipulátoru se nechá využít detekce gest. SDK obsahuje detektor 4 předem definovaných gest, konkrétně jsou to tato: *Circle*, *Swipe*,

Key Tap a *Screen Tap*. Žádný nástroj pro definování vlastních gest k dispozici bohužel není (Leap Motion, 2015b).

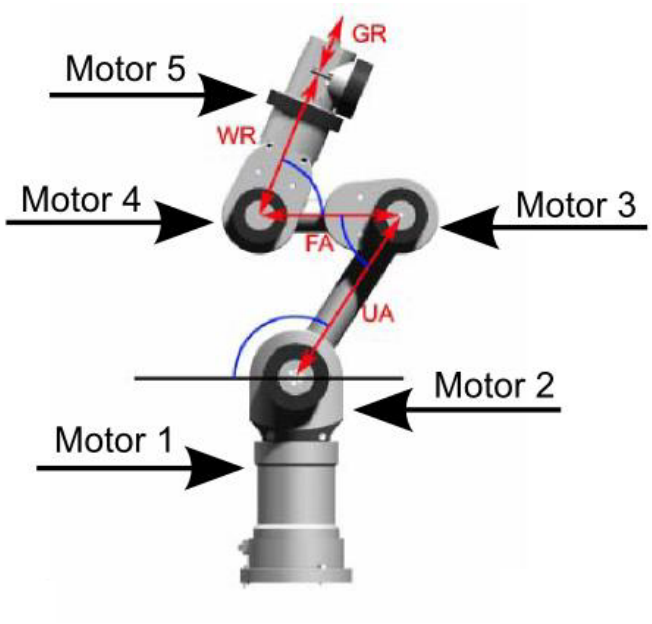
Obrázek 11: Snímání rukou (Nepal's Online Shopping Portal, 2015)

V článku Analysis of the Accuracy and Robustness of the Leap Motion Controller je měřena přesnost snímání. Celková průměrná změřená přesnost byla 0,7 mm (Weichert, Bachmann, Rudak, Fisseler, 2013).

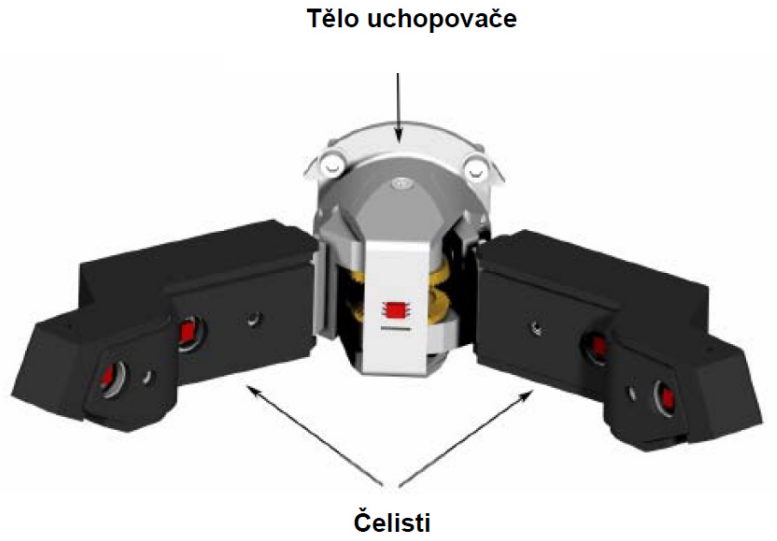
3.3 Katana 6M180

Jedná se o manipulátor série Katana HD300s švýcarského výrobce Neuronics AG. Manipulátory této série jsou navrženy a konstruovány tak, aby je bylo možné použít pro kooperaci s lidskou obsluhou bez omezení na pracovišti. Velikost těchto manipulátorů je menší, takže je lze snadno přemístit. Tyto vlastnosti umožňují jejich použití pro akademické účely. Manipulátor je znázorněn na obrázku 12, technické parametry jsou uvedeny v tabulce 2.

Manipulátor je možné ovládat pomocí knihoven Katana Native Interface (KNI) v programovacím jazyce C++ nebo Python. Další možností je ovládání pomocí výrobcem dodávaného SW nástroje Katana4D, který umožňuje vytvářet programy pro řízení a číst data ze senzorů. Poslední možností pro ovládání je prostředí Control Web, které vytváří rozhraní pro práci s manipulátorem pomocí technologie ActiveX (Vytečka, 2013).



Obrázek 12: Manipulátor Katana 6M180 (Kolaja, 2015, s. 14)

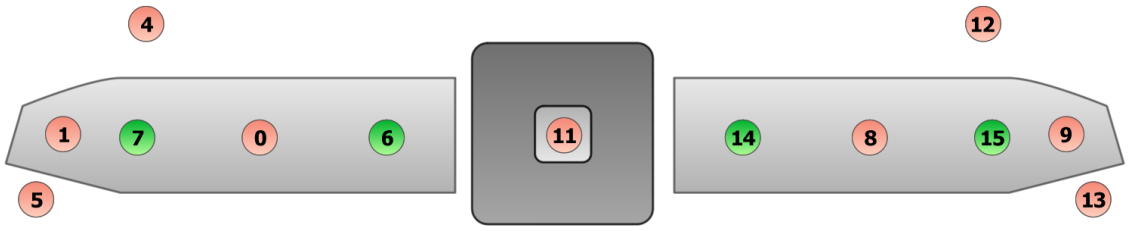


Obrázek 13: Uchopovač typu S03.02 (Neuronics AG, 2006, s. 20)

Jako efektor manipulátoru je použit univerzální uchopovač typu S03.02 z obrázku 13. Skládá se z těla uchopovače, dvou čelistí, 9 infračervených senzorů vzdálenosti a 4 senzorů síly.

Umístění senzorů na uchopovači je znázorněno na obrázku 14. Infračervené senzory vzdálenosti jsou vyznačeny červeně, senzory síly jsou vyznačeny zeleně. Dva IR senzory jsou umístěné na vnější straně čelistí a dva jsou umístěné na jejich spodní straně. Oba typy senzorů poskytují hodnoty v rozsahu 0–255 (8 bitů), kdy 0 před-

stavuje neaktivovaný senzor (Neuronics AG, 2006).



Obrázek 14: Umístění senzorů na uchopovači (Janda, 2015, s. 11)

Tabulka 2: Technické parametry manipulátoru Katana 6M180 (Neuronics AG, 2006)

Technické parametry	**Katana 6M180**
Počet stupňů volnosti (DOF)	5
Max. výška [mm]	854[2]
Úhly vyosení [°]	M1: 0
	M2: 124,25
	M3: 52,7
	M4: 63,5
	M5: 8,5
Operační rozsah [°]	M1: 345,7
	M2: 140
	M3: 241,5
	M4: 232
	M5: 332,2
	M6: 140 (čelisti)
Délka ramene [mm]	UA: 190
	FA: 139
	WR: 185
	GR: 130 (s uchopovačem)

[2]Bez základové desky, s čelistmi.

4 Metodika

V této kapitole jsou popsány konkrétní kroky potřebné k dosažení cíle práce.

1. Zjištění možností senzorů Kinect verze 2 a Leap Motion relevantních k cíli práce.
2. Navržení uživatelského rozhraní pro řízení manipulátoru: návrh GUI a návrh gest. Uživateli je potřeba umožnit provádět tyto akce: volba ruky pro ovládání manipulace, spuštění manipulace, pozastavení manipulace, opětovné spuštění pozastavené manipulace, ovládání uchopovače a ukončení manipulace.
3. Implementace prototypu uživatelského rozhraní s detekcí gest pro ověření použitelnosti detekce.
4. Implementace navrženého uživatelského rozhraní.
5. Připojení manipulátoru k uživatelskému rozhraní. Použití stávajícího řešení pro výpočet inverzní úlohy kinematiky. Ošetření případů, kdy by mohlo dojít k poškození manipulátoru např. nárazem do podložky.
6. Testování vytvořeného uživatelského rozhraní.

5 Realizace

V této kapitole je popsán postup při návrhu a implementaci řídicí aplikace pro ovládání manipulátoru. Testování detekce gest a aplikace je vyčleněno do samostatné kapitoly.

Tato práce navazuje na existující řešení (Kolaja, 2015), které je popsáno na str. 18. Od Kolaji je převzat výpočet inverzní kinematiky pro zrcadlení koncového efektoru manipulátoru podle polohy dlaně operátora a výchozí poloha manipulátoru. Řídicí řetězec byl pozměněn, senzor Microsoft Kinect byl nahrazen jeho druhou verzí a přibyl senzor Leap Motion.

5.1 Analýza požadavků

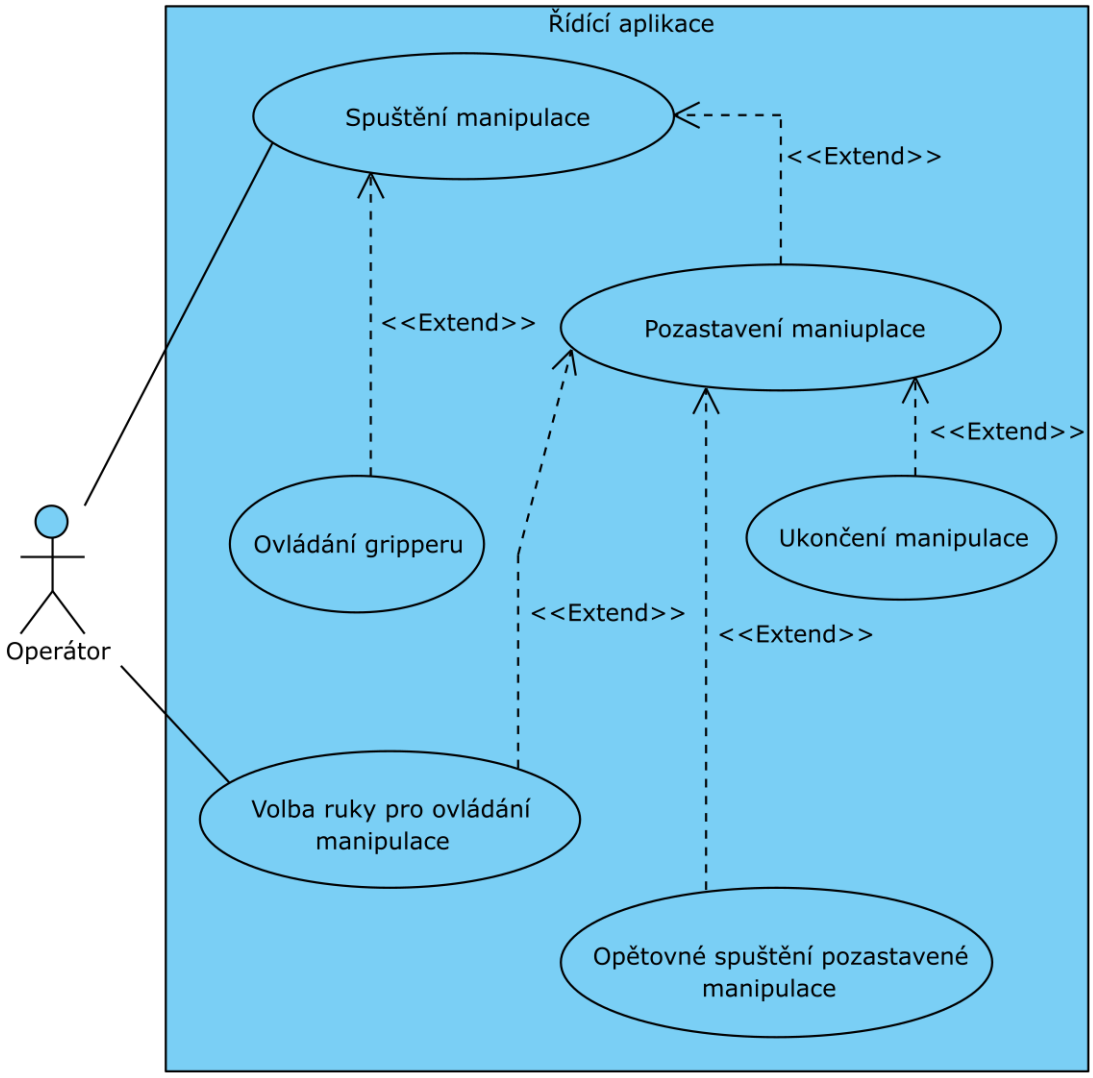
Před zahájením práce bylo potřeba specifikovat požadavky pro řídicí aplikaci. Funkční požadavky vyplývají ze zadání této práce a dále mi byly upřesněny vedoucím práce.

Funkční požadavky

Aplikace musí operátorovi poskytnout tuto funkcionalitu:

- Volba ruky pro ovládání manipulace
- Spuštění manipulace
- Pozastavení manipulace
- Opětovné spuštění pozastavené manipulace
- Ovládání uchopovače
- Ukončení manipulace

Požadovaná funkcionalita aplikace je z pohledu operátora popsána Use Case diagramem na obrázku 15.



Obrázek 15: Use Case diagram

Senzory pro snímání operátora jsou určeny v zadání této práce. Konkrétně jsou to senzory Microsoft Kinect verze 2 a Leap Motion. Stejně tak je určen i použitý manipulátor, kterým je Katana 6M180.

Další požadavky jsou kladeny na bezpečnost uživatele a okolí při práci s manipulátorem. V případě, že dojde k přerušení detekce operátora senzorem Kinect, musí být manipulace ukončena. Aplikace musí umožňovat ovládání manipulátoru pouze jednomu operátorovi, ostatní osoby nesmí do ovládání zasahovat. Aplikace musí zamezit poškození manipulátoru nárazem do podložky. Při uchopování předmětu by mohlo dojít k vyvinutí příliš velké síly, což by mohlo mít za následek nežádoucí deformaci předmětu nebo uchopovače.

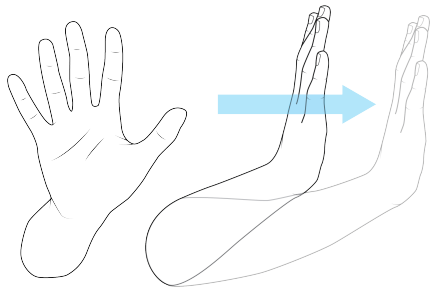
Nefunkční požadavky

Použití oficiálního SDK senzoru Kinect vyžaduje operační systém Windows. Vyžadováno je i použití vývojového prostředí Visual Studio 2012 nebo 2013. Tyto požadavky zároveň splňuje i SDK senzoru Leap Motion.

5.2 Návrh ovládání gesty

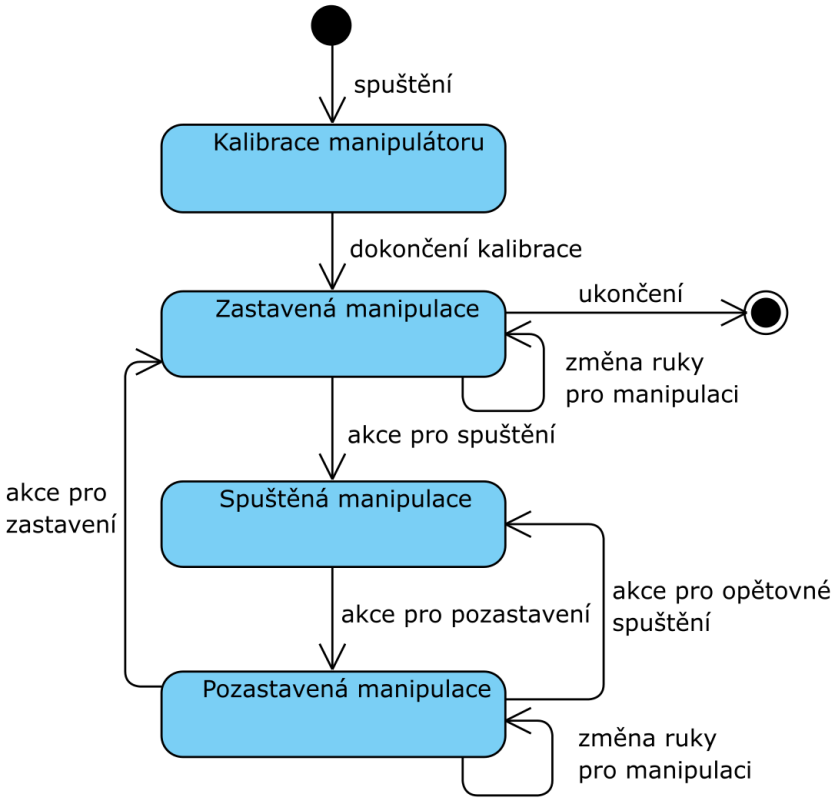
Za účelem ovládání manipulátoru pomocí gest bylo nezbytné nejprve tato gesta navrhnout. Gesta slouží ke spouštění akcí, kterými uživatel ovládá manipulaci a celou aplikaci. Pro tyto akce jsem navrhl gesta v souladu s možnostmi senzorů použitých pro jejich detekci následovně:

- *Volba ruky pro ovládání manipulace* - uživatel zvedne při zastavené nebo pozastavené manipulaci požadovanou ruku nad hlavu, čímž ji nastaví jako hlavní. Ve výchozím stavu je jako hlavní nastavena pravá ruka.
- *Spuštění manipulace* - uživatel podrží zvolenou ruku v oblasti vyznačené v GUI po dobu 2 sekund.
- *Pozastavení manipulace* - uživatel provede gesto zatlačení dlaní směrem od sebe rukou, která není používána pro manipulaci. Gesto je znázorněno na obrázku 16.
- *Opětovné spuštění pozastavené manipulace* - uživatel podrží zvolenou ruku v oblasti vyznačené v GUI po dobu 2 sekund. Vyznačená oblast je pozicována do místa, ve kterém byla hlavní ruka při pozastavení manipulace. Při opětovném spuštění musí být poloha hlavní ruky v osách x, y, a z přibližně totožná s polohou při pozastavení manipulace.
- *Ovládání uchopovače* - uchopovač je ovládán otevřením / zavřením uživatelovy ruky, která není používána pro manipulaci. Otevírání / zavírání ruky je tak kopírováno uchopovačem.
- *Ukončení manipulace* - ukončit manipulaci lze pouze, pokud je pozastavena. Pro ukončení manipulace provede gesto zatlačení rukou, která není použita pro manipulaci (stejným způsobem jako pro pozastavení manipulace).



Obrázek 16: Gesto zatlačení (Simple-ism, 2014)

Uvedená gesta budou sloužit k realizaci přechodů mezi stavy aplikace. Stavy aplikace a přechody mezi nimi jsou znázorněny stavovým diagramem na obrázku 17.

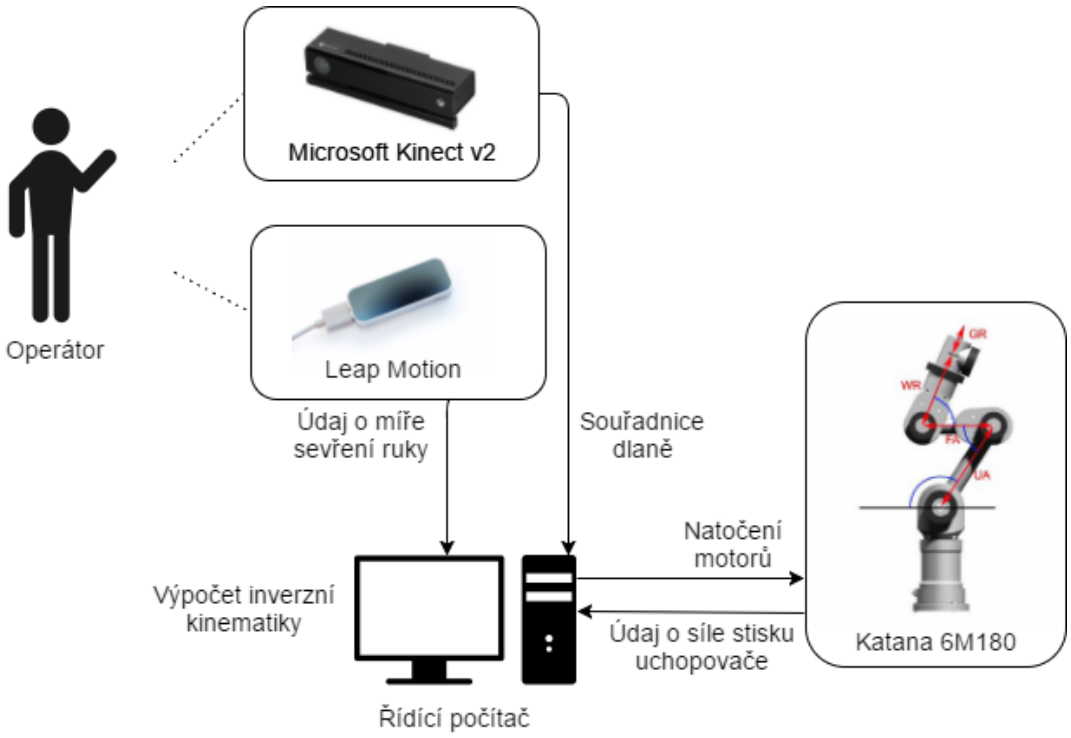


Obrázek 17: Stavový diagram aplikace

5.3 Návrh řídicí aplikace

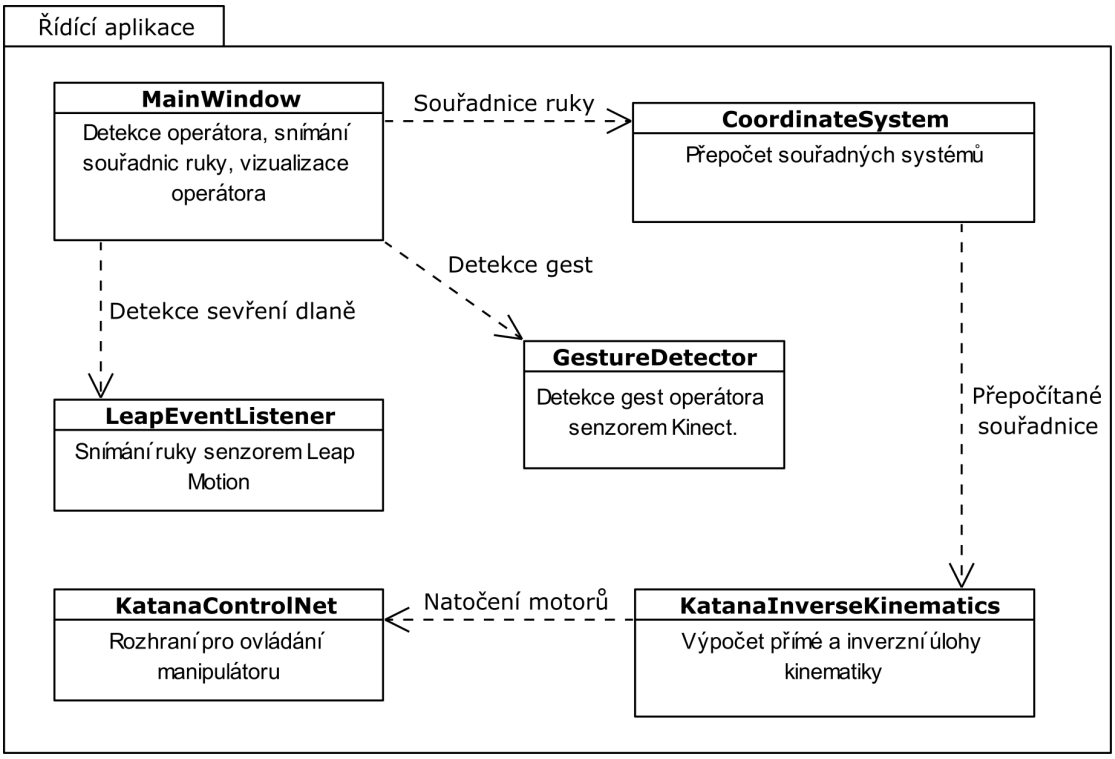
Na obrázku 18 je znázorněno schéma řídicího řetězce. Senzor Kinect detekuje gesta operátora a snímá souřadnice jeho dlaně v souřadném systému senzoru. Souřadnice jsou přepočítány do souřadného systému katany. Úhly natočení motorů manipulátoru jsou získány výpočtem inverzní úlohy kinematiky. Natočení motorů na tyto

úhly v reálném čase způsobí zrcadlení pohybu operátorovy dlaně koncovým efektorem manipulátoru. Senzor Leap Motion snímá míru sevření operátorovy ruky. Tento údaj je použit pro natočení motoru, který ovládá sevření uchopovače. Zároveň manipulátor poskytuje data ze senzorů vzdálenosti a senzorů síly umístěných na čelistech uchopovače.



Obrázek 18: Schéma řídicího řetězce

Na obrázku 19 je znázorněna funkcionalita navržených tříd. Třída `MainWindow` bude sloužit pro vykreslení GUI aplikace. Také zde bude sledován operátor senzorem Kinect a snímána poloha jeho ruky. Třída `GestureDetector` bude sloužit pro detekci gest operátora senzorem Kinect. Třída `LeapEventListener` bude sloužit pro snímání operátorovy ruky. Třídy `CoordinateSystem`, `KatanaInverseKinematics` a `KanataControlNet` budou převzaty ze stávajícího řešení (Kolaja, 2015).

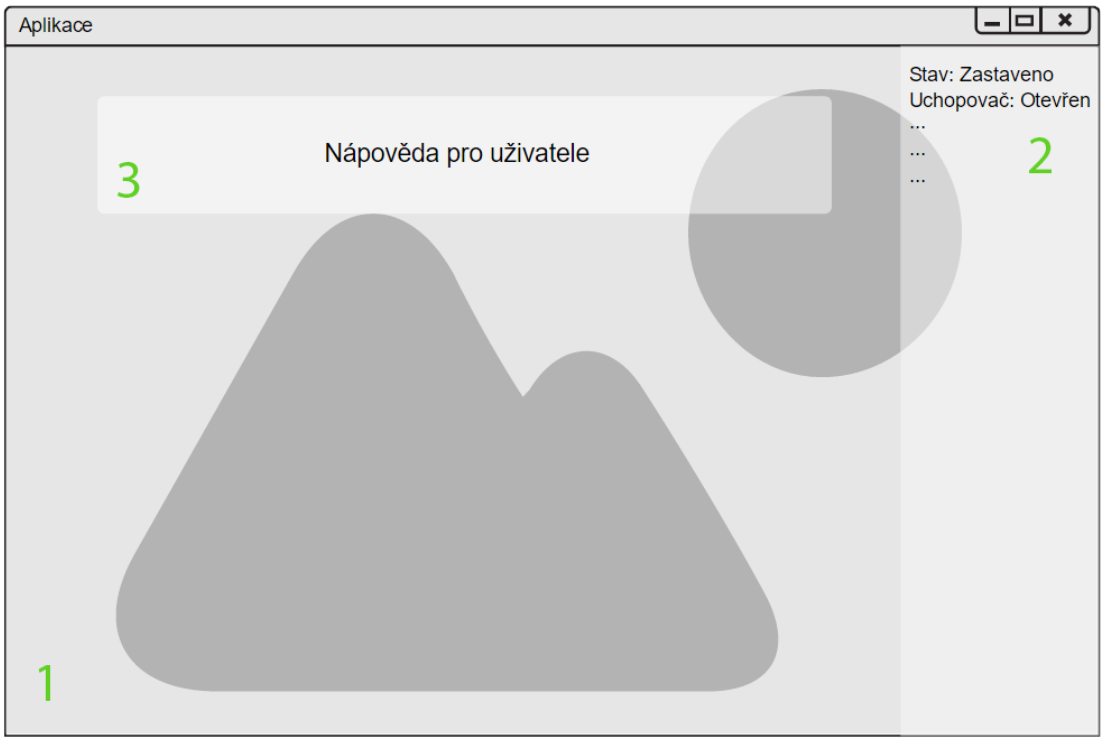


Obrázek 19: Funkcionalita navržených tříd

5.4 Návrh grafického uživatelského rozhraní

Řídicí aplikace musí mít grafické uživatelské rozhraní, aby mohla informovat uživatele o stavu manipulace. Pro navržené ovládání gesty (str. 35) je GUI taktéž vyžadováno, např. pro spuštění manipulace bude muset uživatel umístit svou ruku do vyznačené oblasti.

Navržené GUI je znázorněno pomocí drátového modelu na obrázku 20. Významnou část tvoří obraz z barevné kamery senzoru Kinect (na obrázku vyznačeno číslem 1). Do něho bude vykreslen skeleton operátora s 50% průhledností. Stejným způsobem budou do obrazu vykresleny i další prvky jako např. oblast pro spuštění manipulace nebo pro opětovné spuštění pozastavené manipulace. V pravé části je obraz z kamery částečně zakryt panelem s 50% průhledností pro vypisování stavových informací (na obrázku vyznačeno číslem 2), jako je např. údaj o stavu manipulace nebo údaj o stavu uchopovače. V horní části (na obrázku vyznačeno číslem 3) se nachází textové pole s 50% průhledností, které slouží jako nápověda pro uživatele.



Obrázek 20: Drátový model GUI

5.5 Implementace

Pro implementaci řídicí aplikace jsem zvolil jazyk C# a pro GUI technologii Windows Presentation Foundation (WPF) z frameworku .NET, což se jevilo jako uživatelsky nejpřívětivější možnost. Dalším důvodem byla vysoká četnost ukázek použití těchto technologií v oficiální dokumentaci SDK senzoru Kinect. Jazyk C# je současně podporován i SDK senzoru Leap Motion.

Vazby mezi implementačními třídami jsou znázorněny v diagramu tříd na obrázku B.1 na str. 64. Statické třídy jsou pak znázorněny na obrázku B.2 na str. 65.

Pro implementaci aplikace jsem nejprve musel nastudovat vývoj pro platformu .NET s technologií WPF. Poté jsem musel nastudovat použití oficiálního SDK verze 2.0 pro senzor Kinect a jeho možnosti. Totéž bylo potřeba provést i pro SDK verze 2.3.1 senzoru Leap Motion.

Jakmile jsem měl toto hotovo, začal jsem s implementací prototypu aplikace. Pro realizaci navrženého GUI aplikace (obr. 20) bylo potřeba získat stream snímků z barevné kamery senzoru Kinect a nastavit ho jako pozadí hlavního okna aplikace. Pro vykreslení skeletonu uživatele bylo potřeba získat souřadnice kloubů uživatele, na jejich pozicích vykreslit kruhy a propojit je úsečkami. Toto je implementováno v metodě `DrawBody` statické třídy `Drawing`. Pro vykreslení skeletonu do barevného snímku bylo zapotřebí namapovat souřadnice kloubů z trojrozměrného souřadného

systému Kinectu do dvourozměrného souřadného systému barevné kamery, jinak by byl skeleton v obraze vykreslen mimo tělo uživatele. K mapování souřadnic mezi prostory senzorů slouží třída `CoordinateMapper`, která je součástí SDK senzoru Kinect.

Poté jsem mohl začít implementovat funkcionalitu pro interakci s uživatelem. Protože senzor Kinect rozpoznává až 6 osob (instancí třídy `Body` z SDK) najednou, bylo nutné z vícero možných detekovaných osob zvolit jedinou. Jako osobu, které je umožněno ovládání, jsem zvolil tu, která je nejblíže senzoru Kinect v okamžiku spuštění aplikace. Zároveň dojde k uložení atributu `TrackingId` třídy `Body`, který jednoznačně identifikuje sledovanou osobu. Nemůže se tak stát, že by aplikaci začala ovládat jiná osoba, která se dostane blíže k senzoru Kinect.

Detekce gest

Možnost změny ruky pro manipulaci jsem implementoval jako gesto zdvižení ruky nad hlavu v metodě `SwitchHands` třídy `MainWindow`. Gesto je realizováno porovnáním souřadnic uživatelovy ruky a hlavy, přičemž hodnota souřadnice ruky musí být o 0,1 m větší, než hodnota souřadnice hlavy na ose y. K tomu slouží atribut `Joints` třídy `Body`, který obsahuje seznam objektů třídy `Joint`. K hodnotě souřadnic lze přistoupit následovně: `Joints[JointType.Head].Position.Y`. Zároveň je rozlišeno, kterou rukou uživatel gesto provedl (`JointType.HandLeft` nebo `JointType.HandRight`) a ta je poté nastavena pro ovládání. Pokud senzor Kinect detekuje více osob, toto gesto umožní ovládat aplikaci libovolné z nich.

Pro spuštění manipulace jsem zvolil gesto, ve kterém uživatel musí přesunout ruku do vyznačené oblasti a ponechat ji zde po dobu 2 sekund. Oblast je vykreslována jako kruh v oblasti uživatelova ramene. Gesto je detekováno, pokud souřadnice uživatelovy ruky leží uvnitř kruhu. K tomu slouží metoda `IsPointInsideEllipse`.

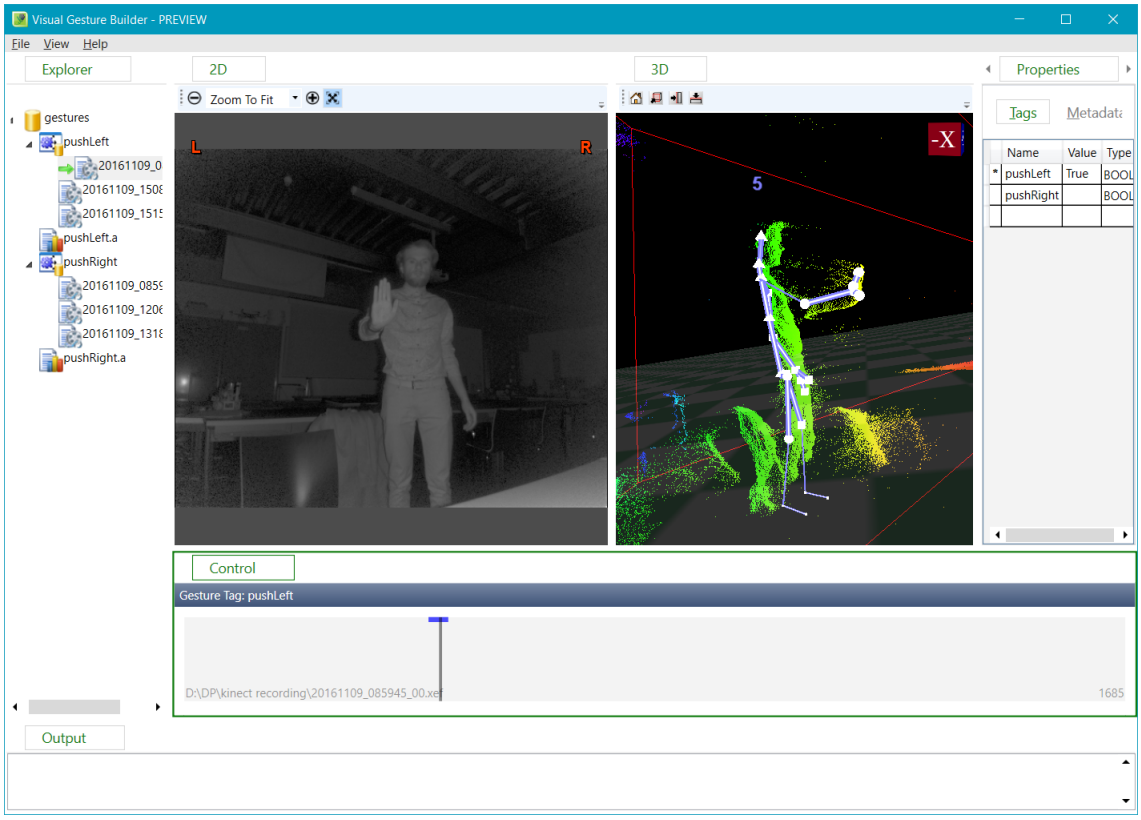
Při spuštěné manipulaci je pohyb uživatelovy ruky zrcadlen manipulátorem. K tomu je zapotřebí vypočítat inverzní úlohu kinematiky a na výsledné úhly natočit motory manipulátoru. K tomu slouží třídy `KatanaInversKinematics`, `CoordinateSystem` a `KatanaControlNet`, které jsou převzaty ze stávájícího řešení (Kolaja, 2015). Při manipulaci je detekováno gesto zatlačení rukou (obr. 16) pro pozastavení manipulace. Detekce gesta je vytvořena pomocí nástroje Visual Gesture Builder, který je popsán na str. 41. Pro detekci vytvořeného gesta slouží třídy `GestureDetector` a `GestureResult`. Při detekci se uloží pozice ruky, kterou uživatel prováděl manipulaci.

Gesto opětovného spuštění je implementováno obdobě jako to pro spuštění manipulace. Kontroluje se v něm, zda uživatel umístil ruku do stejné pozice ve všech třech osách jako před pozastavením manipulace s tolerancí 0,1 m.

Gesto pro ukončení manipulace je totožné s gestem pro její pozastavení. Aplikace však musí být ve stavu pozastavené manipulace, aby mohlo dojít k jejímu ukončení.

Visual Gesture Builder

Pro implementaci detekce složitějšího gesta zatlačení rukou jsem zvolil nástroj Visual Gesture Builder (VGB), který je součástí SDK od verze 2.0. Tento nástroj umožňuje vytvářet gesta pomocí strojového učení ze záznamů nahraných nástrojem Kinect Studio, který je rovněž součástí SDK. Pomocí Kinect Studia je potřeba vytvořit záznam požadovaného gesta a ve VGB do záznamu vyznačit, ve kterých snímcích je gesto prováděno a jakých hodnot nabývá. Je možno vytvářet gesta diskrétní (mají hodnoty *true* nebo *false*) nebo spojitá (mají hodnoty v intervalu $\langle 0; 1 \rangle$). Pro naučení diskrétních gest je použita metoda AdaBoost, pro naučení spojitých gest metoda Náhodného lesa pro regresi. Pro každé gesto je možné použít vícero záznamů a je možné zvolit, zda má být ignorována spodní část těla, pravá nebo levá paže a zda záleží na stavu dlaně. Na obrázku 21 je vidět uživatelské prostředí VGB a označení gesta v záznamu.



Obrázek 21: Visual Gesture Builder

Po naučení gest nástroj sestaví databázi gest do jednoho souboru (`*.gbd`), kterou je možno použít ve vlastní aplikaci k detekci gest. Při detekci diskrétního gesta SDK vyhodnotí *koeficient důvěryhodnosti* (interval $\langle 0; 1 \rangle$).

Ovládání uchopovače

Pro ovládání uchopovače je implementována detekce sevření / otevření dlaně pomocí senzoru Leap Motion v metodě `ControlGripper` třídy `MainWindow`. SDK senzoru umožňuje získat informaci o sevření ruky pomocí metody `GrabStrength` třídy `Hand`. Metoda vrací hodnoty v intervalu $\langle 0; 1 \rangle$. Pokud je tato hodnota větší nebo rovna nastavené prahové hodnotě 0.9, aplikace detekuje ruku jako zavřenou. Tím se v novém vlákně spustí v cyklu zavírání uchopovače manipulátoru. Při zavírání se kontroluje hodnota senzorů síly na čelistech uchopovače pomocí metody `GetGripperData` třídy `KatanaControlNet`. Metoda v parametru přebírá číslo senzoru, jehož hodnotu má vrátit.[3] Pokud alespoň jeden z těchto senzorů vrátí hodnotu větší než 223, je zavírání uchopovače zastaveno. Otevření uchopovače je realizováno natočením motoru č. 6 (ovládá uchopovač) na 2000 enkodérů pomocí metody `MoveMotorToEnc` třídy `KatanaControlNet`, do které jsou tyto hodnoty předány jako parametry.

Při implementaci se ukázalo, že při uchopování předmětu byl na koncový efektor přenášen třes z ruky snímané senzorem Kinect, který znemožňoval přesné uchopení předmětu. Třes mohl být způsobován i nedokonalým snímáním ruky senzorem Kinect. Z tohoto důvodu jsem implementoval zastavení pohybu manipulátoru, pokud je senzorem Leap motion detekována ruka. Tato funkcionalita je implementována v metodě `DetectHand` třídy `MainWindow`.

5.6 Popis aplikace

Po spuštění aplikace se musí nejprve provést kalibrace manipulátoru, která trvá přibližně jednu minutu[4]. Na obrázku 22 je zobrazen screenshot GUI aplikace při probíhající kalibraci. Screenshot je k dispozici v plné velikosti na obr. C.1 na str. 67. Je zde také vidět vykreslený skeleton operátora, textové pole v horní části zobrazující pokyny pro uživatele („Please wait for Katana calibration.“) a panel vpravo zobraující stavové informace.

[3] Senzory síly mají čísla 6, 7, 14 a 15 (obr. 14 na str. 31).

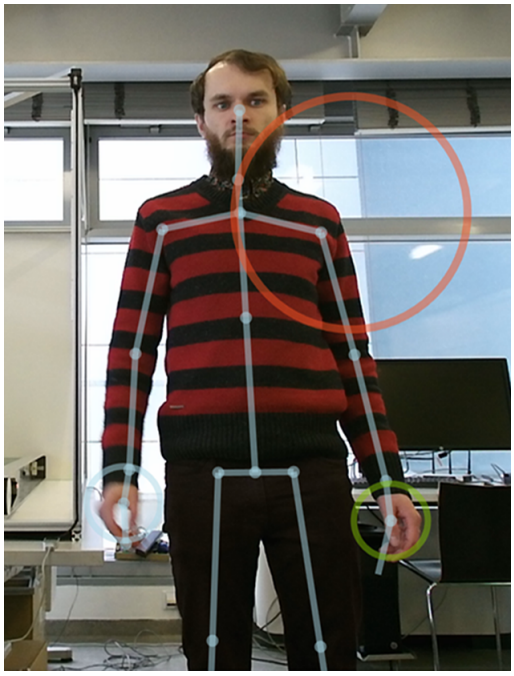
[4] Doba kalibrace závisí na úhlech natočení jednotlivých motorů manipulátoru před jejím spuštěním.

Obrázek 22: Kalibrace manipulátoru (screenshot GUI)

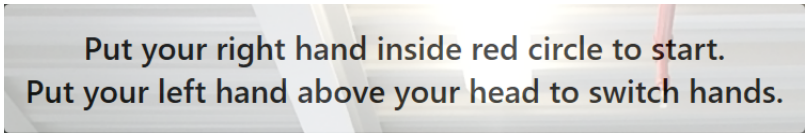
Jakmile je kalibrace hotova, manipulátor se přesune do výchozí polohy a aplikace se přepne do stavu zastavené manipulace. Změny v GUI jsou znázorněny na obrázku 23 (screenshot je k dispozici v plné velikosti na obrázku C.2 na str. 68). Ve vykresleném skeletonu operátora je zeleně vyznačena ruka určená pro ovládání manipulace a červenou kružnicí je vyznačena oblast pro spuštění manipulace (obr. 23a). Textové pole s pokyny pro uživatele zobrazuje popis spuštění manipulace a případné změny ruky pro ovládání (obr. 23b). Informace ve stavovém panelu jsou aktualizovány (obr. 23c).

Pokud je operátor detekován senzorem Kinect, je mu umožněno spuštění manipulace. Pokud by se v dosahu senzoru Kinect nacházelo více osob[5], je ovládání umožněno té, která je nejblíže senzoru. Vzdálenější osoby mají možnost převzít kontrolu nad ovládáním tím, že zvednou svoji ruku nad hlavu. Tato ruka bude poté zároveň označena jako hlavní pro ovládání manipulace. Takto lze kontrolu nad ovládáním převzít jen v případě, že je manipulace zastavena nebo pozastavena.

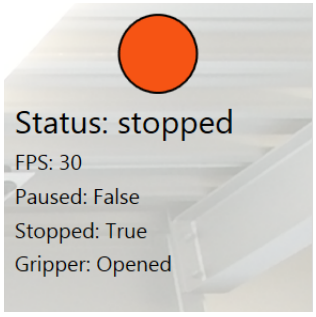
[5]Senzor Kinect verze 2 je schopen současně detekovat až 6 osob.



(a) Vizualizace oblasti pro spuštění a aktivní ruky



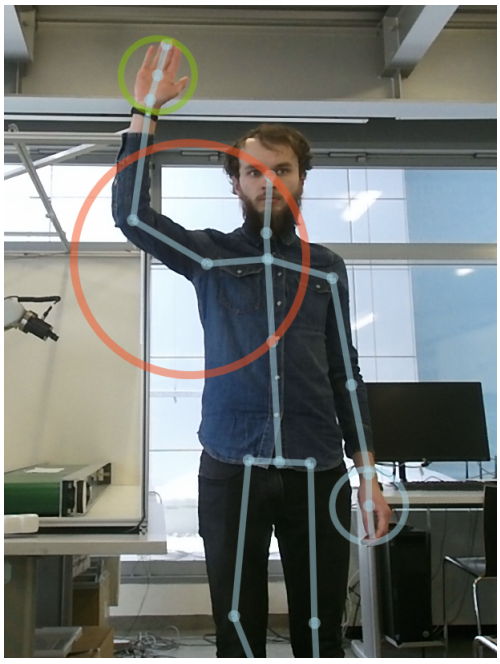
(b) Pokyn pro uživatele



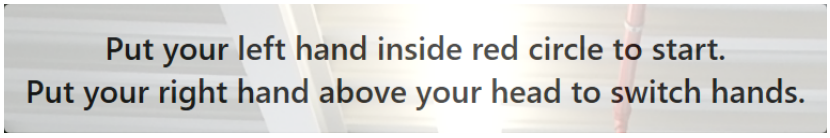
(c) Stavové informace

Obrázek 23: Zastavená manipulace (výřezy)

Pro ovládání manipulace je ve výchozím stavu použita pravá ruka. Ruku pro ovládání může operátor změnit tím, že zvedne požadovanou ruku nad hlavu. Změnu ruky lze provést jen při zastavené nebo pozastavené manipulaci. Změna ruky je znázorněna na obrázku 24 (screenshot je k dispozici v plné velikosti na obr. C.3 na str. 69). Aktivní ruka je barevně odlišena zelenou kružnicí, oblast pro spuštění manipulace je vykreslena na odpovídající straně skeletonu (obr. 24a) a pokyny pro uživatele jsou aktualizovány (obr. 24b). Stejné gesto je použito i pro výše zmíněné převzetí kontroly v případě, že je detekováno více osob.



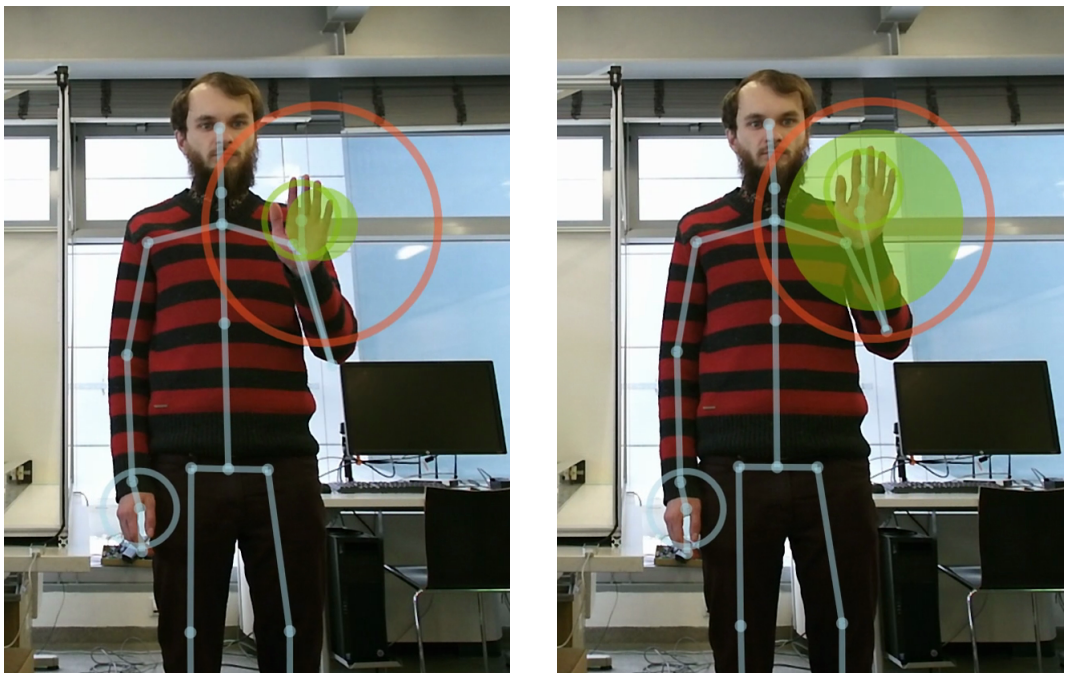
(a) Změna oblasti pro spuštění a aktivní ruky



(b) Pokyny pro uživatele

Obrázek 24: Změna ruky pro manipulaci (výřezy)

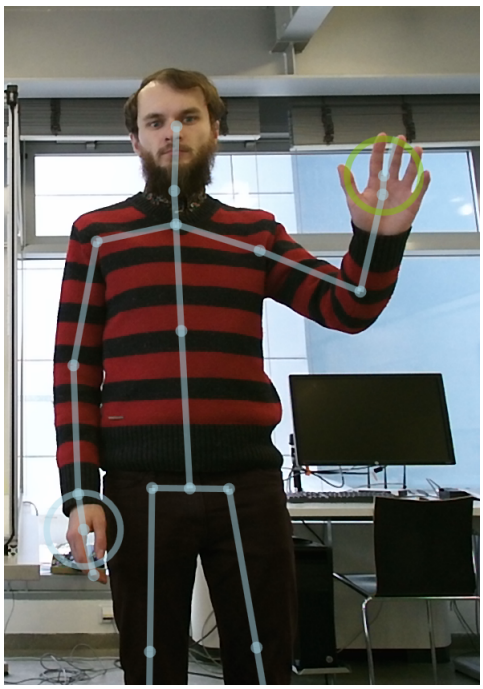
Spuštění manipulace provede operátor tak, že přesune zvolenou ruku do vyznačené oblasti (lze vidět na obr. 23a, 24a a 25). Tato oblast je vykreslena v závislosti na zvolené ruce v oblasti buď pravého, nebo levého ramene. Pro spuštění manipulace musí operátor ponechat ruku v této oblasti po dobu 2 sekund. Odpočet je indikován stejně dlouho trvající animací zvětšujícího se zeleného kruhu od středu červené kružnice až k jejímu okraji. Animace je znázorněna pomocí obrázků 25a a 25b. Po uplynutí 2 sekund je operátor upozorněn zvukovým signálem a manipulace je spuštěna. Pokud by operátor během odpočtu přesunul ruku mimo vyznačenou oblast, je odpočet zastaven a vynulován.



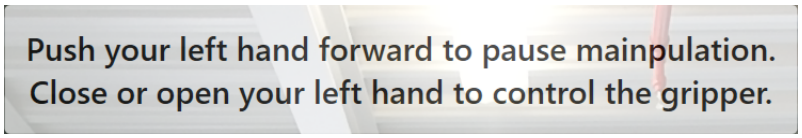
(a) Umístění ruky do vyznačené oblasti (b) Indikace spouštění

Obrázek 25: Spuštění manipulace (screenshot GUI)

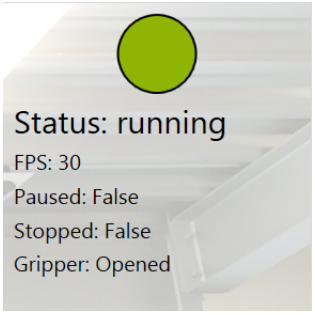
Změny v GUI při probíhající manipulaci jsou zachyceny na obrázku 26 (screenshot je k dispozici v plné velikosti na obr. C.4 na str. 70). Během manipulace je pozice operátorovy ruky zrcadlena koncovým efektorem manipulátoru.



(a) Vizualizace aktivní ruky



(b) Pokyn pro uživatele



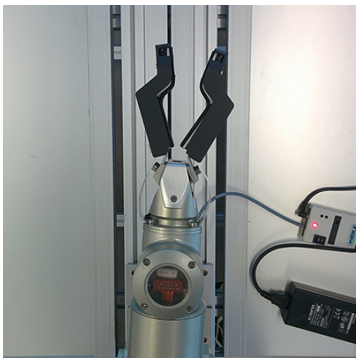
(c) Stavové informace

Obrázek 26: Probíhající manipulace (výřezy)

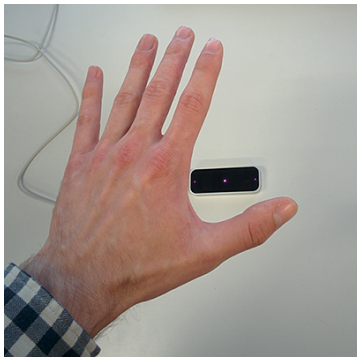
Během manipulace má operátor možnost otevírat nebo zavírat uchopovač manipulátoru. Ovládání uchopovače je znázorněno na obrázku 27. Ve výchozím stavu je uchopovač otevřen. K zavření uchopovače je zapotřebí umístit druhou ruku přibližně do výšky 10 cm nad senzorem Leap Motion a sevřít dlaň (obr. 27a a 27b). K otevření uchopovače je potřeba dlaň nad senzorem zase otevřít (obr. 27c a 27d).

V okamžiku, kdy je ruka detekována senzorem Leap Motion, dojde k pozastavení pohybu manipulátoru (nikoliv k pozastavení manipulace). Jakmile je ruka mimo senzor (není detekována), je pohyb manipulátoru obnoven. Pro ovládání gripperu je nutné použít ruku opačnou, než která je použita pro manipulaci. Pokud je pro manipulaci použita pravá ruka, aplikace očekává nad senzorem Leap Motion levou ruku a pravá ruka by byla ignorována.

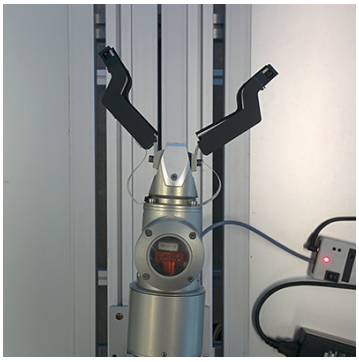
(a) Zavřená dlaň nad senzorem



(b) Zavřený uchopovač



(c) Otevřená dlaň nad senzorem



(d) Otevřený uchopovač

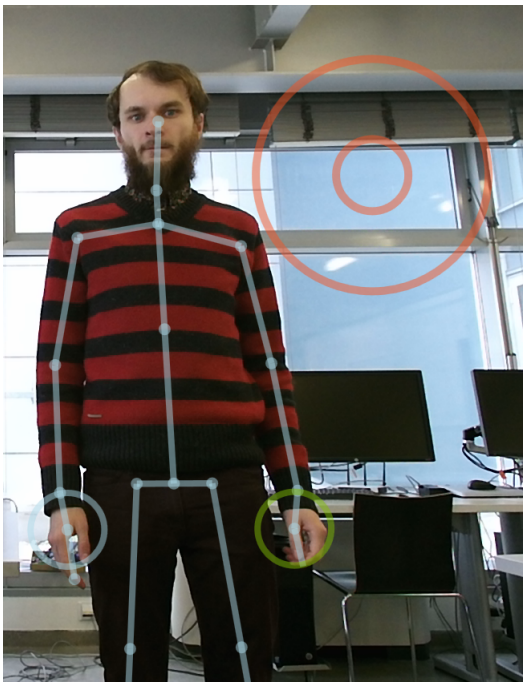
Obrázek 27: Otevření / zavření uchopovače

V případě, že by při probíhající manipulaci přestal být operátor detekován senzorem Kinect (ztratí se z dosahu senzoru, dojde k chybě senzoru, atd.), je manipulace zcela zastavena, aplikace se přepne do stavu zastavené manipulace, manipulátor se vrátí do výchozí polohy a uchopovač se otevře.

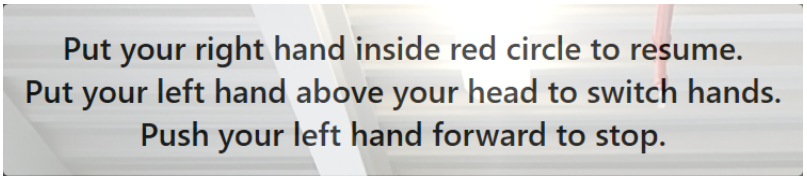
Manipulaci operátor pozastaví provedením gesta zatlačení (obr. 16 na str. 36) opačnou rukou, než kterou používá pro manipulaci. Detekcí gesta se aplikace přepne do stavu pozastavené manipulace. V ní zůstává manipulátor v pozici, ve které byl při detekování gesta. Pozastavená manipulace a změny v GUI jsou zachyceny na obrázku 28 (screenshot je k dispozici v plné velikosti na obr. C.5 na str. 71). Na pozici, ve které se před pozastavením nacházela operátorova ruka pro ovládání manipulace, je vykreslena červená kružnice (obr. 28a). Pokyny pro uživatele (obr. 28b) a informace ve stavovém panelu (obr. 28c) jsou aktualizovány.

Pozastavenou manipulace může operátor ukončit nebo jí znovu spustit. Pro opětovné spuštění je potřeba zvolenou ruku přesunout do stejné pozice, ve které byla před pozastavením s přesností na 0,1 m v osách x, y a z. Pro indikaci pozice v osách x a y slouží červená kružnice. Indikace v ose z (obr. 29) je provedena pomocí textových popisků „Too near“ v případě, že je operátorova ruka moc blízko (obr.

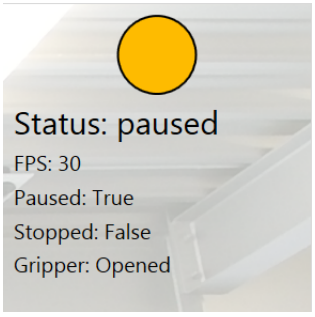
29a), a „Too far“ v případě, že je moc daleko (obr. 29b).



(a) Vizualizace oblasti pro opětovné spuštění a aktivní ruky

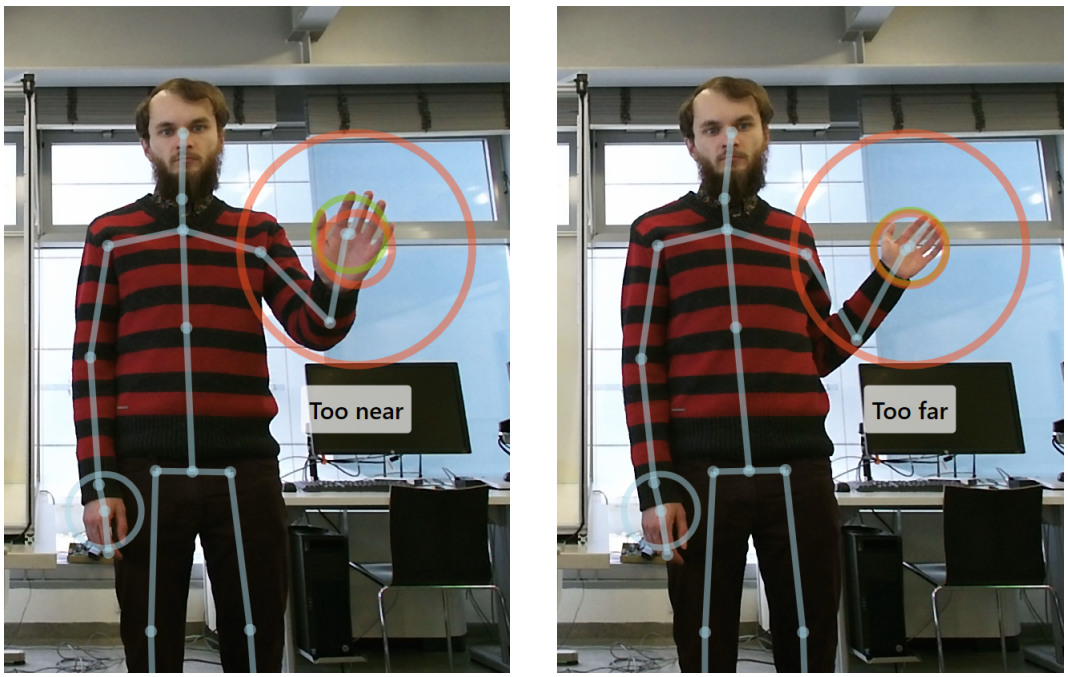


(b) Pokyn pro uživatele



(c) Stavové informace

Obrázek 28: Pozastavená manipulace (výřezy)



(a) Dlaň operátora příliš blízko (b) Dlaň operátora příliš daleko

Obrázek 29: Indikace umístění ruky do přesné pozice (screenshot GUI)

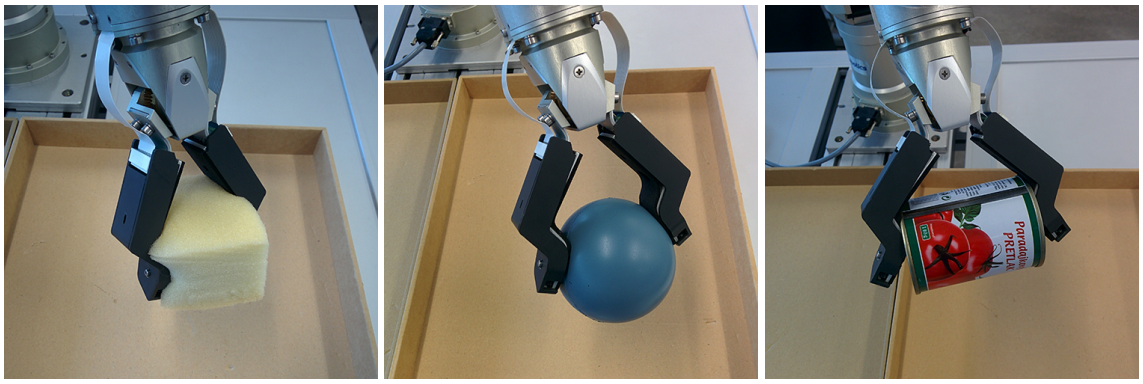
Pro ukončení manipulace je potřeba ji nejdříve pozastavit. K ukončení je potřeba znovu provést gesto zatlačení rukou. Tím se aplikace přepne do stavu zastavené manipulace, manipulátor se vrátí do výchozí polohy a uchopovač se otevře.

Pohled na operátora při práci s vytvořeným uživatelským rozhraním je zachycen na obrázku D.1 na str. 72.

5.7 Kontrola síly stisku uchopovače

V aplikaci je implementována kontrola síly, kterou působí čelisti uchopovače na uchopovaný předmět. Pokud by uchopovač při zavírání působil na předmět příliš velkou silou a hrozilo by poškození předmětu nebo uchopovače, je zavírání ukončeno. Maximální síla, kterou může uchopovač na předmět působit, je zvolena orientačně na základě experimentů. Pro určení přesné síly by bylo potřeba zkalibrovat snímače síly umístěné na čelistech uchopovače (obr. 14 na str. 31).

Na obrázku 30 je vyfoceno sevření předmětů z různě deformovatelných materiálů. Je zde vidět rozdílná deformace předmětů při působení síly stejné velikosti na kostku z molitanu (obr. 30a), míček (obr. 30b) a konzervu (obr. 30c).



(a) Kostka z molitanu (b) Míček (c) Konzerva

Obrázek 30: Sevření předmětů z různých materiálů

6 Testování

V této kapitole je popsáno testování aplikace, jehož cílem bylo ověřit správnost navrženého rozhraní a funkčnost detekce gest na různých operátorech. Zjištěné poznatky byly použity pro vylepšení aplikace a detekce gest. Testování probíhalo ve dvou fázích. Při testování měl každý tester za úkol desetkrát spustit manipulaci a poté ji gestem levé ruky pozastavit. Totéž bylo opakováno i pro druhou ruku.

6.1 První fáze

Získané údaje z první fáze testování jsou vedeny v tabulce 3 a úspěšnost testů je uvedena v tabulce 4. Měřenou hodnotou byl *koeficient důvěryhodnosti*, který představuje úroveň spolehlivosti detekce gesta (Microsoft, 2017b). V tabulce je zaokrouhlen na dvě desetinná místa. Prahová hodnota tohoto parametru byla nastavena na 0,75 včetně. Červeně jsou v tabulce označeny buňky s hodnotou koeficientu důvěrnosti nižší, než je prahová hodnota. Detekovaná gesta s hodnotou koeficientu důvěryhodnosti pod jeho prahovou hodnotou nebyla uznána jako platná.

Tabulka 3: Výsledky první fáze testování detekce gest

Gesto zatlačení levou rukou				
Tester	1	2	3	4
Pokus	K. d.	K. d.	K. d.	K. d.
1.	0,44	1	1	1
2.	0,49	0,86	0,99	0,96
3.	0,83	1	1	1
4.	0,84	1	1	1
5.	0,81	1	1	0,99
6.	0,97	1	1	1
7.	0,97	1	0,41	0,89
8.	0,74	1	1	1
9.	0,6	1	0,4	1
10.	1	1	0,7	1

Gesto zatlačení pravou rukou				
Tester	1	2	3	4
Pokus	K. d.	K. d.	K. d.	K. d.
1.	0,64	0,79	0,92	0,78
2.	0,87	0,58	0,78	0,6
3.	0,43	0,85	0,73	0,24
4.	0,73	0,98	1	0,63
5.	0,65	0,85	0,87	0,83
6.	1	0,76	0,93	0,98
7.	0,79	1	0,73	0,88
8.	1	1	0,84	0,79
9.	0,51	1	0,43	0,8
10.	0,36	0,64	0,59	0,77

Tabulka 4: Úspěšnost první fáze testů

Gesto	Počet detekovaných gest	Procentuální podíl
Zatlačení levou rukou	33/40	82,5 %
Zatlačení pravou rukou	25/40	62,5 %

6.2 Druhá fáze

Testování ve druhé fázi probíhalo stejným způsobem, jako v té první. Pro detekci gest byla použita přepracovaná databáze gest. Výsledky testování jsou uvedeny v tabulce 5 a úspěšnost testů je uvedena v tabulce 6. Prahová hodnota koeficientu důvěryhodnosti byla po první fázi testování snížena na 0,5. Červeně jsou v tabulce označeny buňky s hodnotou koeficientu důvěrnosti pod prahovou hodnotou. Detekovaná gesta s hodnotou koeficientu důvěryhodnosti pod jeho prahovou hodnotou nebyla uznána jako platná.

Tabulka 5: Výsledky druhé fáze testování detekce gest

Gesto zatlačení levou rukou						
Tester	1	2	3	4	5	6
Pokus	K. d.	K. d.	K. d.	K. d.	K. d.	K. d.
1.	0,31	0,47	0,67	0,81	0,81	0,58
2.	0,41	0,53	0,51	0,61	0,71	0,38
3.	0,57	0,54	0,52	0,69	0,57	0,3
4.	0,46	0,51	0,62	0,9	0,53	0,39
5.	0,31	0,52	0,52	0,71	0,57	0,42
6.	0,54	0,57	0,8	0,62	1	0,68
7.	0,54	0,55	0,68	0,71	34	0,42
8.	0,69	0,42	0,84	0,72	0,69	0,67
9.	0,36	0,49	0,74	0,62	0,42	0,46
10.	0,37	0,52	0,71	0,91	0,58	0,21

Gesto zatlačení pravou rukou						
Tester	1	2	3	4	5	6
Pokus	K. d.	K. d.	K. d.	K. d.	K. d.	K. d.
1.	0,63	0,94	0,98	0,53	0,51	0,75
2.	0,29	0,92	0,82	0,42	0,58	0,85
3.	0,58	0,52	0,93	0,53	0,8	0,64
4.	0,53	0,97	0,89	0,78	0,59	0,42
5.	0,49	0,92	1	0,49	0,87	0,42
6.	0,47	0,82	1	0,64	0,53	0,6
7.	0,48	0,81	1	0,73	0,51	0,46
8.	0,29	0,81	1	0,8	0,28	0,73
9.	0,45	0,95	0,99	0,71	0,52	0,8
10.	0,37	0,92	0,92	0,69	0,45	0,72

Tabulka 6: Úspěšnost druhé fáze testů

Gesto	Počet detekovaných gest	Procentuální podíl
Zatlačení levou rukou	42/60	70 %
Zatlačení pravou rukou	46/60	$76,\overline{66}$ %

6.3 Vyhodnocení provedených testů

Po otestování se čtyřmi testery v první fázi se detekce gesta zatlačení pravou rukou jevila jako nedostatečná kvůli častému selhávání, které je patrné z tabulky 3, a nemělo tudíž smysl dále navyšovat počet testerů. Na základě výsledků první vlny testování byla přepracována databáze gest, která slouží pro detekci. Gesta byla kromě mě naučena na dalších třech lidech, což mělo detekci gest více generalizovat. Aby byla zajištěna objektivita testů, byli tito lidé vyloučeni z druhé fáze testování.

Prahová hodnota *koeficientu důvěryhodnosti*, která určuje, zda je detekce gesta uznána jako platná, byla za účelem generalizace po první fázi testování snížena z 0,75 na 0,5 včetně. Tato prahová hodnota musí být nastavena tak, aby detekce gest tolerovala odchylky od vytvořeného gesta (nepřesnosti), a zároveň aby nedocházelo k falešným detekcím. K falešným detekcím by docházet nemělo, protože vytvořená detekce gest rozlišuje mezi pravou a levou rukou. Riziko, že by uživatel při manipulaci omylem provedl gesto tak lze eliminovat a prahová hodnota *koeficientu důvěryhodnosti* může být nastavena na nižší hodnotu.

Testování aplikace ve dvou vlnách pomohlo k implementaci detekce gest s dostatečnou úrovní úspěšnosti. Minimální naměřená úspěšnost detekce je nyní 70 %. Ukázalo se, že každý z testerů provádí gesto zatlačení rukou mírně odlišně, ať už co do pozice dlaně nebo rychlosti provedení. Toto by mělo být eliminováno naučením gesta pomocí nástroje VGB (viz str. 41) na více lidech.

7 Diskuze

Vytvořené uživatelské rozhraní umožňuje ovládat manipulátor přirozeným pohybem operátorovy ruky. Ovládání je umožněno i uživatelům bez expertních znalostí, což se potvrdilo provedenými testy.

Oproti předchozímu řešení je uživateli umožněno řídicí aplikaci ovládat pohodlně gesty. Pro spuštění manipulace musí uživatel umístit ruku do pozice vyznačené v GUI. Pro její pozastavení provede gesto zatlačení rukou. Pro opětovné spuštění je potřeba umístit ruku do pozice před pozastavením, která je indikována v GUI. Manipulaci zastaví uživatel provedením gesta zatlačení rukou při pozastavené manipulaci. Pro zavření nebo otevření uchopovače stačí uživateli zavřít nebo otevřít ruku. Uživatel tak není nucen stále sledovat GUI aplikace a může se plně soustředit na manipulaci (kromě spuštění nebo opětovného spuštění manipulace). V GUI aplikce je vykreslen obraz z barevné kamery senzoru Kinect. Do obrazu z kamery je vykreslen skeleton pro každou detekovanou osobu a prvky pro ovládání manipulace. GUI zobrazuje i pokyny pro ovládání manipulace a slouží tak i jako podpora pro uživatele. Nové řešení umožňuje uživateli práci s manipulátorem i v případě, že senzor Kinect detekuje vícero osob, což zamezuje nechtěnému převzetí kontroly nad manipulací jinou osobou. Uživatelská přívětivost vytvořeného rozhraní je podpořena i možností operátora zvolit si ruku pro ovládání manipulace. Nově navíc přibyla kontrola síly stisku uchopovače. Tato funkcionalita má za úkol zamezit poškození uchopovače nebo uchopovaného předmětu. Pro určení velikosti působené síly by ale bylo potřeba senzory síly umístěné na čelistech uchopovače zkalibrovat.

Jako nežádoucí se ukázal být jev, kdy při zrcadlení ruky operátora koncovým efektorem docházelo i k přenosu otřesů ruky. Přesnost při manipulaci tak byla snížena. Toto však mohlo být způsobeno i nedokonalým snímáním polohy operátorovy ruky senzorem Kinect. Možným vylepšením v tomto směru by byla implementace algoritmu, který by dokázal otřesy alespoň částečně eliminovat.

Možným vylepšením by byla implementace funkce pro učení trajektorie manipulátoru. Operátor by nejprve trajektorii definoval provedením pomocí ruky a manipulátor by ji poté dokázal opakovat. Pro tuto funkcionalitu by ale bylo nutné definovat gesta, která by sloužila pro její ovládání.

Pro ovládání manipulátoru by bylo možné použít i samotný senzor Leap Motion. Problémem by však byl efektivní dosah senzoru, který je menší než pracovní obálka manipulátoru. To by ale bylo možné vyřešit např. změnou měřítka mezi souřadným systémem senzoru a manipulátoru. Bylo by však potřeba zásadně přepracovat vytvořené uživatelské rozhraní.

8 Shrnutí

Práce se zabývá problematikou návrhu a implementace uživatelského rozhraní pro řízení robotického manipulátoru pomocí detekce gest operátora.

Vytvořené rozhraní se skládá ze senzorů Microsft Kinect verze 2 a Leap Motion pro detekci operátora a jeho gest, z manipulátoru Katana 6M180 a z počítače, na kterém běží řídicí aplikace. Řídicí aplikace je napsána v jazyce C#. GUI aplikace je vyvinuto ve frameworku .NET s využitím technologie WPF.

Kapitola Současný stav se zabývá popisem existujících nekonvenčních uživatelských rozhraní pro řízení manipulátoru. Jsou zde zejména uvedeny různé přístupy realizace rozhraní mezi uživatelem a manipulátorem. Na závěr kapitoly jsou tato řešení zhodnocena.

V kapitole Použité technologie je uveden popis použitých technologií v této práci. Jsou zde popsány použité senzory pro snímání lidského těla a pro detekci gest, jejich technické parametry a jejich možnosti pro dosažení cíle práce. Konkrétně se jedná o senzory Microsoft Kinect verze 2 a Leap Motion. Dále je zde popsán použitý manipulátor Katana 6M180.

Kapitola Realizace popisuje návrh a implementaci uživatelského rozhraní. Nejprve je provedena analýza požadavků, které vyplývají ze zadání této práce. Další požadavky jsou kladeny na bezpečnost uživatele a okolí. Poté následuje představení navrženého způsobu ovládání manipulátoru pomocí gest. Další část této kapitoly se zabývá návrhem řídicí aplikace, popisuje roli jednotlivých komponent v navrhovaném uživatelském rozhraní a roli tříd řídicí aplikace. GUI aplikace je navrženo pomocí drátového modelu. Druhá polovina této kapitoly se věnuje implementaci řídicí aplikace a popisu vytvořeného rozhraní z pohledu uživatele. Je zde popsán postup při implementaci aplikace včetně vytvořených tříd a jejich metod. Popis aplikace z pohledu uživatele podrobně vysvětluje použití vytvořeného uživatelského rozhraní a může sloužit jako manuál pro uživatele. Závěrem kapitoly je popsán způsob kontroly síly stisku uchopovače.

Testování je popsáno ve stejnojmenné kapitole. Jsou zde uvedeny podmínky, za jakých testování probíhalo a co bylo jejich cílem. V první fázi došlo k otestování vytvořeného rozhraní na čtyřech lidech. Na základě výsledků z první fáze došlo k přepracování detekce gest. Modifikovaná verze uživatelského rozhraní byla poté testována ve druhé fázi testů, které se účastnilo dalších šest osob. V kapitole jsou v tabulkách uvedeny výsledky z první a druhé fáze testování. Na závěr kapitoly jsou výsledky testů vyhodnoceny.

Kapitola Diskuze zmiňuje výhody a nevýhody vytvořeného řešení oproti tomu stávajícímu. Dále jsou zde navržena možná vylepšení.

9 Závěr

Cílem této práce bylo navrhnout, implementovat a otestovat uživatelské rozhraní pro řízení polohy koncového efektoru stacionárního manipulátoru Katana 6M180 s využitím detekce gest operátora.

Cíl práce byl splněn, operátor je schopen ovládat manipulátor prostřednictvím vytvořeného rozhraní, což bylo ověřeno testy. K detekci gest jsou použity senzory Microsoft Kinect verze 2 a Leap Motion. Minimální úspěšnost detekce gest naměřená při testování byla 70 %. Řídicí aplikace vytvořeného rozhraní navíc kontroluje sílu stisku uchopovače tak, aby nedošlo k jeho poškození nebo k poškození uchopovaného předmětu.

10 Literatura

Capi, G., Hossain, D., Kaneko, S., Bode, K. *Teaching Industrial Robot Manipulators by Easy to Use Interface Systems*. Switzerland: Trans Tech Publications, 2015, 2015(798), 3-7. ISSN 16627482.

Du, G., Zhang, P. *A novel human–manipulators interface using hybrid sensors with Kalman filter and particle filter*. [online]. Guangzhou, 2015 [cit 2017-02-07]. Dostupné z: http://www.sciencedirect.com/science/article/pii/S0736584515001192/pdfft?md5=72a8ce80a8178796e367aeb6d52bea69&pid=1-s2.0-S0736584515001192-main.pdf.

Du, G., Zhang, P. *Markerless human–robot interface for dual robot manipulators using Kinect sensor*. [online]. Guangzhou, 2013 [cit 2017-02-09]. Dostupné z: http://www.sciencedirect.com/science/article/pii/S0736584513000628/pdfft?md5=9c5d8164371d02f4fd17088d1da5495c&pid=1-s2.0-S0736584513000628-main.pdf.

Du, G., Zhang, P., Liu, X. *Markerless Human–Manipulator Interface Using Leap Motion With Interval Kalman Filter and Improved Particle Filter*. IEEE Transactions on Industrial Informatics [online]. 2016, 12(2), 694-704 [cit. 2017-05-05]. DOI: 10.1109/TII.2016.2526674. ISSN 1551-3203. Dostupné z: http://ieeexplore.ieee.org/document/7401061/.

Fernandes, V. B. P., Frank, J. A., Kapila, V. *A wearable interface for intuitive control of robotic manipulators without user training*. [online]. Copenhagen, 2014 [cit 2017-02-07]. Dostupné z: https://www.researchgate.net/profile/Jared_Frank/publication/274711342_A_Wearable_Interface_for_Intuitive_Control_of_Robotic_Manipulators_Without_User_Training/links/5745cf4c08aea45ee8560da6.pdf.

Janda, M. *Realizace komunikačního rozhraní pro robotický manipulační systém*. [online]. Brno, 2013 [cit. 2017-03-20]. Dostupné z: https://is.mendelu.cz/zp/portal_zp.pl?podrobnosti_zp=38316;zp=38316;download_prace=1. Diplomová práce. Mendelova univerzita v Brně, Provozně ekonomická fakulta. Vedoucí práce Oldřich Trenz.

Kolaja, J. *Bezdotykové řízení polohy koncového efektoru manipulátoru*. [online]. Brno, 2015 [cit. 2017-01-19]. Dostupné z: http://is.mendelu.cz/zp/portal_zp.pl?podrobnosti=70895;zp=48492;download_prace=1. Diplomová práce. Mendelova univerzita v Brně, Provozně ekonomická fakulta. Vedoucí práce Vít Ondroušek.

Leap Motion. *API Overview*. C# SDK Documentation [online]. Leap Motion, 2015 [cit. 2017-03-08]. Dostupné z: https://developer.leapmotion.com/documentation/v2/csharp/devguide/Leap_Overview.html.

Leap Motion. *Gestures*. C# SDK Documentation [online]. Leap Motion, 2015 [cit. 2017-03-19]. Dostupné z: https://developer.leapmotion.com/documentation/v2/csharp/devguide/Leap_Gestures.html.

Leap Motion 3D Motion Sensor. Nepal's Online Shopping Portal [online]. 2017 [cit. 2017-03-19]. Dostupné z: http://www.gufshop.com/shop/virtual-reality/leap-motion-3d-motion-sensor/.

Li, T., Hong, J., Zhang, J. *Brain–machine interface control of a manipulator using small-world neural network and shared control strategy*. [online]. Xi'an, 2013 [cit 2017-02-07]. Dostupné z: http://www.sciencedirect.com/science/article/pii/S0165027013003981/pdfft?md5=c0cb94559ead08187d441c1a1eabc3a2&pid=1-s2.0-S0165027013003981-main.pdf.

Microsoft. *Developing with Kinect for Windows*. Windows Dev Center [online]. Microsoft, 2017 [cit. 2017-02-28]. Dostupné z: https://developer.microsoft.com/en-us/windows/kinect/develop.

Microsoft. *DiscreteGestureResult.Confidence Property*. Microsoft API and reference catalog [online]. Microsoft, 2017 [cit. 2017-05-10]. Dostupné z: https://msdn.microsoft.com/en-us/library/microsoft.kinect.visualgesturebuilder.discretegestureresult.confidence.aspx.

Microsoft. *Kinect for Windows SDK*. Microsoft API and reference catalog [online]. Microsoft, 2017 [cit. 2017-03-07]. Dostupné z: https://msdn.microsoft.com/en-us/library/dn799271.aspx.

Microsoft. *Kinect hardware*. Windows Dev Center [online]. Microsoft, 2017 [cit. 2017-03-07]. Dostupné z: https://developer.microsoft.com/en-us/windows/kinect/hardware.

Microsoft. *Microsoft Speech Platform SDK 11 Documentation*. Microsoft API and reference catalog [online]. Microsoft, 2017 [cit. 2017-03-07]. Dostupné z: https://msdn.microsoft.com/en-us/library/dd266409(v=office.14).aspx.

Neuronics AG. *Návod k použití robotů Katana*. 2006, 62 s.

Oka, T., Matsushima, K. *New design of a manipulator control interface that employs voice and multi-touch command*. In: 2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI) [online]. IEEE, 2016, s. 487-488 [cit. 2017-05-06]. DOI: 10.1109/HRI.2016.7451819. ISBN 978-1-4673-8370-7. Dostupné z: http://ieeexplore.ieee.org/document/7451819/.

Paredes-Madrid, L., Gonzalez de Santos, P. *Dataglove-based interface for impedance control of manipulators in cooperative human–robot environments*. Measurement Science and Technology [online]. 2013, 24(2), 025005- [cit. 2017-05-07]. DOI: 10.1088/0957-0233/24/2/025005. ISSN 0957-0233. Dostupné z: http://stacks.iop.org/0957-0233/24/i=2/a=025005?key=crossref.1c0cf4d2683ec1554dbc6b127ab41bba.

Robotic-Interaction Hand Gesture Illustrations. Simple-ism [online]. 2014 [cit. 2017-03-28]. Dostupné z: http://simple-ism.net/robotic-interaction-hand-gesture-illustrations/.

Vytečka, M. *Šachový automat s využitím manipulátoru Katana*. [online]. Brno, 2013 [cit. 2017-03-19]. Dostupné z: https://is.mendelu.cz/zp/portal_zp.pl?podrobnosti_zp=38916;zp=38916;download_prace=1. Diplomová práce. Mendelova univerzita v Brně, Provozně ekonomická fakulta. Vedoucí práce Jiří Šťastný.

Weichert, F., Bachmann, D., Rudak, B., Fisseler, D. *Analysis of the Accuracy and Robustness of the Leap Motion Controller*. Sensors [online]. 2013, 13(5), 6380-6393 [cit. 2017-03-08]. DOI: 10.3390/s130506380. ISSN 1424-8220. Dostupné z: http://www.mdpi.com/1424-8220/13/5/6380/.

Wikipedia. *Kinect*. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2017 [cit. 2017-02-28]. Dostupné z: https://en.wikipedia.org/wiki/Kinect.

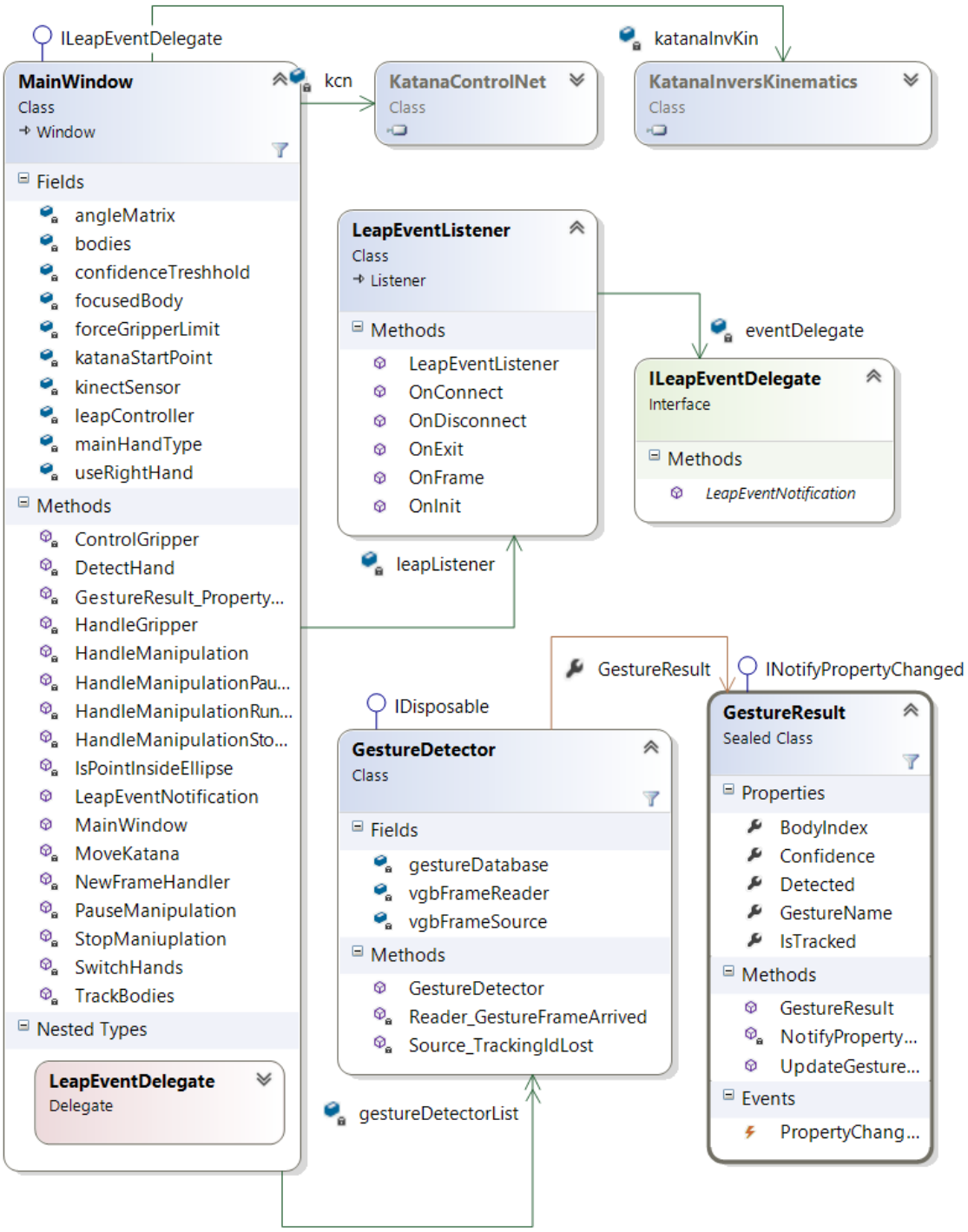
Wikipedia. *Leap Motion*. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2017 [cit. 2017-03-07]. Dostupné z: https://en.wikipedia.org/wiki/Leap_Motion.

Přílohy

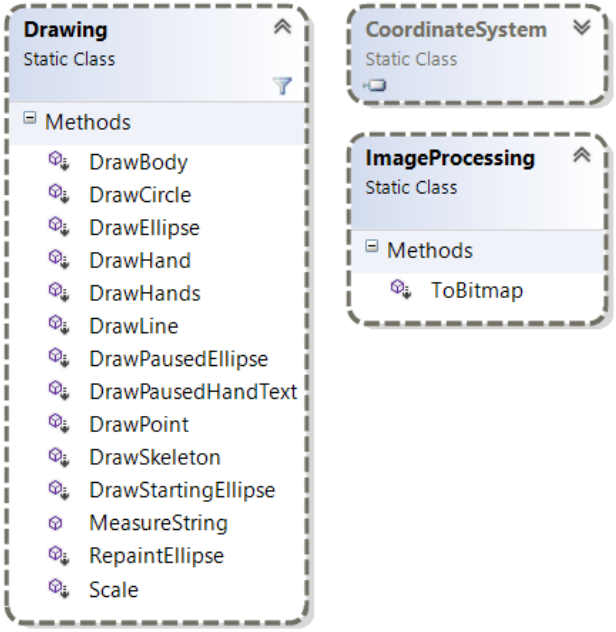
A CD s aplikací

Na přiloženém CD se nachází elektronická verze této práce ve formátu PDF a spustitelná aplikace umožňující ovládání manipulátoru Katana 6M180.

B Diagram tříd



Obrázek B.1: Diagram tříd



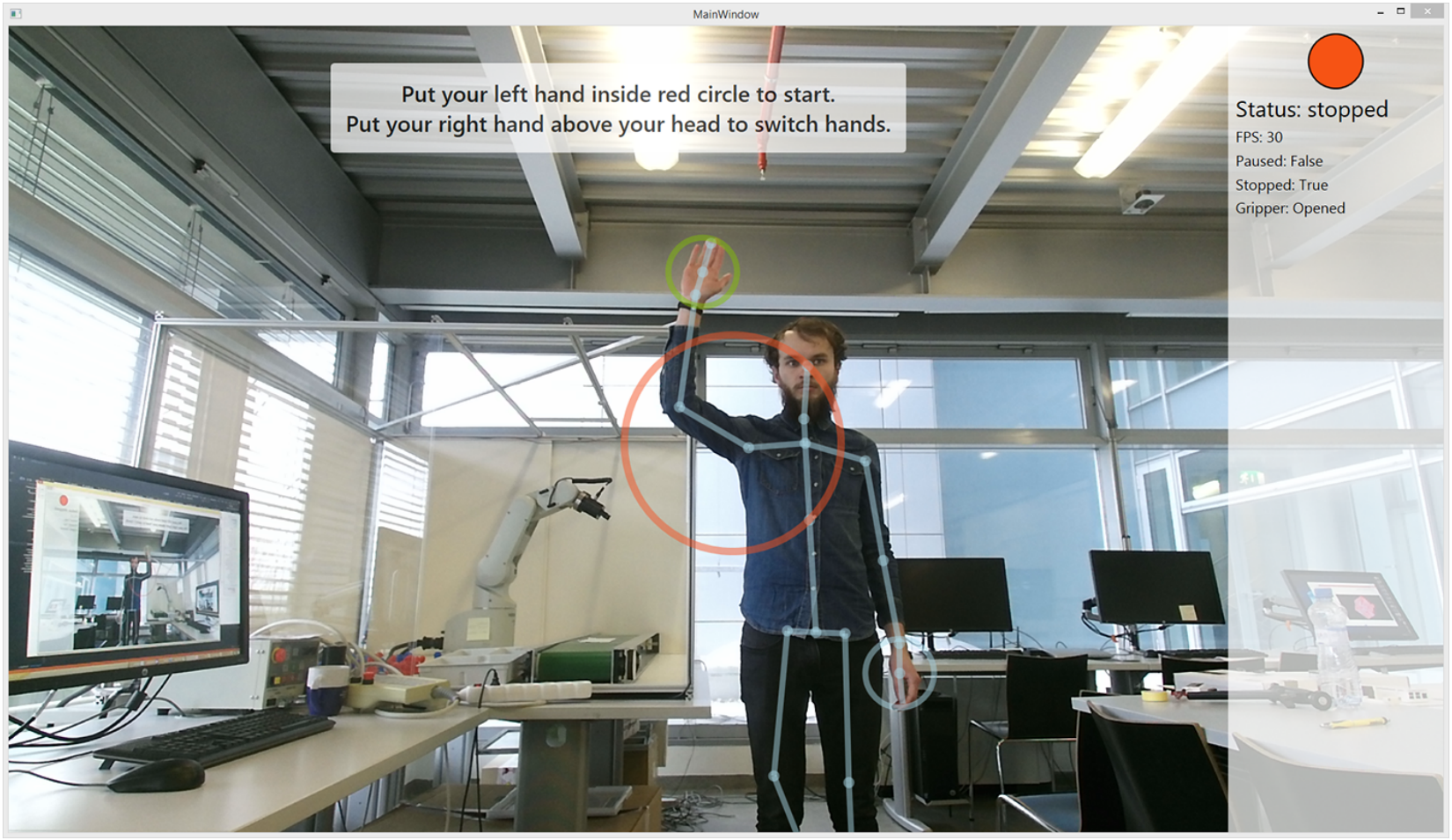
Obrázek B.2: Diagram statických tříd

C Screenshoty GUI

V této příloze jsou uvedeny screenshoty GUI aplikace v plné velikosti.

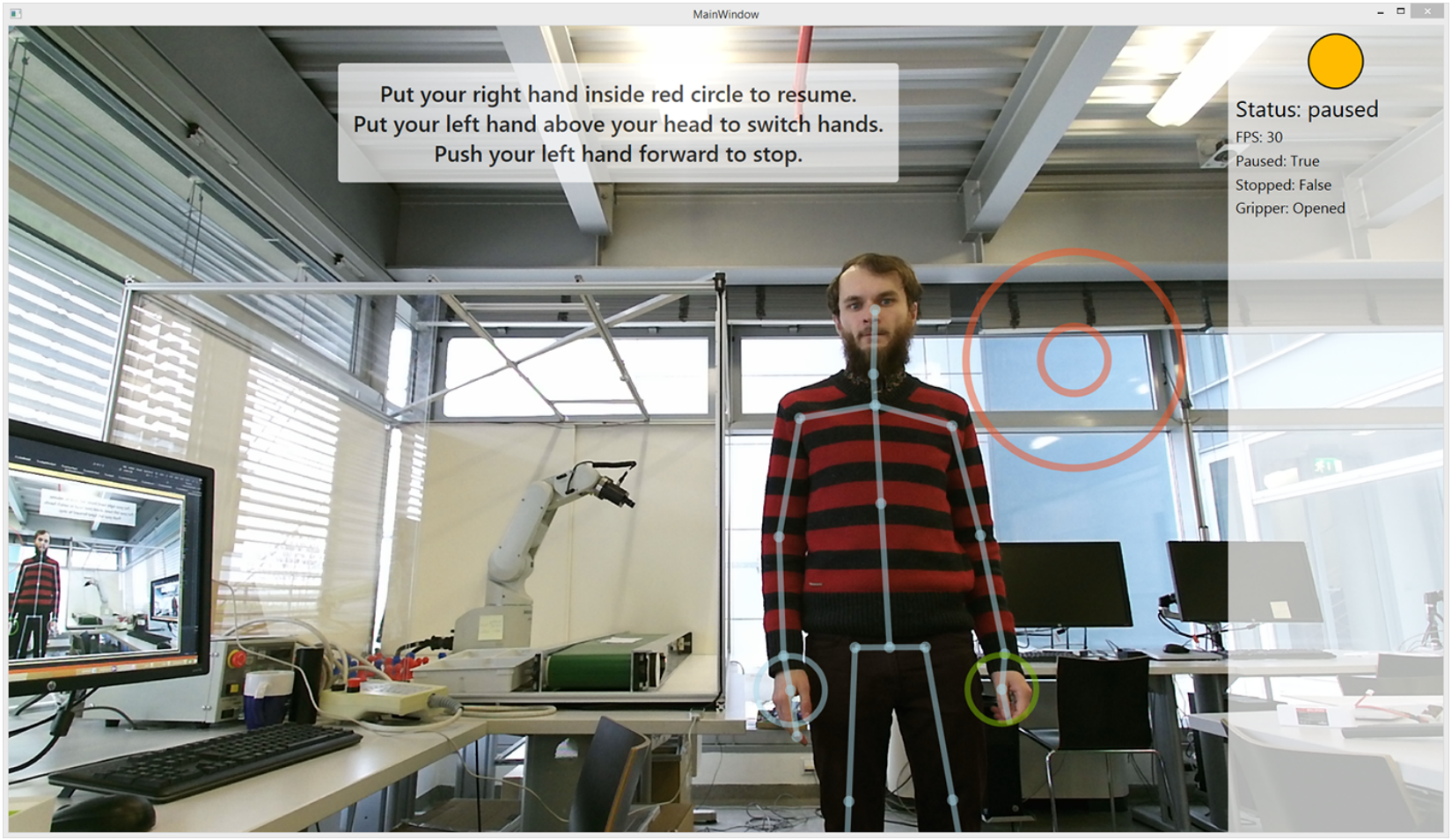
Obrázek C.1: Kalibrace manipulátoru

Obrázek C.2: Zastavená manipulace



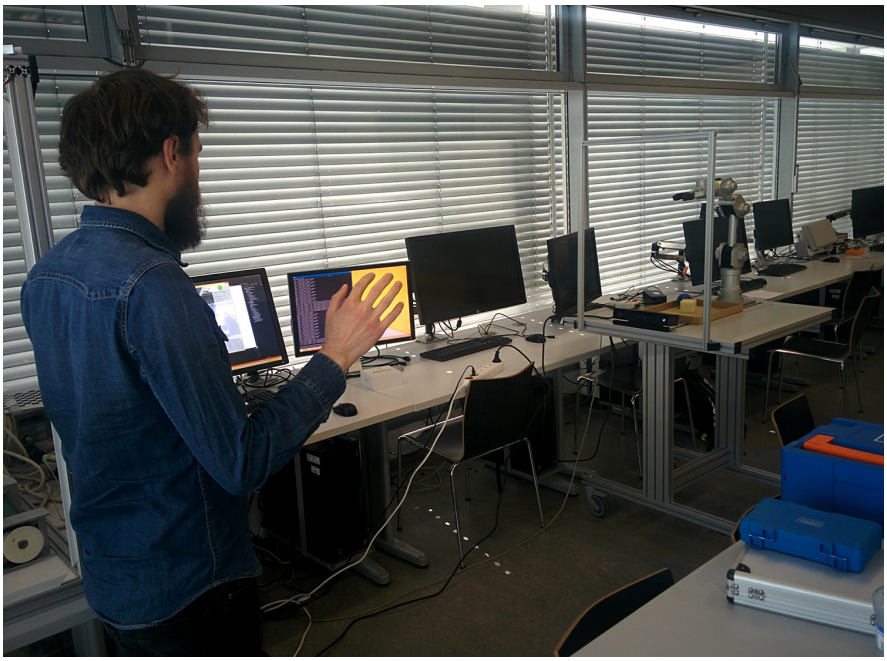
Obrázek C.3: Změna ruky pro manipulaci

Obrázek C.4: Probíhající manipulace manipulace

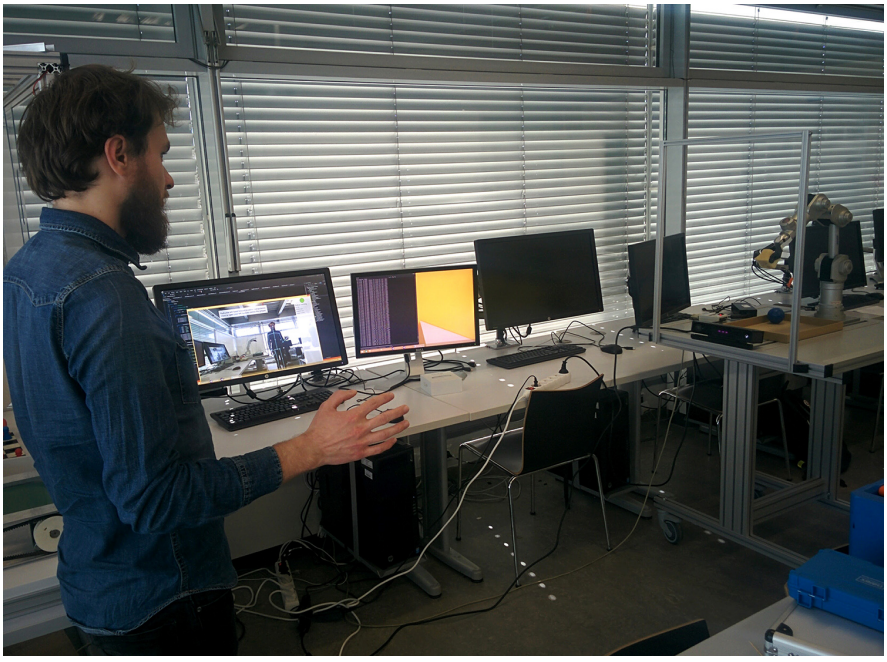


Obrázek C.5: Pozastavená manipulace

D Fotografie vytvořeného rozhraní



(a) Ovládání polohy manipulátoru



(b) Uchopení předmětu

Obrázek D.1: Fotografie vytvořeného uživatelského rozhraní