

Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra informatiky a výpočetní techniky

Bakalářská práce

Sběr 3D dat pro rehabilitační software ve virtuální realitě

Místo této strany bude
zadání práce.

Prohlášení

Prohlašuji, že jsem bakalářskou práci vypracoval samostatně a výhradně s použitím citovaných pramenů.

V Plzni dne 6. května 2020

Jakub Frank

Abstract

This bachelor thesis is focused on data acquisition and data analysis for rehabilitation software used by patients with multiple sclerosis. Intent of the software is to increase the availability of therapy for patients with this disease. This thesis aims to be able to record movements with available accelerometers and trackers and afterward analyze movements in real-time. Analysis should be based on appropriate evaluation parameters. Part of this thesis is the integration of this analysis into the final rehabilitation application and verification if the evaluation of movement can be done based only on the orientation of sensors without the need for tracking stations.

Abstrakt

Tato bakalářská práce se zabývá sběrem a analýzou 3D dat pro rehabilitační software určený pro rehabilitaci pacientů s roztroušenou sklerózou, který má za cíl zvýšit dostupnost terapie pro pacienty s touto chorobou. Cílem práce je umožnit záznam vzorových cviků pomocí běžně dostupných akcelerometrů a trackerů a následná analýza pohybu pomocí vhodných hodnotících parametrů. Součástí práce je také integrace tohoto řešení do výsledné aplikace a ověření, zda je možné provádět analýzu pohybu pouze na základě znalosti orientace použitých senzorů bez nutnosti použití sledovacích stanic.

Obsah

1	Úvod	7
2	Teoretická část	9
2.1	Roztroušená skleróza	9
2.2	Rehabilitace	9
2.2.1	Fyzioterapie a její využití pro léčbu RS	10
2.2.2	Metoda PNF	11
2.3	Virtuální realita	12
2.3.1	Historie a vývoj	12
2.4	Konkrétní VR zařízení	12
2.4.1	HTC Vive	13
2.4.2	Oculus Go	14
2.4.3	Oculus Quest	15
2.5	Druhy sledování pohybu	16
2.5.1	IMU	16
2.5.2	SteamVR Tracking	18
3	Záznam dat	20
3.1	Záznam dat pomocí IMU	20
3.2	Záznam dat z HTC VIVE	21
3.3	Porovnání	21
3.4	Výsledky měření	24
3.5	Formát naměřených dat	24
4	Segmentace a úprava dat	26
4.1	Přehrání naměřených pohybů	26
4.2	Segmentace dat	26
4.3	Začlenění do aplikace	27
5	Vyhodnocování pohybů	28
5.1	AAPD	28
5.1.1	Hodnocení správnosti	28
5.1.2	Nastavení parametrů	28
5.1.3	Vyhodnocování pohybu v AAPD	29
5.2	Požadavky na změny	30
5.3	Realizace změn	31

5.3.1	Rozbor požadavků	31
5.3.2	Implementace	31
6	Integrace do rehabilitačního softwaru	38
6.1	Požadavky na aplikaci	38
6.1.1	Cvičení	38
6.1.2	Hry	38
6.1.3	Možnost konfigurace	38
6.1.4	Vizualizace	38
6.1.5	Vyhodnocování a statistiky	39
6.1.6	Záznam a přehrávání	39
6.2	Realizace	39
6.2.1	Cvičení	39
6.2.2	Hry	40
6.2.3	Konfigurace	42
6.2.4	Statistiky	43
6.2.5	Vizualizace	43
6.2.6	Záznam a přehrávání pohybů	44
7	Experimentální část	45
7.1	Scénář experimentu	45
7.2	Konfigurace AAPD	45
7.3	Vyhodnocení experimentu	50
8	Závěr	51
	Literatura	54

1 Úvod

Pro pacienty s roztroušenou sklerózou a jinými neurologickými chorobami je rehabilitace jedinou a zároveň velmi zdoluhavou cestou, jak se vrátit do běžného života. Standardně pacient dochází na terapii několikrát týdně na specializované pracoviště, kde absolvuje sérii cvičení. V současné době je nutná přítomnost fyzioterapeuta po většinu času provádění terapie, a tím se zásadně snižuje její dostupnost.

Cílem rehabilitačního softwaru vyvíjeného na Katedře informatiky a výpočetní techniky na Fakultě aplikovaných věd ZČU ve spolupráci s Fakultní nemocnicí Královské Vinohrady je umožnit tuto terapii přenést i do domácího prostředí bez nutnosti účasti specializovaného pracovníka po celou dobu terapie. Aplikace by měla umožnit pacientovi rehabilitovat samostatně a zároveň poskytnout vyhodnocení provedeného cvičení pro analýzu přínosu rehabilitace.

Vývoj rehabilitační aplikace probíhá v rámci týmu studentů. Přínosem této práce pro výslednou aplikaci bude realizace části ohledně sběru 3D dat a následného vyhodnocování pohybů. V první části dojde k analýze současně dostupných zařízení pro virtuální realitu a možností jejich využití pro účely rehabilitační aplikace. Důraz je kladen zejména na autonomní VR zařízení z důvodu jejich vysoké přenositelnosti a absence sledovacích zařízení.

Součástí této práce dále bude tvorba modulu pro záznam a vyhodnocování pohybů a jeho následná integrace do rehabilitačního softwaru. Bude nutné definovat vhodný formát pro ukládání dat o pohybech. Vyhodnocování pohybů bude probíhat na základě přesně stanovených kritérií, které umožní fyzioterapeutovi analyzovat správnost vykonaného cvičení. Toto řešení bude řádně otestováno, aby se ověřila možnost jeho reálného využití v klinické praxi.

Závěrem této práce by měla být teze, zda bude možné použít některý z autonomních VR systémů, případně zda je nutné další vývoj omezit pouze na systémy s úplným sledováním pohybu.

První, teoretická část práce bude zaměřena na roztroušenou sklerózu, vývoj virtuální reality, současně dostupná VR zařízení. Součástí teoretické části bude i rozbor možností získávání dat z pohybových senzorů. Na základě tohoto rozboru budou porovnány možnosti sběru dat a určen vhodný kandidát pro výslednou aplikaci.

V další části práce bude popsán modul pro analýzu pohybů a jeho integrace do rehabilitační aplikace. V závěru práce bude proveden a vyhodnocen

experiment, zda je možné analyzovat pohyb bez nutnosti využití sledovacích stanic.

2 Teoretická část

2.1 Roztroušená skleróza

Celým názvem též *roztroušená mozkomíšní skleróza* byla jako nemoc poprvé popsána roku 1860. Jedná se o nevléčitelnou, autoimunitní nemoc doposud neznámého původu, která ovšem není sama o sobě smrtelná. Při onemocnění vznikají v bílé hmotě mozku a míše zánětlivá ložiska, tzv. léze, o velikosti několika milimetrů až centimetrů. V těch dochází k odumření myelinové vrstvy nervů, a tím ke zhoršení, až přerušení vedení nervového vzruchu. Pomocí remyelinizace¹ a nahuštění sodíkových a draslíkových kanálků na postižené místo dochází k částečné obnově schopnosti přenášet nervový vzruch. Lokální zánět, který doprovází chorobu, ovšem tuto obnovu značně oddaluje a obnovené vedení je podstatně pomalejší než to původní.

V průběhu nemoci se střídají fáze relapse a remise. V průběhu relapse (ataky) dochází k zhoršení stavu pacienta a jeho neurologických obtíží. V průběhu remise dochází naopak k částečné obnově myelinu a nemoc se nijak neprojevuje.

Mezi nejčastější prvotní příznaky onemocnění patří mravenčení, snížení citlivosti končetin a poruchy koordinace pohybů. V průběhu nemoci může dojít k ochrnutí některé části těla, častým urologickým obtížím nebo ke ztrátě zraku. Příznaky v pokročilém stádiu jsou velmi podobné mozkové mrtvici. Základem léčby jsou kortikoidy a léky potlačující záněty ve fázi relapse v kombinaci s fyzioterapií a rehabilitací.

Text v této kapitole vychází z [12].

2.2 Rehabilitace

Rehabilitace je dlouhodobý proces, který využívá multidisciplinárních strategií s cílem dosažení co největší nezávislosti pacienta s ohledem na omezení způsobená jeho onemocněním. V první fázi je důležité určit přesnou diagnózu pacienta a na jejím základě určit přesný postup a priority léčby. V neurorehabilitaci se využívá několik testů, které ověřují správnost léčby. Jejich výčet je podrobně uveden na straně 4 v [16].

¹obnova myelinové vrstvy nervu

2.2.1 Fyzioterapie a její využití pro léčbu RS

Při stanovení prognózy léčby se vychází z celé řady faktorů a na jejich základě se poté zvolí co nejvhodnější průběh léčby včetně jejího časového horizontu. V moderní fyzioterapii již není tak důležité dodržování přesných metodologických pravidel jako znalost a porozumění teoretickým východiskům a jejich následná aplikace terapeutem.

V průběhu terapie u neurologicky nemocných se kombinují dva přístupy.

Blokové schéma

V této metodě se využívá opakování stále stejné motorické dovednosti. Na neurální úrovni dochází k přenosu v rámci jedné dráhy a díky tomu dochází k rychlejšímu osvojení motorické dovednosti pacientem. Naučená stopa má ale větší tendenci k vymizení než u jiných způsobů asociativního učení.

Náhodné schéma

Informace je pravděpodobně přenášena pomocí několika neurálních drah, které jsou aktivovány při různých motorických dovednostech. Tento přístup vede k lepšímu zapamatování motorické dovednosti a je méně náchylný ke změnám spojeným s nečinností.

Při provádění terapie záleží hodně i na vnějších podnětech. Je důležité, aby cvičení probíhalo v prostředí, které odpovídá svými podmínkami běžnému životu. Návčik pohybu se dá přirovnat k vykonání programu. Při učení motorické dovednosti se na neurální úrovni vytvoří *engram*, neboli paměťová stopa. Pro vykonání této činnosti je následně tato stopa aktivována v závislosti na vnějším podnětu a tím dojde k vykonání série naučených událostí.

Pomocníkem při terapii je i mentální procvičování, které se dá využít při činnostech, v kterých je pacient již zběhlý a zároveň vyžadují i určitou přesnost při svém vykonávání. Na základě odborných vyšetření bylo dokázáno, že při mentálním procvičování dochází k aktivaci stejných částí CNS² jako při skutečném vykonávání této činnosti.

Pro návčik pohybů je velmi důležité používat aktivní pohyb (pacient cvičí sám), případně fyzioterapeutem aktivovaný pohyb (terapeut uvede správný pohyb). Z důvodu odlišnosti sensorických podnětů při vykonávání pasivního pohybu nevede tato metoda k zapamatování dovednosti. V průběhu návčiku by též pacient měl být korigován pouze při vybočení ze zadaných hranic pohybu, nikoliv při každé odchylce. Při poskytnutí přílišné zpětné vazby

²Centrální nervová soustava

pacient přestává spoléhat na své vlastní opravné strategie a začíná více spoléhat na vnější zdroj. Fyzioterapie se musí přizpůsobit aktuálnímu zdravotnímu stavu pacienta, především jeho únavě. Při únavě dochází k ovlivnění jak samotného výkonu, tak i procesu učení. Při nástupu únavy nebo bolesti u pacientů s RS je nutné terapii přerušit.

Terapie samotná je pak záležitostí dlouhodobou. Zpočátku se začíná intenzivním programem cvičení s terapeutem, který postupně přechází v individuální program. Pacient se snaží porozumět možnostem řízení vlastního těla a jeho následné kontrole při každodenních činnostech. Jedná se zejména o tyto činnosti:

- Sed
- Zvedání ze sedu do stoje a sedání
- Stoj
- Ovlivnění zalomeného kolena
- Nákrok a chůze

Text této kapitoly vychází z [15] a [16].

2.2.2 Metoda PNF

Metoda PNF³, známá též pod názvem *Kabatova metoda*, je velmi efektivní metoda pro fyzioterapii aplikovaná u širokého spektra pohybových diagnóz. Základem metody je pohyb končetin v diagonálách, které jsou rozlišeny na *flekční* (ohnutí) a *extenční* (narovnávací) části. Při opakovaném cvičení dochází k zvýšení rozsahu pohybu a zároveň také k posílení aktivovaných svalů. Nedílnou součástí PNF je učení a zapamatování nových pohybových vzorců, kterého lze dosáhnout díky opakování naučených pohybů.

Pohybové vzorce použité v PNF vycházejí z funkčních cílů, jako je například podání předmětu z poličky nebo pití z lahve. Kabatova metoda se používá u pacientů s funkčním onemocněním pohybového systému, pacientů s neurologickým onemocněním a v ortopedii při léčbě poúrazových a pooperačních stavů [5].

³Proprioceptivní nervosvalová facilitace

2.3 Virtuální realita

2.3.1 Historie a vývoj

První výskyt předchůdců virtuální reality se datuje již ke konci 18. století. Nejstarší zařízení je stereoskop, což je zařízení umožňující pomocí zdvojených zrcadel nahlížet na 2D obrázky jako na prostorové. V roce 1958 si nechal Morton Heilig patentovat první headset pro virtuální realitu. O 4 roky později, v roce 1962, stejný autor vynalezl přístroj Sensorama, viz obrázek 2.1. Zařízení bylo statické a mělo simulovat průjezd motorkou po městě, včetně zvuku a vibrací motoru.

Cílem virtuální reality je co nejdůvěryhodněji poskytnout iluzi pobytu ve virtuálním prostředí, neboli modelu. Tento model může představovat nějaké reálné prostředí nasnímané pomocí fotoaparátu, případně se může jednat o umělé prostředí vytvořené pomocí počítače. Pro zobrazení modelu je potřeba mít k dispozici hardware s dostatečným výkonem pro renderování výsledného obrazu. Uživatel využívá náhlavní soupravy s integrovanými displeji na kterých se zobrazuje vygenerovaný obraz. Tato náhlavní souprava, neboli headset, zpravidla obsahuje nějakou technologii pro rozpoznání pozice v prostoru a následné vygenerování odpovídající scenerie. V řadě dnešních VR zařízení je možnost připojit periferie pro ovládání pohybu a uživatelského prostředí ve virtuálním světě. [3]

Virtuální realita má široké využití zejména v herním průmyslu, strojním průmyslu, designu, kosmonautice a medicíně.

2.4 Konkrétní VR zařízení

V současné době je kladen tlak na cenu, kvalitu zobrazení a přenositelnost VR zařízení. Toto vede k velké nabídce na trhu s VR zařízeními a příslušenstvím. Některá zařízení potřebují ke svému běhu připojený počítač, který provádí všechny příslušné výpočty a headset se stará pouze o zobrazení. Oproti tomu takzvaná *standalone* zařízení provádí renderování i zobrazení přímo v headsetu. Nesmírnou výhodou tohoto provedení je přenositelnost a nezávislost na dalším hardwaru.

S vývojem chytrých telefonů jsou na trhu k dispozici i adaptéry na vytvoření headsetu přímo z chytrého telefonu. Nejznámějším zástupcem těchto zařízení je *Samsung Gear VR*. Toto řešení ovšem trpí nedostatky v kvalitě a v důvěryhodnosti simulovaného prostředí, a tím zaostává za speciálními VR zařízeními.

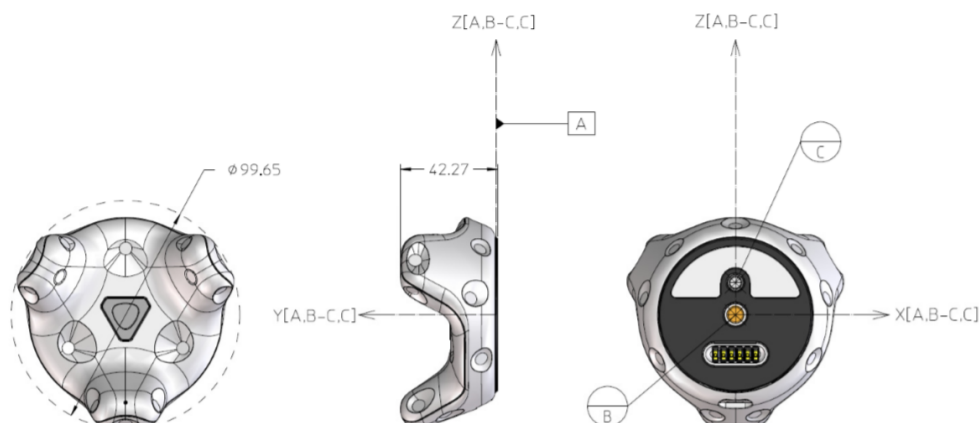


Obrázek 2.1: Simulátor virtuálního prostředí Sensorama [1]

Pro přehled je zde uveden výčet několika typů headsetů, z nichž každý se liší způsobem sledování pohybu v prostoru.

2.4.1 HTC Vive

Platforma *HTC Vive* byla představena v roce 2016 díky spolupráci společností *Valve* a *HTC*. Jedná se o velmi rozšířený systém pro virtuální realitu s možností řady rozšíření. Základem je headset s dvěma displeji o rozlišení 1080×1200 px a frekvencí obnovování 90 Hz. Headset se stará pouze o vykreslování obrazu, takže pro jeho použití je potřeba použít počítač, který obstará vše ostatní, včetně napájení headsetu. V základní verzi je headset připojen k počítači pomocí kabelu, ale systém lze rozšířit i o bezdrátový



Obrázek 2.2: Schéma Vive trackeru [13]

přenos. V tomto případě je ovšem potřeba použít externí baterii pro napájení. Aplikace pro tuto platformu jsou dostupné z integrovaného obchodu *Viveport*, případně z dalších komerčních obchodů, jako je například *Steam*.

HTC Vive disponuje širokým spektrem periferií a doplňků. Pro sledování pohybu využívá technologii *SteamVR Tracking*, kdy je potřeba do místnosti umístit alespoň 2 sledovací majáky, neboli *lighthouse*, které sledují pohyb připojených zařízení. Více podrobností o sledování pohybu pomocí *SteamVR Tracking* je k nalezení v části 2.5.2.

Controllery a Trackery

Controllery, neboli ovladače, slouží k ovládání prostředí virtuální reality. Jsou navrženy pro uchopení do rukou a doplněny o ovládací tlačítka. Přenos dat mezi řídicím počítačem a controllerem zajišťuje bezdrátová síť a o jejich napájení se stará integrovaná baterie.

Trackery slouží k rozšíření prostředí virtuální reality. Díky možnosti uchycení pomocí stativového šroubu umožňují snadnou montáž například na makety zbraní nebo makety sportovního vybavení. Pomocí *pogo* pinů je možné přiřadit i ovládací prvky, které mohou ovládat například výstřel ze zbraně.

2.4.2 Oculus Go

Headset pro virtuální realitu *Oculus Go* od výrobce *Oculus*, spadajícího pod *Facebook*, byl představen v roce 2018. Jedná se o autonomní VR zařízení, které nepotřebuje pro svojí funkčnost žádný další hardware, jako například

počítač. Další velkou výhodou oproti konkurentům je nízká pořizovací cena, která se v dubnu 2020 pohybovala v rozmezí 7000 Kč - 9000 Kč. Headset ale nedisponuje sledováním polohy v prostoru, nýbrž pouze sledováním orientace pomocí IMU senzorů (viz 2.5.1). Snížení stupňů volnosti pohybu je hlavní nevýhodou *Oculus Go* oproti jeho dražším konkurentům a předurčuje headset zejména ke sledování 360° videa nebo pro hraní jednodušších her.

Zařízení je opatřeno jedním LCD displejem o úhlopříčce 5" a rozlišením 2560 px x 1440 px. Zvládne pracovat s frekvencí 60 nebo 72 snímků za vteřinu. Pro zobrazení se využívají Fresnelovy čočky, které oproti klasickým čočkám vynikají nižší hmotností. Výrobce udává, že v době vydání se jednalo o nejlepší čočky, které zatím byly v jeho headsetu použity. O výpočetní sílu se stará stejný čtyřjádrový čip jaký byl použit například u *Samsungu S7*. Headset je dostupný ve variantě s 32 GB nebo 64 GB integrovanou pamětí. K ovládání slouží ruční malý controller, také vybavený senzory IMU, které umožňují sledování jeho rotace.

Jako operační systém zařízení používá *Android*. Vývoj aplikací probíhá pomocí vlastního SDK⁴ odvozeného od *Android SDK*, které lze integrovat do herních engine jako *Unreal* nebo *Unity*. Headset je vybavený připojením *Wi-Fi* a *Bluetooth*, takže ho lze snadno spárovat například s mobilním telefonem. Aplikace na tuto platformu lze stahovat z *Oculus Store*, kde jsou k dispozici placené i neplacené aplikace.

Text této sekce je založený na informacích převzatých z [10].

2.4.3 Oculus Quest

Oculus Quest je dalším autonomním VR zařízením od společnosti *Oculus*. Jako první headset od této společnosti umožňuje sledovat pohyb v šesti stupních volnosti bez potřeby jakéhokoliv dalšího zařízení. *Oculus Quest* lze tedy použít kdekoliv bez nutnosti složité instalace sledovacích stanic a bez nutnosti výkonného počítače jako u ostatních konkurentů.

Tracking systém *Oculus Insight*, stejně jako většina ostatních tracking systémů, využívá jako primární zdroj dat senzory IMU (podrobněji viz 2.5.1) s obnovovací frekvencí 1000 Hz. Inside - out tracking systém dále využívá 4 kamery umístěné na headsetu k vytvoření 3D mapy okolního prostoru. Pro automatické rozpoznávání obrazu využívá výrazné objekty v prostoru, jako ostré rohy a hrany. Uživatel si označí bezpečnou zónu, kde se může pohybovat. Pokud se přiblíží k jejím hranicím, je upozorněn jejich vizualizací ve virtuálním prostředí, a pokud tuto bezpečnou zónu opustí úplně, dojde k ukončení projekce.

⁴vývojové prostředí, software development kit

K ovládání slouží dva controllery s oblouky ve směru nahoru k headsetu. Tyto oblouky jsou osazeny několika IR diodami sloužícími pro tracking *Oculus Insight*. Nevýhodou tohoto systému oproti sledování pomocí základových stanic je nutnost vizuálního kontaktu mezi headsetem a controllerem, což znemožňuje využití sledování šesti stupňů volnosti senzory umístěnými na různých částech těla.

Headset je vybavený OLED displayem o rozlišení $1440\text{ px} \times 1600\text{ px}$ a obnovovací frekvencí 72 Hz. Na trhu jsou dostupné dvě verze s 64 GB nebo 128 GB integrovanou pamětí. Cena setu je srovnatelná jako u zařízení *HTC Vive*. Operační systém je stejně jako u *Oculus Go* založený na *Android* a SDK lze volně stáhnout ze stránek výrobce. Pro stahování aplikací slouží také *Oculus Store*. Zařízení umožňuje připojení na *Wi-Fi* a případné spárování s mobilním telefonem pomocí *Bluetooth*.

Informace v této sekci jsou převzaté z [4].

2.5 Druhy sledování pohybu

2.5.1 IMU

Senzory *IMU* (inercial measurement unit) slouží k získání dat o pohybu sledovaného zařízení. Jedná se zpravidla o mikročip obsahující akcelerometr, gyroskop a senzor magnetického pole. Jejich hlavní výhoda spočívá v nízké ceně, malé energetické náročnosti a hlavně vysoké frekvenci obnovování. Svými malými rozměry a hmotností umožňují snadnou integraci do chytrých telefonů, létajících dronů, případně zařízení z kategorie nositelné elektroniky, takzvaných *wearables*. Senzory inerciální měřící jednotky poskytují následující sady třírozměrných dat:

Akcelerometr slouží k zjištění zrychlení posuvného pohybu tělesa včetně gravitačního zrychlení. Pokud je objekt v klidu, dá se pomocí odečtení gravitačního zrychlení určit jeho orientace vůči zemskému povrchu.

Gyroskop sleduje hodnoty úhlové rychlosti zařízení nezávisle na gravitaci, čehož se dá využít pro sledování změny orientace v prostoru.

Senzor magnetického pole udává intenzitu magnetického pole Země ve třech osách, která ovšem může být ovlivněna lokálním magnetickým polem. Magnetické pole Země se mění v závislosti na aktuální zeměpisné délce a šířce, tudíž pro výpočet orientace samotná data z tohoto senzoru nestačí.

Samotná sada dat pouze z jednoho z těchto senzorů nestačí pro určení orientace a polohy v prostoru, protože ani jeden z těchto senzorů není schopný samostatně udávat důvěryhodná data nezávisle na pohybu zařízení.

Výpočet orientace a polohy

Odhad orientace zařízení se provádí integrací dat z gyroskopu, tedy úhlové rychlosti zařízení v daném čase. Pro odhad pozice se používá dvojitá integrace dat z akcelerometru, tedy lineárního zrychlení. Od těchto dat je ale potřeba odečíst hodnotu gravitačního zrychlení působícího na zařízení. Vektor gravitačního zrychlení se určí pomocí aktuální orientace zařízení.

Při měření orientace a polohy pomocí IMU se časem naměřené hodnoty přestávají shodovat se skutečnou hodnotou orientace a polohy v prostoru. Této chybě se říká *drift*. Jednou z příčin je nepřesnost samostatných senzorů vlivem konstrukce senzoru, kdy i v případě, že se zařízení nachází v klidu, dochází k naměření malých odchylek. Dalším zdrojem nepřesností je numerická integrace.

Pro odhad počáteční orientace se předpokládá, že zařízení setrvává v klidu. Akcelerometr tedy udává pouze hodnotu (vektor) gravitačního zrychlení a magnetometr udává vektor směřující k magnetickému severu. Ten je ještě ovlivněný inklinací⁵ v aktuální zeměpisné poloze. Díky hodnotám těchto vektorů, které se oba nacházejí v jiné souřadné soustavě, se dá odhadnout aktuální orientace zařízení. Jedná se pouze o odhad, jelikož v okolí zařízení může působit lokální magnetické pole, které ovlivní funkci magnetometru. Tato část vychází z [8], kde je k nalezení i velmi podrobný popis těchto odhadů.

Fúze senzorů

Pro odhad orientace v prostoru se využívá principu *Kalmanových filtrů*, které na základě nepřesných vstupních dat poskytují odhad výstupní hodnoty. Tento princip je uplatňován při kombinaci dat z několika různých senzorů, například GPS a IMU. Algoritmus pracuje s důvěryhodností jednotlivých sad na základě znalosti jejich kovariance. Fúze může probíhat hardwarově přímo v IMU, programově nebo pomocí některé z dostupných knihoven. Podrobný popis principu *Kalmanových filtrů* je dostupný v [6].

⁵odchylka siločar magnetického pole Země od jejího povrchu, na obou magnetických pólech se rovná 90°

2.5.2 SteamVR Tracking

Základem tohoto systému, vyvinutým společností *Valve*, je sledování pomocí IMU, které je výpočetně nenáročné a dokáže dodávat data s frekvencí 500 Hz . Pro odstranění *drift* (viz 2.5.1) se využívá systém základových stanic, takzvaných *lighthouse*.

Základová stanice je vybavena dvěma rotujícími kotouči, jedním v horizontální ose a druhým ve vertikální ose, které se oba otáčejí s frekvencí 60 Hz a v jednom místě jsou opatřeny diodou vysílající infračervené paprsky. Rozdíl fází diod na jednotlivých kotoučích je přesně 180° . Další 2 sady statických IR diod jsou určeny pro synchronizační (reset) pulsy. Reset pulsy jsou vyslány vždy, když rotující diody vyzařují v úhlu 90° od dopředného vektoru základové stanice. Následně je celý prostor v úhlu 180° postupně osvětlen jednou z rotujících diod, poté dojde k vyslání dalšího synchronizačního pulsu a následně dojde opět k vyslání signálu druhou rotující diodou, tentokrát v druhé ose (horizontální nebo vertikální).

Sledované zařízení (například headset nebo controller) má na svém povrchu umístěno několik fotodiód, které snímají IR záření vysílané základovými stanicemi. Po zaznamenání synchronizačního pulsu se spustí časovač, který měří čas než fotodióda zaznamená signál z rotujícího kotouče. Díky znalosti frekvence otáčení (60 Hz) a znalosti rozložení jednotlivých diod na sledovaném zařízení se dá dopočítat aktuální poloha zařízení v prostoru před základovou stanicí. Takto se ovšem získá poloha pouze v jedné ose, v druhé ose se zjistí se zpožděním přibližně $16,6\text{ ms}$ (odpovídá polovině doby otočení jednoho kotouče při frekvenci otáčení 60 Hz). Přesnou polohu tedy nelze získat v jednom okamžiku, nýbrž s časovým odstupem jednotlivých os. Proto se tento systém využívá pouze pro zpřesnění dat z IMU, nikoliv jako hlavní zdroj dat o orientaci a poloze.

Systém dokáže pracovat pouze s jednou základovou stanicí, nicméně z důvodu přesnosti a variability pohybů je doporučeno používat dvě stanice. Ideální umístění je do dvou protilehlých rohů sledovaného prostoru.

SteamVR Tracking 2.0

V roce 2018 společnost *Valve* představila druhou generaci tracking systému pro virtuální realitu. Došlo k použití jiných diod pro záznam IR záření. Nové diody jsou konstrukčně jednodušší a umožňují předávat i informaci zakódovanou v signálu. Toho se využívá k přenosu informace, v jakém úhlu byl paprsek světla vyslán ze základové stanice. Informace se poté dekóduje přímo v zařízení a odpadá tak nutnost používat synchronizační pulsy. Další změnou je použití pouze jednoho rotujícího kotouče, čímž došlo k výrazné

redukci ceny.

Druhá generace tracking systému je oproti svému předchůdci konstrukčně daleko jednodušší, což snížilo cenu základových stanic. Nevýhodou je ovšem nekompatibilita staršího hardware s novým lighthouse systémem, který není vybavený synchronizačními pulsy. Opačná kompatibilita (hardware první generace a lighthouse druhé generace) možná je, konkrétním příkladem je zde VIVE Tracker 2018, který dokáže pracovat s oběma verzemi tracking systémů.

Informace o *SteamVR Tracking* pocházejí ze zdroje [14].

3 Záznam dat

Výsledná aplikace bude v reálném čase zaznamenávat 3D data o pohybu pacienta. Odborníci z Fakultní nemocnice Královské Vinohrady v Praze poskytli informace, že u pacientů s roztroušenou sklerózou je při cvičení důležité sledovat a vyhodnocovat nejen pohyb končetiny, nýbrž i trupu a hlavy, aby došlo ke správnému naučení pohybových vzorců a případné korekci cvičení. Po konzultaci padlo rozhodnutí sledovat pohyb následujících částí těla:

- hlava
- hrudník
- paže (pod ramenem)
- předloktí

V rámci prvního experimentu bylo potřeba porovnat různé metody záznamu pohybu. Jako nejvhodnější kandidáti se jeví metoda záznamu pohybu pomocí IMU (viz 2.5.1), případně komplexnější metoda záznamu pomocí VIVE Trackerů (viz 2.5.2). Pro oba způsoby byly implementovány záznamové aplikace, které jsou podrobně popsány a porovnány dále.

3.1 Záznam dat pomocí IMU

Pro záznam dat pomocí IMU byla použita aplikace *Sensorstream IMU+GPS*, která je volně dostupná ke stažení z *Google Play* a lze ji nainstalovat do zařízení s operačním systémem *Android* verze 2.3.3 a vyšší. Autorem aplikace je Alex Lorenz. Aplikace umožňuje ukládat data ze senzorů dostupných na zařízení přímo do paměti ve formátu `.csv` se znakem `,` jako oddělovačem. Jedna řádka souboru odpovídá formátu `"t, s, x, y, z"`.

- `t` - čas
- `s` - číslo senzoru určené platformou *Android*, orientaci odpovídá číslo 81

Aplikace také umožňuje odesílat naměřená data po síti přes UDP stream. V grafickém rozhraní se dá zvolit IP adresa příjemce a port, na kterém bude naslouchat. Jeden datagram odpovídá jednomu řádku `.csv` souboru.

Pro účely testování byla vytvořena záznamová aplikace pro příjem a ukládání dat do počítače. GUI implementované v nástroji *Java FX* umožňuje jednoduché navolení parametrů a adres vysílajících zařízení. Z důvodu minimalizace nákladů na testování byly použity sériově vyráběné chytré telefony s operačním systémem *Android*. Ty se upevnily na paži, předloktí a hrudník rehabilitačního pracovníka, který předváděl předem určené cviky a následně simuloval chyby při jejich vykonávání, viz sekce 3.3. Senzor na hrudníku byl umístěný uprostřed v oblasti prsou, zbylé senzory vždy přibližně v polovině příslušné kosti tak, aby při upažení směřovaly nahoru. Orientace je ukládána v eulerovských úhlech. Zaznamenaná data se ukládají na disk počítače ve formátu popsaném výše.

3.2 Záznam dat z HTC VIVE

Záznamová aplikace pro HTC VIVE vznikla v herním enginu *Unity*. Z důvodu usnadnění následného zpracování byla snaha o zachování co nejpodobnějšího formátu dat jako z IMU senzorů, tedy o ukládání eulerovských úhlů rotací. Technologie navíc umožňuje uložení přesné pozice sledovaného prvku v prostoru. Každý záznam tedy obsahuje údaj o čase, poloze a orientaci senzoru. Dále je možné ukládat data i z headsetu, což u záznamu z IMU nebylo možné z důvodu nepoužití headsetu.

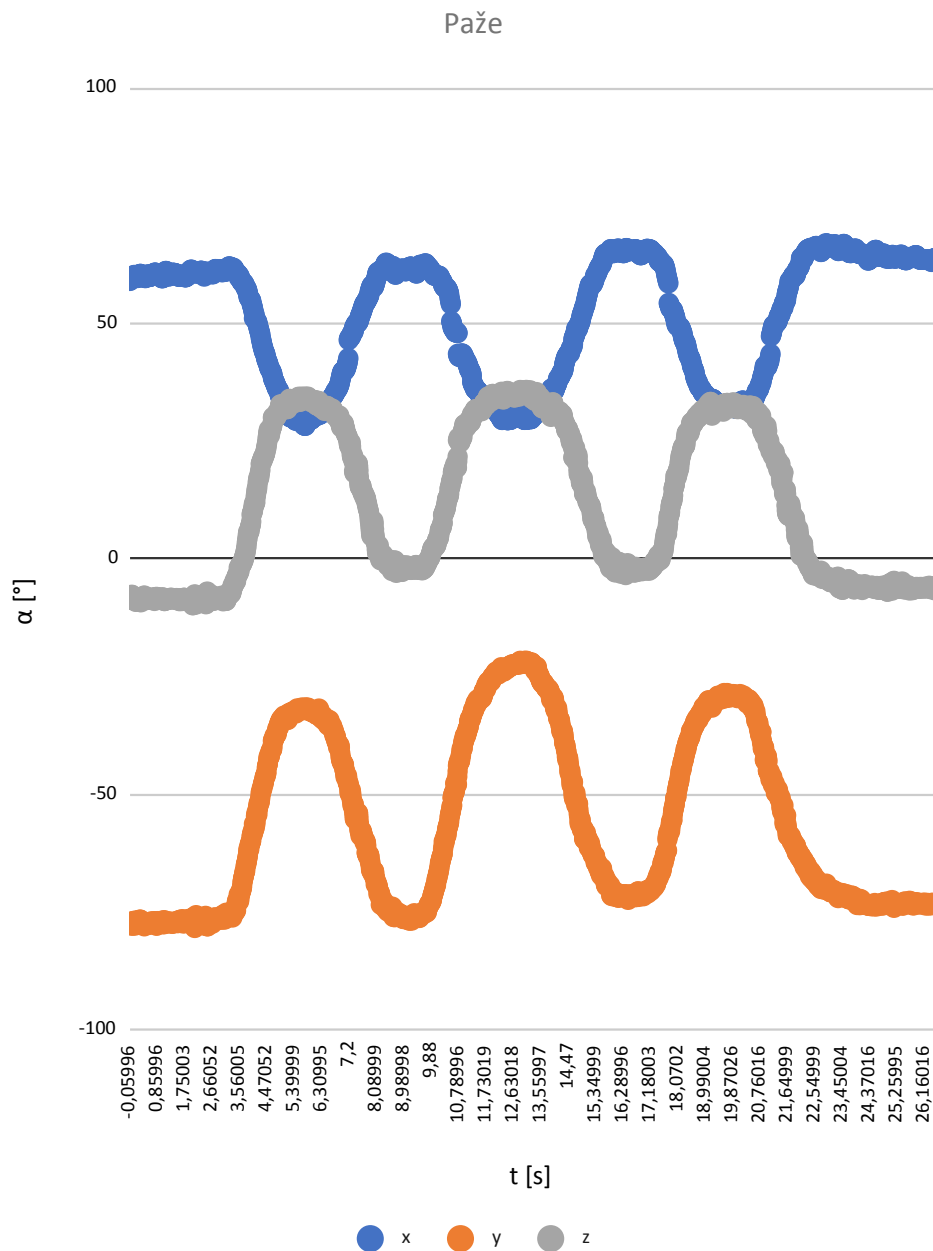
Ve fázi testování a pořizování prvotních záznamů bylo zjištěno, že při použití více sledovaných zařízení v systému *SteamVR Tracking* dojde při opětovném spuštění aplikace k záměně jednotlivých senzorů ve scéně a tudíž je před začátkem měření potřeba určit, který reálný senzor odpovídá sledovanému objektu a aplikaci. Tento problém se podařilo vyřešit pomocí pluginu *Vive Input Utility*, který dokáže svázat logickou roli trackeru s jeho sériovým číslem.

3.3 Porovnání

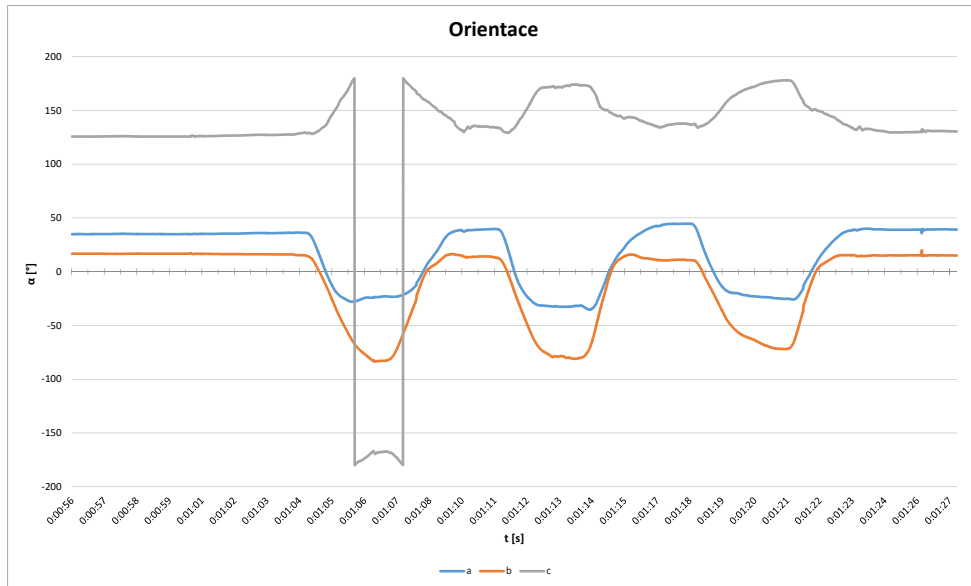
Pro porovnání výše zmíněných druhů záznamu dat o pohybu byl vybrán jeden referenční cvik. Tímto cvikem byla simulace pití pravou rukou. Jedná se o cvik, který je běžně zařazený do rehabilitace pacientů s roztroušenou sklerózou a o jeho správné provedení se postaral fyzioterapeut z Fakultní nemocnice Královské Vinohrady v Praze. V průběhu měření byl tento cvik opakován třikrát. Z důvodu rozpoznání jednotlivých senzorů grafy u *HTC Vive* začínají až v 56. sekundě.

Čtenář může porovnat data o orientaci z IMU (obrázek 3.1) a z *HTC*

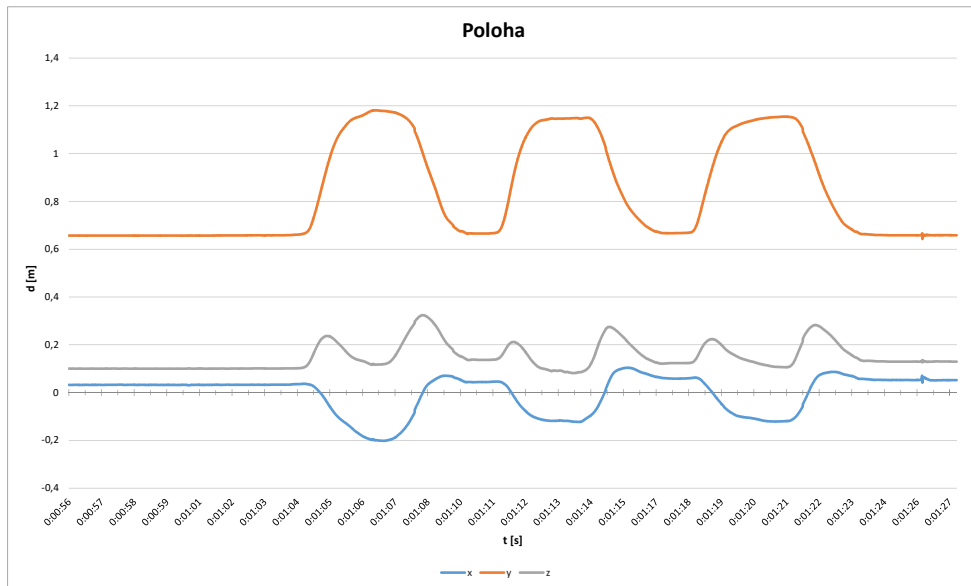
Vive (obrázek 3.2). Pro představu je přiložen i graf průběhu polohy z *HTC Vive* na obrázku 3.3.



Obrázek 3.1: Naměřená data orientace ze sensorů IMU na předloktí



Obrázek 3.2: Naměřená data orientace z HTC Vive z trackeru na předloktí



Obrázek 3.3: Naměřená data polohy z HTC Vive z trackeru na předloktí

3.4 Výsledky měření

Po porovnání výsledků obou způsobů měření bylo rozhodnuto o dalším vývoji rehabilitačního softwaru pro platformu *HTC Vive*. Sensory IMU sice poskytují na první pohled důvěryhodná data, ale kvůli výskytu *drift* chyby se jejich přesnost bude časem snižovat, a bylo by potřeba senzory kalibrovat před zahájením cvičení a pravděpodobně i v jeho průběhu. Dalším důvodem je momentální nedostupnost adekvátního hardwaru, který by svojí velikostí umožňoval snadné připevnění na tělo pacienta. Použití velkých chytrých telefonů kvůli jejich rozměrům by bylo značně komplikované pro následující praktické využití finální aplikace.

HTC Vive vyniká vyšší počáteční investicí na pořízení hardwaru a následnou složitější instalací u klienta kvůli nutnosti použití základových stanic a konfiguraci prostoru. Oproti inerciálním sensorům ale *SteamVR Tracking* poskytuje důvěryhodná data po celou dobu běhu aplikace. Navíc umožňuje sledovat i hodnoty absolutní polohy v prostoru.

Další vývoj rehabilitační aplikace se tedy bude prozatím soustředit na platformu *HTC Vive* s vidinou možného budoucího přechodu na IMU v případě dostupného hardwaru pro sledování pohybu v prostoru (například *Oculus Quest* a vhodné senzory jako nositelná elektronika). Dalším předpokladem pro změnu sensorů je ověření možnosti určit správnost vykonávaného pohybu pouze na základě znalostí orientace sensorů.

3.5 Formát naměřených dat

Bylo dohodnuto, že data o pohybu se budou držet jednotného formátu napříč celým rehabilitačním softwarem. Tímto se minimalizuje riziko chyb z důvodu špatného formátu, což by způsobovalo problémy zejména při odděleném vývoji jednotlivých komponent. Všechna desetinná čísla se ukládají s desetinou tečkou, nikoliv desetinou čárkou. Konvence ukládání dat je následující:

```
hh:mm:ss.mmmmm; pX;pY;pZ;oX;oY,oZ\n
```

Hodnoty začínající na *p* značí pozici (v metrech) v příslušné ose a hodnoty začínající na *o* značí orientaci (v eulerovských úhlech) okolo příslušné osy. Data se ukládají do 4 souborů s jmennou konvencí:

- `Arm.csv`
- `Forearm.csv`
- `Chest.csv`

- `Head.csv`

Dále je ještě doplněn soubor `Neutrals.csv`, kam se ukládají transformace sensorů v neutrální pozici, jelikož ty se neshodují s počáteční pozicí cviku.

4 Segmentace a úprava dat

Zaznamenaná data od fyzioterapeutů slouží jako vzor pro vyhodnocování cvičení reálných pacientů. Z tohoto důvodu je tedy potřeba si zaznamenaná data znovu přehrát a případně oříznout jednotlivé cviky, které následně využije jako předlohu modul pro určení polohy ruky při cvičení.

4.1 Přehrání naměřených pohybů

Pro přehrání pohybů vznikla samostatná aplikace v herním engine *Unity*. Po spuštění načte všechna data v zadané složce. Pro správné fungování aplikace musejí tato data být ve formátu definovaném v 3.5. Z načtených dat se vytvoří *animation clip*, což je funkcionalita *Unity* pro přehrávání animací z externího zdroje, který je svázaný s odpovídajícím objektem ve scéně. Načtená data se ještě vzorkují s četností 20 vzorků za sekundu (*timespan* = 50 ms). Orientace se převádí z eulerovských úhlů na kvaterniony, čímž je vyřešen přechod mezi hodnotami 360° a 0° . Jednotlivé datové sady jsou ještě interpolovány, aby animace byla plynulá. Pro výpočet interpolace křivky *Unity* využívá dva parametry - tangens vstupního a výstupního úhlu křivky v konkrétním bodě. Výpočet těchto úhlů probíhá podle vzorce 4.1 a stejná hodnota se používá pro vstupní i výstupní parametry.

$$tg_i = \frac{pos_{i+1} - pos_{i-1}}{2 * timespan / 1000} \quad (4.1)$$

Zvolené modely objektů odpovídají reálným zařízením *HTC Vive* (headset, controller, tracker). Prvotní prototyp aplikace vyvinul pan Ing. Petr Vaněček, Ph.D. a tento byl dále v rámci této práce upravován do finální podoby. Jednalo se o změny formátu .csv souborů a úpravu vizualizace.

4.2 Segmentace dat

Segmentace dat a oříznutí validních cviků probíhá manuálně. Nahrávání dat se řídí stisknutím tlačítka pro začátek a ukončení nahrávání. Segmentace musí probíhat tak, aby zaznamenaná data obsahovala jen jeden pohyb a ne nic před ním a po něm. Zároveň musí ale ten, kdo obsluhuje záznam dat, dávat pozor, aby nedošlo k pozdnímu spuštění záznamu, případně jeho předčasnému ukončení. V případě nutných úprav je možné data oříznout manuálně,

ideálně za využití aplikace pro přehrání naměřených pohybů.

4.3 Začlenění do aplikace

Dle požadavků od Fakultní nemocnice Královské Vinohrady byla možnost zaznamenat a následně přehrát jednotlivé cviky i v průběhu terapie začleněna do druhé verze výsledné aplikace. Podrobný popis této funkcionality se nachází v sekci 4.1.

5 Vyhodnocování pohybů

5.1 AAPD

Pro vyhodnocení pohybu ruky byla vytvořena samostatná knihovna v prostředí *.NET*, jejíž prvotní verze vznikla v rámci bakalářské práce Vítka Poóra [11]. Součástí této práce je následná úprava knihovny od Vítka Poóra na základě požadavků popsanych v 5.2. Cílem tohoto modulu je určit správnost vykonávaného pohybu na základě záznamu vzorového cviku vykonaného fyzioterapeutem. Využívá princip tunelu, který odpovídá válcovitému tělesu o poloměru určeném konfiguračním parametrem (viz 5.1.2) a jehož střed určují body zaznamenaného vzorového cviku. Základem knihovny je vyhodnocování 24 rozměrných dat (4 senzory, každý 6 DOF¹ - poloha a orientace) pomocí *24D* tunelu.

5.1.1 Hodnocení správnosti

Pro vyhodnocení se využívá dvou parametrů - fáze a kvalita.

Fáze slouží k určení, jakou část cviku pacient vykonal. Je udávána v procentech, kdy 0 % značí počátek pohybu a hodnota 100 % okamžik největší extenze cviku. V ideálním případě se tedy pacient pohybuje nejdříve od 0 % do 100 % a následně se vrací opět do 0 %. Z důvodu odlišné rychlosti vykonávání pohybu v jeho jednotlivých částech (v koncových bodech pohyb zpravidla probíhá pomaleji) se hodnota fáze určuje na základě vzdálenosti mezi jednotlivými body, nikoliv na základě aktuálního indexu v poli zaznamenaných bodů.

Kvalita je určena vzdáleností k nejbližšímu bodu tunelu. Stejně jako fáze se udává v procentech od 0 % do 100 %, kde 100 % znamená nejlepší možný výsledek (vzdálenost od středu tunelu je menší nebo rovna jeho poloměru) a lineárně klesá k 0 % určených parametrem pro maximální vzdálenost od tunelu.

Podrobný popis výpočtu hodnotících parametrů se nachází v sekci 5.3.2

5.1.2 Nastavení parametrů

Základní konfigurace parametrů tunelu a vyhodnocování pohybu je určena v třídě *Settings*. Při integraci do rehabilitační aplikace se předpokládá mo-

¹stupeň volnosti, degree of freedom

difikace těchto parametrů například pomocí `.xml` souboru.

Váhy sensorů slouží pro určení důležitosti senzoru. Ke každému stupni volnosti každého senzoru se přiřazuje váha jako desetinné číslo z intervalu od 0 do 1. Takto lze třeba nastavit, že na polohu zápěstí je kladen větší důraz než na orientaci hlavy.

TunnelRadius udává poloměr tunelu. Pokud je vzdálenost senzoru od středu tunelu menší nebo rovna tomuto parametru, je pohyb vyhodnocen jako maximálně správný, tedy kvalita se rovná *100 %*.

MaximumQualityRange Pokud součet vzdáleností mezi nejbližším bodem trajektorie a aktuální polohou senzoru přesáhne tuto hodnotu, je kvalita vyhodnocena jako nulová.

DistanceRatioCoefficient určuje poměr vzdálenosti mezi hlavou a předloktím pro aktuálního uživatele a fyzioterapeuta, který cvik předváděl ($\frac{d_u}{d_t}$).

5.1.3 Vyhodnocování pohybu v AAPD

Pro vyhodnocení správnosti se v AAPD využívá datová struktura KD-strom, jejíž princip je vysvětlen v kapitole 2.3.2 práce [11]. Při vyhodnocování se berou v potaz váhy jednotlivých sensorů a jejich stupňů volnosti. Pro zvýšení přehlednosti je vytvořený `enum DataType`, který slouží k označení sensorů a jejich hodnot ve formátu `Data`. Enum může nabývat následujících hodnot:

- Head
- Chest
- Forearm
- Arm
- Unknown

Knihovna definuje rozhraní `IDataProcessor`, které mimo jiné implementuje metodu `Input`, jejímž kontraktem je přijímat data o transformaci jednotlivých sensorů (každému senzoru odpovídá právě jedno volání této metody) a metodu `Process`, které její kontrakt ukládá výpočet parametrů fáze a kvality z posledních dostupných dat.

V prvotní verzi AAPD jsou k dispozici dvě implementace tohoto rozhraní. První, `FileDataProcessor`, slouží k přehraní naměřených pohybů a

slouží čistě pro testovací účely. Druhá implementace `DataProcessor` už je určena pro real-time výpočet hodnotících parametrů při běhu rehabilitačního softwaru.

Vyhodnocení, zda byl celý cvik proveden správně, provádí samotná rehabilitační aplikace (viz 6.1). Hodnotícími parametry je zde dosažená fáze a kvalita. Tyto si nastavuje fyzioterapeut pro každý cvik a každého pacienta zvlášť v externím konfiguračním souboru, jehož struktura je vysvětlena v 6.1.3. Cvik se vyhodnotí jako správný, pokud hodnota aktuální fáze je větší nebo rovna než parametr `phase_target` a zároveň hodnota kvality cvičení je větší nebo rovna než `quality_threshold`. V případě nedosažení těchto hodnot nebude prováděný cvik uznán jako platný a je potřeba ho opakovat.

5.2 Požadavky na změny

První verze AAPD byla podrobena testování v druhé polovině roku 2019. Toto testování probíhalo na specializovaném pracovišti ergoterapie ve Fakultní nemocnici Královské Vinohrady v Praze a zúčastnilo se ho deset vybraných pacientů. Knihovna byla po celou dobu testování nastavena tak, aby při vyhodnocování brala v potaz především polohu předloktí (s váhou 1), zatímco u ostatních transformací byla váha nastavena na minimální hodnotu.

Po testování přišla řada požadavků na úpravu vyhodnocení správnosti pohybu. Následující výčet se týká způsobu vyhodnocení správnosti prováděného pohybu.

Změna výpočtu fáze Cílem je změnit výpočet fáze pohybu tak, aby cvik začínal v 0 % a končil ve 100 %, nikoliv opět v 0 %. Dojde tak k lepší představě o již vykonané části cviku a adekvátnějšímu vyhodnocení dosažené fáze.

Přidání nových cviků Pro rozšíření nabídky cviků přidat nové, jejichž cílem je opsat zápěstím spirálu umístěnou před pacientem. Toto cvičení by mělo být dostupné pro obě ruce a levotočivou i pravotočivou spirálu.

Vylepšení vyhodnocování Vylepšit systém vyhodnocování pohybu, důraz je kladen především na orientaci sledovaných částí těla a případnou vizualizaci správných hodnot ve virtuálním prostředí.

5.3 Realizace změn

5.3.1 Rozbor požadavků

Pro implementaci výše zmíněných požadavků bylo potřeba změnit způsob, jakým knihovna AAPD vyhodnocuje cvičení. Při testování AAPD v rámci práce [11] bylo zjištěno, že oproti použití KD-stromu se čas potřebný k nalezení nejbližšího bodu tunelu výrazně zkrátí při použití algoritmu na principu hrubé síly, kdy se prochází celá kolekce naměřených hodnot a porovnává jejich vzdálenost od aktuální transformace senzorů.

S ohledem na požadavky na nové cviky (spirály) bylo rozhodnuto o použití algoritmu hrubé síly, který snáze umožní omezit skupinu prohledávaných bodů na okolí posledního nalezeného nejbližšího bodu, nikoliv na celou množinu. Tímto se zamezí podvádění při cvičení, jelikož pacient se nebude moci skokově dostat do cíle, ale bude muset svým pohybem opsat celou trajektorii.

Další změna nastává ohledně hodnocení kvality. Kvalita bude vyhodnocována průběžně v průběhu celého cvičení způsobem, že pokud aktuální hodnota klesne pod prahovou hodnotu nastavenou parametrem `quality_threshold`, nebude umožněn další posun fáze a pacient se tedy bude muset vrátit do místa, kde došlo k poklesu kvality, a odtud pokračovat ve správném vykonávání cviku.

Pohyby bude potřeba rozdělit do dvou kategorií: s návratem a bez návratu. Pohyb bez návratu, jako například spirála, nevyžaduje žádné složitější vyhodnocení, po dosažení cílové fáze jednoduše stačí přejít k dalšímu opakování nebo cviku. Interval, kde k tomuto dojde, odpovídá části tunelu, kde $phase \geq phase_{target}$.

Oproti tomu u pohybu s návratem bude potřeba uchovávat informaci o maximální dosažené fázi a teprve pro návratu, respektive dosažení konce cviku, vyhodnotit, zda dosažená fáze překonala požadovanou hranici. Je potřeba tedy určit vrchol pohybu, nejlépe automaticky, bez nutnosti manuálního zásahu, a následně rozeznávat, zda se pacient pohybuje před tímto vrcholem nebo se již vrací do místa začátku, respektive konce. Začátek a konec cvičení jsou sice uloženy jako odlišné body, ale předpokládá se, že jsou v téměř totožné poloze, což vychází z charakteru cvičení.

5.3.2 Implementace

Součástí této práce je úprava první verze knihovny AAPD dle výše zmíněných požadavků. Změn doznalo rozhraní `IDataProcessor`. V metodě `Build` přibyl parametr `qualityThreshold`, jelikož vyhodnocování kvality nadále

probíhá přímo v procesoru², nikoliv v rehabilitační aplikaci. Dále bylo rozhraní rozšířeno o metodu `startNewMovement`, které její kontrakt udává reset cviku a tunelu do počátečního bodu.

Nové rozhraní `IApexProcessor`, které implementuje zároveň také rozhraní `IDataProcessor` je určené pro hodnocení pohybů s návratem. Metoda `getApexIndex` vrátí index vrcholu pohybu v kolekci zaznamenaných bodů. Poslední metoda, `isReturning` po zavolání vrátí logickou hodnotu, zda je aktuální fáze cviku za vrcholem a pacient směřuje ke konci cviku, nebo zda se naopak pacient k vrcholu teprve blíží.

Zároveň došlo k implementaci nových procesorů - `CombinedProcessor`, který kombinuje porovnávání vzdáleností poloh a orientace na základě konfigurace pomocí parametrů (vah) v konfiguračním souboru, a `ApexMovementProcessor` určený pro pohyby s návratem. Principy fungování těchto procesorů jsou popsány dále v této kapitole.

Výpočet hodnotících parametrů

KD-strom z první verze AAPD byl nahrazen algoritmem založeným na hrubé síle. Nový algoritmus prohledává kolekci bodů v okolí poslední zaznamenané pozice senzoru. Výpočet probíhá zvláště pro vzdálenost a orientaci.

Vzdálenost dvou pozic je určena podle vzorců 5.1 a 5.2 jako součet dílčích vzdáleností pro jednotlivé senzory vynásobených vahami jejich pozic c_k , které jsou určeny konfiguračními parametry. Dílčí vzdálenost d_k se vypočítá jako vzdálenost aktuální pozice senzoru od pozice zaznamenaného bodu v kolekci. Číslo k v těchto vzorcích značí číslo (index) senzoru a číslo i značí index v kolekci bodů.

$$d_{pos} = d_1 * c_1 + d_2 * c_2 + d_3 * c_3 + d_4 * c_4 \quad (5.1)$$

$$d_k = |pos_{i,k}, pos_{curr}| \quad (5.2)$$

Rozdíl orientací je vypočten jako velikost úhlu (v radiánech) mezi dvěma maticemi rotace pojmenovaných M a N . Výpočet velikosti úhlu probíhá podle vzorců 5.3 a 5.4 převzatých z [2].

$$R = MN^T \quad (5.3)$$

$$\alpha = \left| \arccos \left(\frac{tr R - 1}{2} \right) \right| \quad (5.4)$$

Operace $tr R$ je stopa, neboli *trace*, matice R . Jedná se o algebraickou operaci jejímž výsledkem je skalární hodnota. Stopa se získá součtem prvků

²Implementace rozhraní `IDataProcessor`

na diagonále matice. Pozorný čtenář si všimne, že tato operace lze zapsat i pomocí výrazu 5.5.

$$trR = tr(MN^T) \quad (5.5)$$

Na základě definice operace stopy tedy není v tomto případě nutné provádět celou operaci násobení matic M a N^T , nýbrž stačí vypočítat hodnotu prvků na diagonále matice R a tyto hodnoty následně sečíst. Odstraněním násobení celých matic se sníží složitost tohoto algoritmu o jeden řád na $O(n^2)$. Algoritmus je popsáný pseudokódem ve výpisu 5.1. Celková vzdálenost rotací je vypočtena opět jako součet dílčích vzdáleností vynásobených jejich vahami podle vzorce 5.6. Koeficienty o_i zde značí váhy orientací jednotlivých senzorů.

$$d_{rot} = \alpha_1 * o_1 + \alpha_2 * o_2 + \alpha_3 * o_3 + \alpha_4 * o_4 \quad (5.6)$$

Listing 5.1: Výpočet hodnoty stopy matice

```
double t = 0;
for (int i = 0; i < 3; i++) {
    for (int j = 0; j < 3; j++) {
        t += M[i][j] * NT[j][i];
    }
}
```

Porovnávání vzdáleností

Porovnání vzdáleností a nalezení nejbližšího obrazu aktuální pozice v naměřených datech je kontraktem metody `calcDistance` ve třídě `CombinedProcessor`. Celková vzdálenost aktuální pozice od obrazu v zaznamenaných datech je určena jako součet vzdáleností poloh a rotací vypočtených podle vzorce 5.7. Tyto hodnoty jsou již upraveny o váhy jednotlivých senzorů definovanými pomocí konfiguračních parametrů, tudíž není potřeba jejich žádná další úprava.

$$d_{total} = d_{pos} + d_{rot} \quad (5.7)$$

Postupně se prochází kolekce dat v okolí poslední určené pozice při posledním volání metody `Process`. V každé iteraci se vypočte vzdálenost $|p_i, p_{curr}|$ a ta se následně porovná s nejmenší vzdáleností získanou v předchozích iteracích. Před první iterací se hodnota nejlepší vzdálenosti vždy nastaví na `double.MaxValue`. Pokud je nová hodnota menší než nejmenší hodnota získaná z předchozích iterací, je nejlepší hodnota nastavena na novou hodnotu a současně se uloží i její index.

Tímto způsobem se prohledává okolí nejbližšího bodu, který je určený předchozím voláním metody `Process`, dokud vzdálenost bodu v kolekci $|p_i, p_{last}|$ nepřesáhne hodnotu `DistanceThreshold` ze třídy `Settings`, případně dokud není dosaženo konce kolekce.

Metoda `calcDistance` tímto postupným porovnáváním určí vzdálenost nejbližšího bodu v kolekci a jeho index. Ze vzdálenosti se vypočítá kvalita mapování z intervalu $\langle tunnelRadius, maxTunnelDistance \rangle$ do intervalu $\langle 1, 0 \rangle$. Pokud je celková vzdálenost $d \leq tunnelRadius$, je kvalita rovna 100 %, naopak pokud vzdálenost $d \geq maxTunnelDistance$, je kvalita vyhodnocena jako 0 %.

Hodnota `maxTunnelDistance` se získá podle vzorce 5.8. Hodnota π reprezentuje největší možný úhel mezi dvěma rotacemi, tedy největší možnou vzdálenost rotací.

$$\begin{aligned} maxTunnelDistance = \\ = \sum_{j=1}^4 \left((tunnelRadius + maxQualityDistance) * c_{pos}^j \right) + \sum_{j=1}^4 \left(\pi * c_{rot}^j \right) \end{aligned} \quad (5.8)$$

Jestliže kvalita překročí hodnotu parametru `qualityThreshold`, je aktualizována i hodnota fáze. Takto nedojde k posunu po trajektorii, dokud pacient nedosáhne požadované kvality.

Pro urychlení výpočtu se při vytváření instance procesoru v metodě `Build` vypočítají následující hodnoty:

Vzdálenosti mezi jednotlivými body se ukládají do pole `distanceLut`.

Hodnota v poli na indexu i je určena podle vzorce 5.9. Pole má stejnou délku jako kolekce `dat` a jeho poslední index je dosazena hodnota 0. Tímto způsobem se usnadní kontrola překročení délky pole v metodě `calcDistance`.

$$d_i = \sum_{j=1}^4 |p_i^j, p_{i+1}^j| * c_{pos}^j \quad (5.9)$$

Matice rotace jednotlivých zaznamenaných bodů jsou také vytvořeny předem a uloženy do dvourozměrného pole `frames`.

Pohyb s návratem

Pro vyhodnocení pohybu s návratem slouží třída `ApexMovementProcessor`. Vyhodnocování vzdáleností polohy a orientace funguje na stejném principu jako ve třídě `CombinedProcessor`. Při vytváření instance procesoru pro po-

hyb s návratem je navíc ještě vypočten index vrcholu pohybu. Vysvětlení postupu výpočtu vrcholu se nachází dále v této kapitole.

Při vyhodnocování pohybu se pak ještě bere v potaz, zda se pacient k tomuto vrcholu teprve blíží (cesta „nahoru“), nebo již směřuje ke konci cviku (cesta „dolů“). V průběhu cesty nahoru se postupně zvětšuje hodnota maximální dosažené fáze až do dosažení vrcholu. Při cestě dolů se tato hodnota již nezvětšuje a pouze probíhá kontrola, zda pohyb probíhá správně v zadaných mezích kvality.

Pacient může při vykonávání cvičení udělat chybu a začít se vracet ještě před dosažením vrcholu. Pokud je tato chyba způsobena například třesem, může nadále pokračovat v cvičení a snažit se cvik dokončit. Pokud ovšem rozdíl aktuální fáze a nejvýše dosažené fáze přesáhne hodnotu parametru `phaseDelta` s výchozí hodnotou 5 %, vyhodnotí se toto jako že cvik vzdal a vrcholu se mu nepodařilo dosáhnout. Poté je na trajektorii v intervalu $\langle apex, n \rangle$ nalezen nejbližší bod a dojde ke skokové změně fáze na fázi tohoto bodu. Následně probíhá návrat směrem k cíli bez možnosti vylepšení dosažené fáze.

Vyhodnocení dosažené fáze při cviku probíhá až po návratu do koncového bodu cviku. Dosažená fáze je určena podle vzorce 5.10 jako podíl maximální dosažené fáze a fáze vrcholu cviku. Tento postup je zvolen, protože vyhodnocení dosažení vrcholu je z hlediska celkového vyhodnocení pohybu nejdůležitější, a zároveň vrchol pohybu se nemusí nacházet vždy přesně v 50 % fáze cviku.

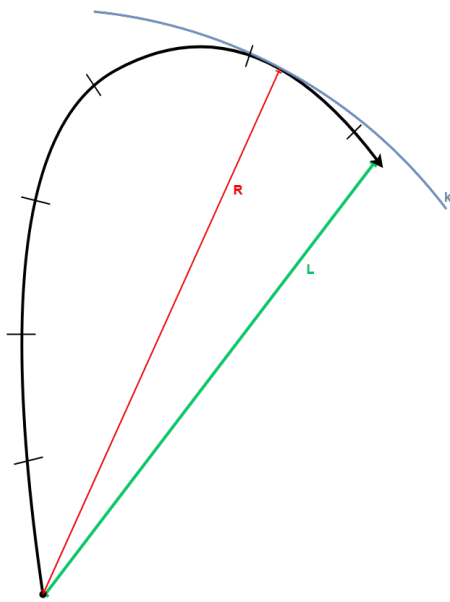
$$phase_{total} = \frac{phase_{max}}{phase_{apex}} \quad (5.10)$$

Výpočet vrcholu pohybu

Z důvodu zařazování nových cviků do aplikace není přípustné určovat vrchol pohybu manuálně. Z tohoto důvodu byl do implementace začleněn výpočet vrcholu cviku. Vrchol nelze určit na základě maximální vzdálenosti od počátečního bodu trajektorie, jelikož kvůli možnému elipsovitému tvaru trajektorie nemusí tento být skutečně vrcholem pohybu. Tento případ je znázorněn na obrázku 5.1. Poloměr R kružnice k odpovídá vzdálenosti nejvzdálenějšího bodu trajektorie od počátku.

Výpočet je tedy prováděn po menších segmentech, kde by, na základě znalosti dosavadních a nově požadovaných cviků, nemělo k tomuto jevu dojít. Výpočet vrcholu se provádí na naměřených datech ze senzoru na zápěstí, jelikož tam je poloměr pohybu největší a tudíž i trajektorie pohybu tohoto senzoru je nejdelší v porovnání s ostatními senzory.

Vrchol pohybu je vypočtený pomocí algoritmu naznačeném pseudokó-



Obrázek 5.1: Schéma hledání vrcholu pohybu.

dem ve výpisu 5.2. Hodnota `apexIndex` zde figuruje jako atribut třídy `ApexMovementProcessor`. Konstanta `STEP` udává velikost segmentu a hodnota této konstanty se rovná `30`. Proměnná `iBase` určuje index počátku segmentu. Na začátku má tedy hodnotu `0` a po průchodu segmentu se skokově mění o konstantu `STEP`. Při průchodu segmentem se postupně porovnává vzdálenost $lastDist = |d_{iBase}, d_i|$ s maximální dosaženou vzdáleností $maxDist$ v aktuální segmentu. Pokud pohyb pokračuje dále, tato hodnota se bude zvyšovat po celou dobu průchodu segmentem a na jeho konci tedy bude platit rovnost $lastDist = maxDist$. Ovšem v případě, že se v segmentu nachází vrchol pohybu, bude se vzdálenost $lastDist$ po jeho nalezení zmenšovat, tudíž na konci segmentu bude platit, že $lastDist < maxDist$ a prohledávání může být ukončeno. Index vrcholu pohybu je uložený jako atribut třídy v proměnné `apexIndex`.

Listing 5.2: Určení vrcholu pohybu.

```
int i = 0;
int iBase = 0;
DataCollection collection = allData[DataType.Forearm];

while (i < size) {
    float dist = 0;
    double maxDist = 0;
    double lastDist = 0;
    int j = 0;
    while (j < STEP && i < size) {
        lastDist = (collection[i].position -
                    collection[iBase].position).SqrtMagnitude();
        if (lastDist > maxDist) {
            maxDist = lastDist;
            apexIndex = i;
        }
        i++;
        j++;
    }
    if (lastDist < maxDist) {
        return;
    }
    iBase = i;
}
```

6 Integrace do rehabilitačního softwaru

Výsledná aplikace pro rehabilitaci ve virtuální realitě vznikla v rámci této práce a dále prací [7] a [9]. Aplikace umožňuje pacientovi provádět kontrolovanou terapii ve virtuální realitě pomocí předem určených vzorů cvičení a hraní jednoduchých her. Cílem je umožnit pacientovi vykonávat terapii ve virtuálním prostředí, včetně vizualizace správného provedení cviku, a zvýšit tak atraktivitu terapie.

6.1 Požadavky na aplikaci

6.1.1 Cvičení

Hlavní součástí aplikace by měla být možnost provádět kontrolované cvičení a jeho jasné vyhodnocení pomocí parametrů fáze a kvality. Cvičení se bude týkat horních končetin. Základní podoba cviků jsou PNF diagonály (viz 2.2.2), nácvik vstávání ze židle a obkreslení spirály v prostoru.

6.1.2 Hry

Pacientovi bude umožněno hrát jednoduché hry pro rozvinutí pohybu rukou. Základní hrou je sbírání míčků ze sedu v prostoru okolo pacienta na dosah ruky. Fyzioterapeut bude moci sám určit pozice těchto míčků a jejich pořadí.

6.1.3 Možnost konfigurace

Celá aplikace by měla být vhodně konfigurovatelná. Jednak musí být možnost nakonfigurovat vyhodnocování cvičení podle aktuálního fyzického stavu pacienta. Následně musí fyzioterapeut sestavit vhodnou kombinaci cviků a her. Je potřeba též vzít v potaz odlišné fyzické proporce fyzioterapeuta a pacientů, aby nedocházelo k chybám při vyhodnocování pohybu vlivem odlišné vzdálenosti mezi senzory na těle.

6.1.4 Vizualizace

Jedním z cílů aplikace ve virtuální realitě je umožnit pacientovi alespoň na chvíli přesun do jiného, byť virtuálního prostředí, a zvýšit tak atraktivitu

terapie. Další důležitou součástí se stane vizualizace správného pohybu končetiny, která navodí iluzi správně vykonávaného cviku, a měla by takto zvýšit pacientovu motivaci.

6.1.5 Vyhodnocování a statistiky

Hodnocení pacientova postupu a zlepšování v průběhu terapie se bude hodnotit na základě vygenerovaných statistik. Ty by měly obsahovat informace o maximální dosažené fázi, průměrné vzdálenosti od ideální trajektorie a průměrné kvalitě při vykonávaném cviku.

6.1.6 Záznam a přehrávání

Pro lepší vyhodnocení chyb při cvičení by mělo být umožněno zaznamenat vybrané cviky a tento záznam následně přehrát přímo v aplikaci. Pacient nebo terapeut si takto mohou prohlédnout pohyb přímo ve VR a snadno tak identifikovat případné chyby. Na záznamu by také měly být vidět ideální trajektorie pohybu.

6.2 Realizace

Aplikace byla realizována na pomoci herního enginu *Unity* a cílena na platformu *HTC Vive*. Její vývoj byl primárně cílem práce [7], v rámci které vznikla většina funkcionalit popsaných níže. Další práce [9] měla za cíl vytvořit důvěryhodné animace pohybu rukou a jejich textur, a to včetně možnosti automatické tvorby nové animace pro nový pohyb bez nutnosti manuálního zásahu. Výsledná aplikace byla v druhé polovině roku 2019 testována na specializovaném pracovišti ve Fakultní nemocnici Královské Vinohrady a na základě výsledků testování postupně upravována do finální podoby.

Práce [9] a [7] vznikaly současně s touto prací, *Sběr 3D dat pro rehabilitační software ve virtuální realitě*, takže jejich vývoj byl vzájemně provázaný a jednotlivé funkcionality jsou na sobě závislé.

6.2.1 Cvičení

Vyhodnocování průběhu cvičení probíhá knihovnou APPD detailně popsanou v sekci 5.1. Tato knihovna poskytuje informace o aktuální fázi, kvalitě a v případě cviku s návratem o maximální dosažené fázi cvičení. Vyhodnocení, zda pacient vykonal alespoň požadovanou fázi cviku, probíhá v samotné rehabilitační aplikaci. Pokud se jedná o cvik s návratem, po dosažení alespoň

95% fáze se aplikace dotáže knihovny na celkovou dosaženou fázi a pokud ta přesáhne hodnotu `targetPhase`, je cvik započten jako úspěšný. V případě pohybu bez návratu se pouze porovnává hodnota aktuálně dosažené fáze s `targetPhase`. Po úspěšném provedení cviku je uživatel informován přehráním zvuku a zvýšením počtu úspěšných cviků na informační tabuli před sebou, případně přesunem na další cvik.

Před začátkem cvičení se po stisknutí mezerníku celý tunel přesune na aktuální pozici hrudníku pacienta, která odpovídá neutrální pozici v zaznamenaných datech. V rámci každého cviku je pacientovi vykreslena ideální trajektorie pohybu (pro každý senzor zvlášť). Jedná se o barevně odlišené spojnice jednotlivých bodů trajektorie. Dále se v průběhu cvičení volitelně zobrazuje správná poloha ruky jako animace. Tato animace ruky vznikala jako součást práce [9].

Vizualizace orientace

Součástí této práce byla vizualizace správných orientací senzorů v průběhu cvičení. Pro přehledné zobrazení je využita barevná krychle (viz obrázek 6.1). Jsou použity tři barvy v souladu s barvami os v *Unity* (osa x - červená, osa y - zelená, osa z - modrá). Stěna krychle, která se nachází v kladném směru osy je vždy obarvena příslušnou barvou, zatímco stěna proti zůstává bez barvy. Ve vizualizaci se krychle zobrazují po přesném počtu bodů, který určuje konstanta `boxDensity` z třídy `TunnelController`.

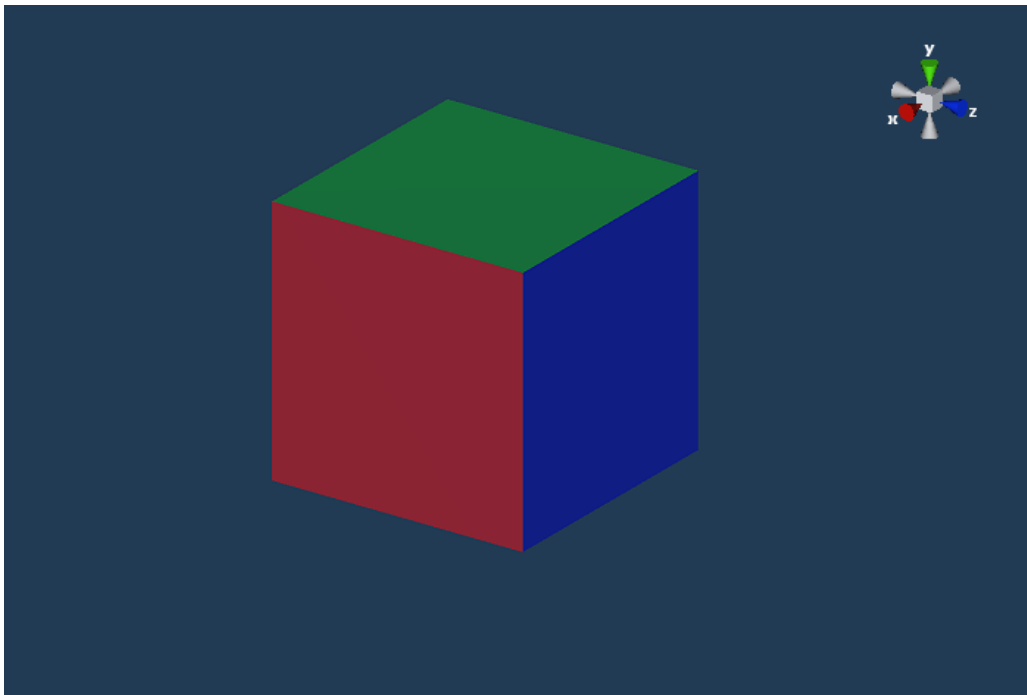
Vizualizace jednotlivých senzorů probíhá také těmito krychlemi doplněnými o název konkrétního senzoru. Snahou pacienta by tedy mělo být co nejvíce se přiblížit orientacemi senzorů orientacím vizualizovaným v tunelu.

6.2.2 Hry

První implementovanou hru představuje sbírání míčků. Pacient má za úkol sebrat míčky v prostoru před sebou pomocí vizualizované ruky. Barevně jsou odlišené sebrané míčky, nesebrané míčky a následující míček. Pokud následující míček není v dohledu, zobrazí se uživateli šipka směřující k němu. Pacient po celou dobu hry na informační tabuli před sebou vidí, jaké má skóre a kolik míčků mu ještě chybí sebrat.

Záznamová aplikace pro hru s míčky

Součástí této práce byla implementace samostatné aplikace pro vytvoření záznamu pozic míčků do hry. Záznam provádí terapeut, nikoliv pacient. Pro usnadnění ovládání má terapeut v ruce jeden controller. Po stisknutí tlačítka



Obrázek 6.1: Krychle pro vizualizaci orientace senzorů

`trigger` na tomto ovladači se uloží pozice bodu jako relativní poloha zápěstí vůči hrudníku. Takto lze zaznamenat libovolný počet bodů. Celá konfigurace pro hru je na konci záznamu, po stisknutí tlačítka `menu` na ovladači, uložena do `.xml` souboru ve formátu odpovídajícímu výpisu 6.1.

Listing 6.1: Formát konfigurace pro hru s míčky.

```
<?xml version="1.0" ?>
<configuration>
  <point index="0">
    <x>0,1739491</x>
    <y>0,6412234</y>
    <z>0,0942809</z>
  </point>
  <point index="1">
    <x>0,6818695</x>
    <y>0,1146753</y>
    <z>-0,1950586</z>
  </point>
  <point index="2">
    <x>-0,3152865</x>
    <y>0,1878939</y>
    <z>0,06403685</z>
  </point>
</configuration>
```

6.2.3 Konfigurace

Aplikace musí být přizpůsobitelná jednotlivým pacientům a specifikům jejich terapie. Musí být umožněno vytvořit různé kombinace her, cvičení a hodnotících kritérií. Konfigurační soubory jsou uloženy ve formátu `.xml`. Celá konfigurace aplikace je podrobně popsána v práci [7].

Cviky a hry

Základní konfigurační vlastností aplikace je možnost nakombinovat jednotlivé cviky, hry a jejich opakování. U cviků se dále určují hodnotící parametry jako cílová fáze a kvalita, kterých musí pacient při cvičení dosáhnout.

Definice cviků

Pro každý pohyb se musí určit, kde na disku se nachází příslušná sada naměřených dat, zda je pohyb určený pro pravou nebo levou ruku, a zda se jedná o pohyb s návratem či bez návratu. Definice cviků také obsahuje cestu k datům pro animaci pohybu, které jsou uloženy v samostatném `.csv` souboru. Takto je umožněno přidávat nové cviky bez nutnosti překladu a sestavení celé aplikace.

Konfigurace tunelu

Pro tunel existuje samostatný konfigurační soubor, který umožňuje nastavovat parametry jako váhy jednotlivých senzorů a mezní vzdálenosti pro určování kvality. Podrobné vysvětlení parametrů pro konfiguraci tunelu se nachází v sekci 5.1.2.

Samostatně probíhá konfigurace parametru `distaceRatioCoeficient`, který slouží ke korekci dat na základě různých fyzických proporcí terapeuta a uživatele. Pro tuto konfiguraci slouží samostatná scéna. Pro určení příslušného parametru musí pacient, případně terapeut, upažit a následně se změřit vzdálenost mezi headsetem a senzorem na zápěstí. Hodnota `distaceRatioCoeficient` je určena jako poměr této vzdálenosti u terapeuta a pacienta. S tímto parametrem pracuje knihovna AAPD, kde se podle něho upravují pozice jednotlivých trackerů při tvorbě tunelu.

6.2.4 Statistiky

Aplikace poskytuje zpětnou vazbu fyzioterapeutovi v podobě zaznamenaných statistik o provádění cviku. Jako hodnotící parametry byly určeny dosažená fáze, průměrná kvalita po dobu vykonávání cviku a průměrná vzdálenost od ideální trajektorie. Tyto hodnoty se po vykonání cvičení ukládají do samostatného `.xml` souboru. Implementace této funkcionality je podrobně popsána v práci [7].

6.2.5 Vizualizace

V aplikaci se odlišují dva různé pohledy. Pacient s headsetem cvičí ve virtuálním prostředí a fyzioterapeut má možnost na monitoru sledovat průběh cvičení a ovládat celou aplikaci.

Virtuální prostředí

Virtuální prostředí, ve kterém se nachází pacient, je založeno na modelu zobrazujícím *Stonehenge*. Pacient sedí přímo uprostřed kruhů z kamenů a před sebou vidí informační tabuli, kde se zobrazují instrukce a informace o cvičení. V průběhu cviku se zobrazuje ideální trajektorie tunelu, kterou by měl cvičící následovat svým pohybem. V této trajektorii jsou zvýrazněné body, které se nacházejí nejbližší aktuální pozici pacienta. V průběhu cviku se zobrazuje animace ruky a jejího ideálního pohybu. Umístění modelu ruky lze upravovat pomocí šipek tak, aby co nejvíce odpovídal pozici skutečné ruky.

Tvorba animací a textur je součástí práce [9], kde jsou podrobně popsány principy a tvorba animací.

Pro usnadnění nácviku vstávání je scéna doplněna o model židle, aby pacient neměl pocit, že si sedá do prázdna. Pozice židle se určuje na základě dat z trackeru umístěného na zadní straně opěradla.

Pohled fyzioterapeuta

Fyzioterapeut může průběh celého cvičení sledovat na monitoru. Zde se mu v aplikačním okně zobrazuje především pohled pacienta. Dále je okno doplněno o postranní panely pro konfiguraci a sledování vyhodnocování pohybu. Tyto panely lze po stisknutí tlačítka na okraji minimalizovat pouze na lištu. Pravý panel umožňuje ukládat aktuální pozici ruky a měnit průhlednost ruky tak, aby nerušila pacienta při cvičení. V levém panelu se zobrazují data o cvičení jako je fáze, kvalita, aktuální cvik, počty opakování a tak dále. Pro ladící účely je možnost přidat ještě panel, kde se zobrazuje aktuální hodnota sledované proměnné.

6.2.6 Záznam a přehrávání pohybů

Pro analýzu pohybu jinak než pomocí hodnotících parametrů byla aplikace rozšířena o záznam a přehrávání cviků. Rozšíření aplikace probíhalo v rámci této práce. Záznam dat probíhá stejně jako v již zmíněné záznamové aplikaci. Přibylo zaznamenání dat z tunelu, které jsou uloženy do samostatných souborů. Vzhledem k tomu, že se jedná pouze o body, ukládá se pouze jejich pozice v absolutní hodnotě. Orientace v tomto případě není potřeba, jelikož tunel je vykreslený jako spojnice bodů. Záznam ovládá terapeut v panelu umístěném v aplikačním okně na monitoru počítače. Před záznamem musí být zvolena složka, kam se budou zaznamenána data ukládat. Začátek i konec měření se určí stiskem tlačítka na monitoru.

Zaznamenaná data se následně dají přehrát jako animace. Ovládání funkcionality provádí terapeut také na počítači. Po výběru složky je vytvořen nový `AnimationClip`, který se následně v nekonečné smyčce přehrává. Zobrazí se i celá trajektorie pohybu s barevně odlišenými senzory, stejně jako při cvičení. Bíle jsou zobrazeny správné trajektorie. Terapeut či pacient si mohou zaznamenané pohyby v průběhu přehrávání prohlédnout pomocí headsetu a analyzovat případné chyby. Při přehrávání nedochází k výpočtu hodnotících parametrů, analýza pohybu je možná pouze vizuálně.

7 Experimentální část

Součástí práce byl experiment, který měl ověřit výsledky vyhodnocování pohybu pro různé konfigurační parametry. Hlavním cílem bylo zjistit, zda je možné provádět analýzu pohybu pouze na základě údajů o orientaci senzorů, což by zjednodušilo nároky na tracking systém a umožnilo použít konstrukčně jednodušší a levnější IMU senzory. Tímto by se zvýšila přenositelnost celé aplikace a bylo by možné ji poskytnout pacientům pro domácí terapii bez složité instalace sledovacích stanic a konfigurace celého systému.

7.1 Scénář experimentu

Aby se zamezilo chybě vlivem odlišného provedení cviku, musí porovnání probíhat nad stejnými záznamy pohybu. Zaznamenané pohyby budou vyhodnoceny pomocí procesorů z knihovny AAPD. Konfigurace knihovny se bude lišit v nastavených vahách jednotlivých senzorů a jejich poloh a orientací. Figurantovi, kterého představuje zdravý jedinec, bude umožněno krátké seznámení s aplikací a vyzkoušení pohybu ve virtuálním prostředí. Následně dojde k záznamu tří cviků:

- 1. diagonála
- 3. diagonála
- spirála

Tyto cviky jsou zvoleny tak, aby byly co nejvíce odlišné a reprezentovaly jak pohyb s návratem, tak pohyb bez návratu. Aby vyhodnocení probíhalo správně, musí se ještě uložit i transformace, na kterou byl před začátkem měření transformován tunel. Stejná transformace se musí aplikovat na tunel před začátkem každého vyhodnocení.

7.2 Konfigurace AAPD

Pro získání větší variability výsledků byly zaznamenány výše zmíněné cviky od dvou figurantů. Vyhodnocení probíhalo s třemi různými konfiguracemi knihovny. Všechny pohyby měly nastavený parametr cílové fáze na 90 % a práh kvality na 30 %. První konfigurace využívá základní nastavené váhy

senzorů, jak je popsáno ve výpisu 7.1, které využívá jak data o poloze, tak o orientaci.

Listing 7.1: Základní konfigurace knihovny AAPD

```
<settings>
  <tunnel_radius>0.1</tunnel_radius>
  <maximum_quality_range>0.5</maximum_quality_range>
  <coefficients>
    <data_type name="head">
      <position x="0.5" y="0.5" z="0.5" />
      <rotation x="0.00001" y="0.00001"
              z="0.00001" />
    </data_type>

    <data_type name="forearm">
      <position x="1" y="1" z="1" />
      <rotation x="0.5" y="0.5" z="0.5" />
    </data_type>

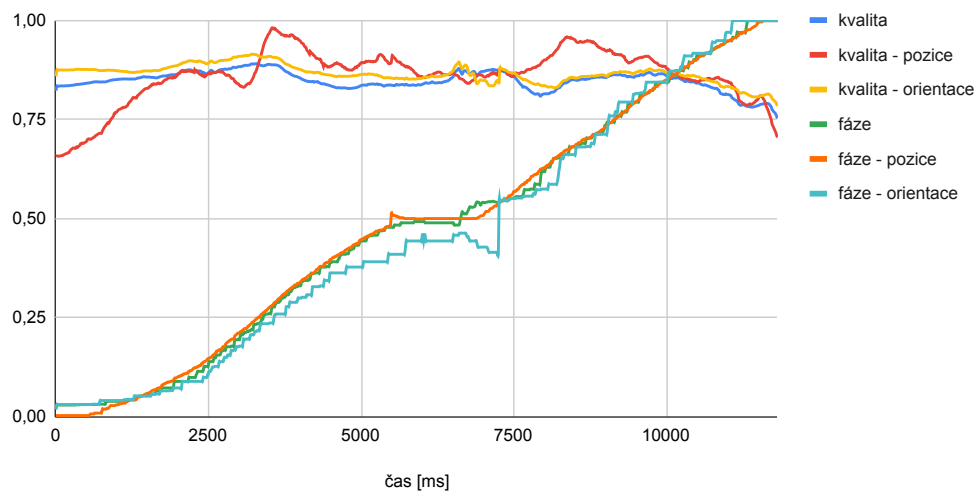
    <data_type name="arm">
      <position x="1" y="1" z="1" />
      <rotation x="0.5" y="0.5" z="0.5" />
    </data_type>

    <data_type name="chest">
      <position x="1" y="1" z="1" />
      <rotation x="0.5" y="0.5" z="0.5" />
    </data_type>
  </coefficients>
</settings>
```

Druhá konfigurace využívá čistě data o pozici, zatímco třetí pouze data o orientaci senzorů. U obou těchto konfigurací je nastavena poloviční váha senzoru na hlavě oproti ostatním senzorům. Parametry rozměrů tunelu pro vyhodnocení kvality zůstávají stejné jako v prvním případě u všech tří konfigurací.

Diagonála 1

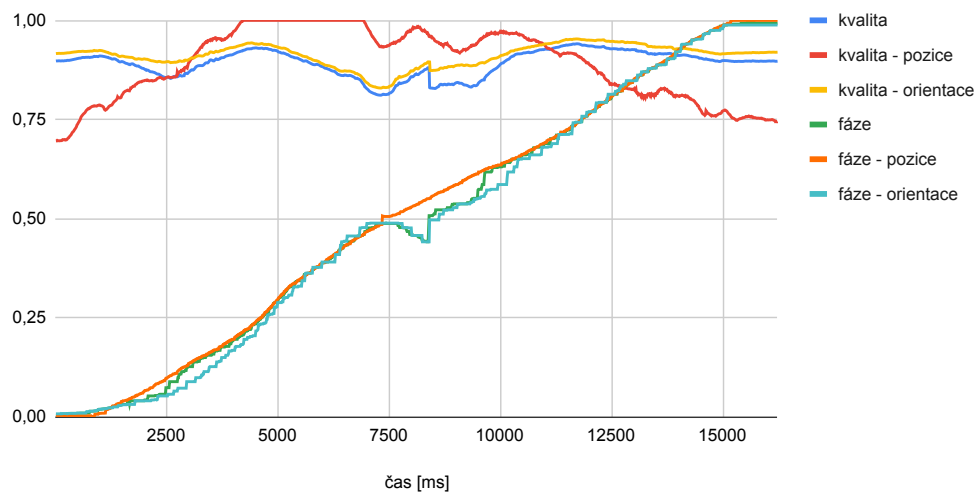
Figurant 1



Obrázek 7.1: 1. diagonála, figurant 1

Diagonála 1

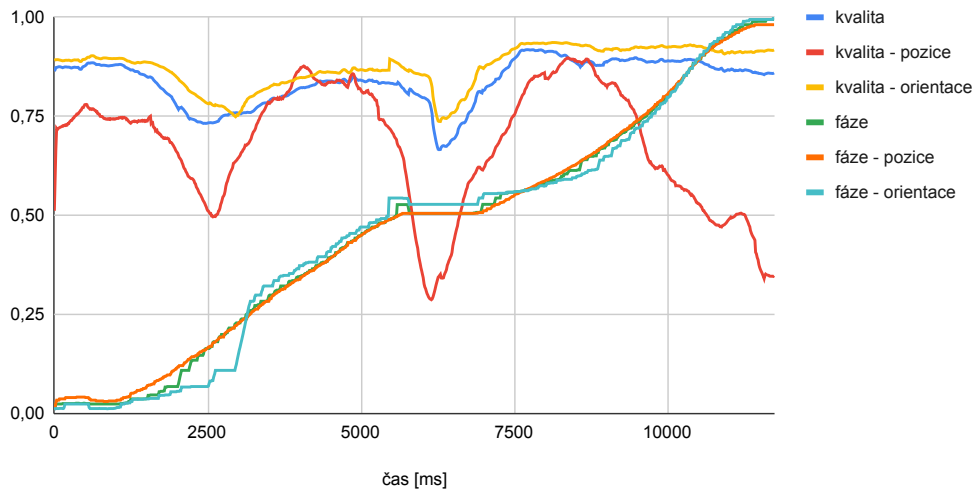
Figurant 2



Obrázek 7.2: 1. diagonála, figurant 2

Diagonála 3

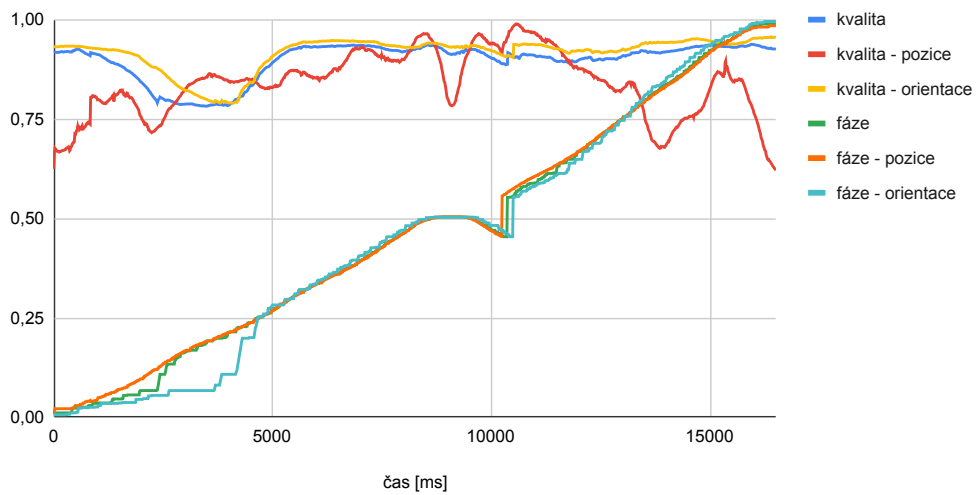
Figurant 1



Obrázek 7.3: 3. diagonála, figurant 1

Diagonála 3

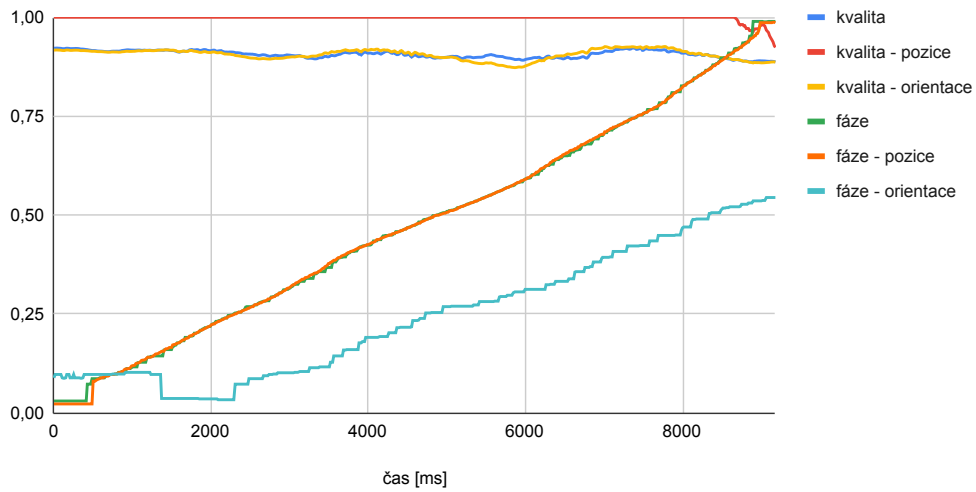
Figurant 2



Obrázek 7.4: 3. diagonála, figurant 2

Spirála

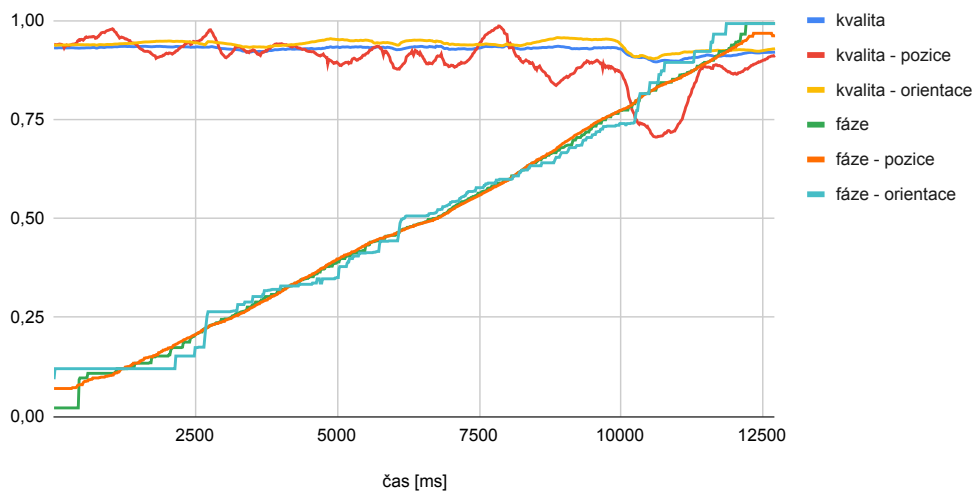
Figurant 1



Obrázek 7.5: Spirála, figurant 1

Spirála

Figurant 2



Obrázek 7.6: Spirála, figurant 2

7.3 Vyhodnocení experimentu

Výsledky experimentu jsou zobrazeny v grafech výše. Datové řady bez přívlastku se vztahují ke standardní konfiguraci. Řady s přívlastkem vždy odpovídají příslušné konfiguraci vysvětlené v sekci 7.2.

U grafů z 1. diagonály (7.1 a 7.2) je vidět přibližně shodný průběh fáze pro všechny druhy vyhodnocení. V některých případech lze okolo 50% fáze vidět skokovou změnu, kterou zapříčinilo dosažení vrcholu a následný návrat (vysvětleno v 5.2). Jedná se tedy o očekávané chování. Vyhodnocení kvality se po většinu času u obou figurantů drží nad hodnotou 75 %. Z rozdílů kvality pro rotace a orientace je vidět, že figurant 1 se držel více ideální trajektorie tunelu, ale nepovedlo se mu tolik dodržet požadovanou rotaci ruky (klesající kvalita v poslední čtvrtině záznamu u 7.1). Figurant 2 se naopak snažil více dodržet požadované rotace a pravděpodobně nekladl takový důraz na pozici, čemuž nasvědčuje pokles kvality pouze u pozice v obrázku 7.2, nikoliv u ostatních konfigurací.

3. diagonála (obrázek 7.3 a obrázek 7.4) představuje cvik s podstatně větším rozsahem pohybu než 1. diagonála. U grafu pro figuranta 1 lze pozorovat značné kolísání kvality při použití pouze pozice, zejména pak v okolí vrcholu pohybu (okolo fáze 50 %). Průběh fáze je opět téměř shodný u obou grafů, pouze u figuranta 2 lze pozorovat mírnou odchylku orientací od ostatních hodnot na začátku měření. Tato odchylka bude pravděpodobně způsobena pozdním zahájením abdukce ruky.

Grafy z posledního cviku, spirály, jsou na obrázku 7.5 a obrázku 7.6. V grafech lze vidět téměř shodný průběh fáze u vyhodnocení se standardními parametry a u vyhodnocení na základě pozic. Zajímavé je porovnání hodnot fáze u orientací. U figuranta 1 je jasně vidět, že fáze u orientace roste přibližně poloviční rychlostí oproti fázi u jiných způsobů vyhodnocení, zatímco u figuranta 2 je vidět, že průběh fáze sice není úplně jemný, ale přesto přibližně kopíruje průběh fáze u ostatních hodnot. Tyto změny pravděpodobně budou způsobeny pouze malými rozdíly v orientacích senzorů, které vycházejí z definice tohoto cviku. Průběh kvality splňuje očekávání a po většinu času se pohybuje u horní hranice intervalu.

8 Závěr

Všechny vzniklé části této práce byly ve spolupráci s ostatními členy týmu zahrnuty do rehabilitační aplikace a ta byla následně předána specializovanému pracovišti ergoterapie ve Fakultní nemocnici Královské Vinohrady. Na základě prvotních měření bylo rozhodnuto, že vývoj této aplikace bude probíhat na platformě *HTC Vive*.

Vytvořená záznamová aplikace zajišťuje záznam nových cviků a manuální segmentaci. Jejím použitím fyzioterapeut zaznamená libovolné nové pohyby sám, bez nutnosti přítomnosti vývojáře. Nahrazení KD-stromu v knihovně AAPD algoritmem založeným na hrubé síle umožnilo přesnější vyhodnocování pohybu, zejména pak díky oddělenému výpočtu pro složky pozice a orientace, což potvrdilo i závěrečné měření v kapitole 7. Rozdělení pohybů na pohyby s návratem a bez návratu přineslo vhodnější způsob vyhodnocování dosažené fáze. Vyhodnocování kvality po celou dobu cviku zase umožnilo průběžnou kontrolu prováděného pohybu, nikoliv pouze v závěru cviku.

Rehabilitační aplikace splňuje požadavky na provádění terapie ve virtuální realitě u pacientů s roztroušenou sklerózou. Momentálně je aplikace závislá na přítomnosti fyzioterapeuta, který určuje pořadí pohybů a nastavuje parametry pro jejich vyhodnocování. U této aplikace se předpokládá budoucí vývoj cílený na možné domácí použití bez nutnosti asistence další osoby.

Výsledky závěrečného experimentu této práce nevyvrací možnost použití samostatných IMU senzorů pro vyhodnocování pohybu. Při vyhodnocení pouze na základě orientace senzorů bylo ve většině případů dosaženo téměř schodných výsledků jako při použití pozic i orientací. Je potřeba ale zohlednit, že *SteamVR Tracking* poskytuje velmi přesná a kalibrovaná data. Oba figuranti byli zdraví a vykonávání těchto cviků jim nečinilo žádný problém z hlediska motoriky. U reálných pacientů se dají předpokládat problémy s třesem a pohyblivostí rukou. Při zohlednění těchto faktorů se dá říci, že by bylo možné některé cviky vyhodnocovat pouze pomocí orientace, ale data ze senzorů by musela být přesná a pravděpodobně by bylo potřeba senzory pravidelně kalibrovat (například uvedením do neutrální polohy). Vyhodnocování se stává složité zejména u spirál, kdy se těžko odhaduje, v jaké fázi se pacient nachází, jelikož rozdíly rotací jsou velmi malé. Nejpřesnější výsledky poskytovala kombinace pozice i orientace, čemuž nasvědčovaly i teoretické předpoklady. Použití samostatných rotací by pravděpodobně bylo pro některé cviky možné, ovšem tato metoda by před integrací do rehabilitačního

softwaru vyžadovala další testování s použitím konkrétního hardwaru jako jsou IMU senzory.

Seznam obrázků

2.1	Simulátor virtuálního prostředí Sensorama [1]	13
2.2	Schéma Vive trackeru [13]	14
3.1	Naměřená data orientace ze sensorů IMU na předloktí	22
3.2	Naměřená data orientace z HTC Vive z trackeru na předloktí	23
3.3	Naměřená data polohy z HTC Vive z trackeru na předloktí	23
5.1	Schéma hledání vrcholu pohybu.	36
6.1	Krychle pro vizualizaci orientace sensorů	41
7.1	1. diagonála, figurant 1	47
7.2	1. diagonála, figurant 2	47
7.3	3. diagonála, figurant 1	48
7.4	3. diagonála, figurant 2	48
7.5	Spirála, figurant 1	49
7.6	Spirála, figurant 2	49

Literatura

- [1] ALRAIZZAH, A. – LAMYA, F. – FATTOUH, L. Environments and System Types of Virtual Reality Technology in STEM: A Survey. *International Journal of Advanced Computer Science and Applications*. 06 2017, 8. doi: 10.14569/IJACSA.2017.080610.
- [2] BELOUSOV, B. *Distance between rotations* [online]. 2016. [cit. 2020/04/16]. Dostupné z: <http://www.boris-belousov.net/2016/12/01/quat-dist/>.
- [3] CHRIS, W. *Virtual reality* [online]. 2007. [cit. 2020/03/03]. Dostupné z: <https://www.explainthatstuff.com/virtualreality.html>.
- [4] ENDICOTT, S. *Oculus Quest: Everything you need to know* [online]. androidcentral, 2019. [cit. 2020/04/18]. Dostupné z: <https://www.androidcentral.com/oculus-quest#summary>.
- [5] *Kabatova metoda* [online]. FYZIOklinika. [cit. 2020/02/24]. Článek o kabátově metodě. Dostupné z: <https://www.fyzioklinika.cz/clanky-o-zdravi/proprioceptivni-nervosvalova-facilitace-kabatova-metoda>.
- [6] KIM, Y. – BANG, H. Introduction to Kalman Filter and Its Applications. 2019. doi: 10.5772/intechopen.80600. Dostupné z: <https://doi.org/10.5772/intechopen.80600>.
- [7] KODERA, J. Automatická analýza pohybu ramene pro účely rehabilitace ve virtuální realitě, bakalářská práce. Vedoucí práce Libor Váša. *Západočeská univerzita v Plzni*. 2020.
- [8] KOK, M. – HOL, J. D. – SCHÖN, T. B. Using Inertial Sensors for Position and Orientation Estimation. *Foundations and Trends® in Signal Processing*. 2017, 11, 1-2, s. 1–153. ISSN 1932-8346. doi: 10.1561/20000000094. Dostupné z: <http://dx.doi.org/10.1561/20000000094>.
- [9] KÖNIG, A. 3D animace paže podle naměřených dat pro účely rehabilitace ve virtuální realitě, bakalářská práce. Vedoucí práce Libor Váša. *Západočeská univerzita v Plzni*. 2020.
- [10] OLŠAN, J. *Oculus Go uveden. Mobilní autonomní brýle přináší virtuální realitu za nízkou cenu* [online]. cnews, 2018. [cit. 2020/04/18]. Dostupné z: <https://www.cnews.cz/facebook-oculus-go-virtualni-realita-vydani>.

- [11] POÓR, V. Automatická analýza pohybu ramene pro účely rehabilitace ve virtuální realitě, bakalářská práce. Vedoucí práce Libor Váša. *Západočeská univerzita v Plzni*. 2019.
- [12] SIBŘINOVÁ, H. *Co je to roztroušená mozkomíšní skleróza* [online]. 2016. [cit. 2020/02/20]. Článek o roztroušené skleróze. Dostupné z: <http://ereska.cz/rs/index.html>.
- [13] *HTC Vive Tracker Developer Guidelines, Ver. 1.3* [online]. HTC Corporation, 2017. [cit. 2020/04/29]. Dostupné z: https://dl.vive.com/Tracker/Guideline/HTC_Vive_Tracker_Developer_Guidelines_v1.3.pdf.
- [14] VITILLO, A. *All you need to know about SteamVR Tracking 2.0* [online]. skarredghost, 2017. [cit. 2020/04/20]. Dostupné z: <https://skarredghost.com/2017/06/07/need-know-steamvr-tracking-2-0-will-foundation-vive-2/>.
- [15] ŘASOVÁ, K. et al. *Možnosti fyzioterapie v léčbě roztroušené sklerózy mozkomíšní*. Unie Roska, Česká MS společnost, 2008. ISBN 978-80-254-3704-9.
- [16] ŘASOVÁ, K. *Fyzioterapie u neurologicky nemocných*. Ceros, 2007. ISBN 978-80-239-9300-4.

Přílohy

Na vloženém disku se nacházejí následující přílohy:

AAPD Solution Tato složka obsahuje zdrojové kódy knihovny AAPD.

V podsložce **release** se nachází sestavená knihovna ve formátu `.dll`.
Repozitář s knihovnou je dostupný na adrese <https://gitlab.com/frankkuba/AAPD>.

Ball Configurator V této složce je umístěný *Unity* projekt se záznamovou aplikací pro hru s míčky. Repozitář s tímto projektem je dostupný na adrese <https://gitlab.com/frankkuba/ball-configurator>.

IMU Data Recorder Složka obsahující zdrojové kódy a aplikaci pro záznam dat z IMU senzorů pomocí UDP streamu. Repozitář pro tuto aplikaci je dostupný na adrese <https://bitbucket.org/frankkuba/imu-data-recorder>.

Návod na použití rehabilitačního SW PDF soubor s uživatelskou příručkou pro rehabilitační software. Tento soubor vznikl v rámci této práce a dále prací [7] a [9].

Recorder Složka s *Unity* projektem se záznamovou aplikací pohybů pro rehabilitační software.

VR Rehabilitation Build Složka obsahující sestavenou rehabilitační aplikaci. V podsložce `VR Arm Rehabilitation_Data/StreamingAssets` se nachází konfigurační soubory a zaznamenané předlohy pohybů.

VR Rehabilitation Project Složka obsahující *Unity* projekt s rehabilitační aplikací. Repozitář s projektem se nachází na adrese <https://gitlab.com/frankkuba/vr-arm-motion>.