

Západočeská univerzita v plzni
Fakulta strojní

Studijní program: B0715A270013 Strojní inženýrství

Studijní specializace: Průmyslové inženýrství a management

BAKALÁŘSKÁ PRÁCE

Využití hand tracking technologie ve virtuální realitě

Autor: Radek RAJTMAJER

Vedoucí práce: Ing. Jan KUBR

Akademický rok 2021/2022

ZÁPADOČESKÁ UNIVERZITA V PLZNI

Fakulta strojní

Akademický rok: 2021/2022

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Radek RAJTMAJER**
Osobní číslo: **S19B0480P**
Studijní program: **B0715A270013 Strojní inženýrství**
Specializace: **Průmyslové inženýrství a management**
Téma práce: **Využití hand tracking technologie ve virtuální realitě**
Zadávací katedra: **Katedra průmyslového inženýrství a managementu**

Zásady pro vypracování

1. Úvod
2. Popis technologie hand trackingu a možnosti ovládnání ve VR
3. Analýza současného stavu
4. Analýza možností implementace
5. Popis realizace vlastní implementace
6. Závěr

Rozsah bakalářské práce: **30 – 40 stran**
Rozsah grafických prací: **0**
Forma zpracování bakalářské práce: **tištěná**

Seznam doporučené literatury:

1. OKITA, A. *Learning C# Programming with Unity 3D*. Second edition, Boca Raton, FL. USA: Routledge, 2019. 690 p. ISBN-13: 978-1138336810.
2. SUNG, K., SMITH, G. *Basic Math for Game Development with Unity 3D: A Beginner's Guide to Mathematical Foundations*. Bothel, WA. USA: Apress, 2019. 424p. ISBN 978-1484254424.
3. LINOWES, J. *Unity Virtual Reality Projects: Learn Virtual Reality by developing more than 10 engaging projects with Unity 2018*. 2nd Edition, Birmingham, UK: Packt Publishing, 2018. 492 p. ISBN 978-1788478809.
4. LaVALLE, S. M. *Virtual Reality*. Cambridge University Press, 2020. 390 p., dostupné zdarma online na <http://lavalle.pl/vr/>
5. *Oficiální Unity3D návody dostupné na <https://learn.unity.com/>*

Vedoucí bakalářské práce: **Ing. Jan Kubr**
Katedra průmyslového inženýrství a managementu

Konzultant bakalářské práce: **Ing. David Krákora**
Katedra průmyslového inženýrství a managementu

Datum zadání bakalářské práce: **20. září 2021**
Termín odevzdání bakalářské práce: **27. května 2022**

L.S.

Doc. Ing. Milan Edl, Ph.D.
děkan

Doc. Ing. Michal Šimon, Ph.D.
vedoucí katedry

Prohlášení o autorství

Předkládám tímto k posouzení a obhajobě bakalářskou práci, zpracovanou na závěr studia na Fakultě strojní Západočeské univerzity v Plzni.

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně, s použitím odborné literatury a pramenů, uvedených v seznamu, který je součástí této bakalářské práce.

V Plzni dne:

.....

podpis autora

Poděkování

Rád bych poděkoval Ing. Janu Kubrovi za vstřícnost při konzultacích a radách při tvorbě bakalářské práce.

ANOTAČNÍ LIST BAKALÁŘSKÉ PRÁCE

AUTOR	Příjmení Rajtmajer	Jméno Radek	
STUDIJNÍ OBOR	B0715A270013 Strojní inženýrství		
VEDOUCÍ PRÁCE	Příjmení (včetně titulů) Ing. Kubr	Jméno Jan	
PRACOVNÍŠTĚ	ZČU - FST - KPV		
DRUH PRÁCE	DIPLOMOVÁ	BAKALÁŘSKÁ	Nehodící se škrtněte
NÁZEV PRÁCE	Využití hand tracking technologie ve virtuální realitě		

FAKULTA	strojní	KATEDRA	KPV	ROK ODEVZD.	2022
----------------	---------	----------------	-----	--------------------	------

POČET STRAN (A4 a ekvivalentů A4)

CELKEM	59	TEXTOVÁ ČÁST	45	GRAFICKÁ ČÁST	0
---------------	----	---------------------	----	----------------------	---

<p>STRUČNÝ POPIS (MAX 10 ŘÁDEK)</p> <p>ZAMĚŘENÍ, TÉMA, CÍL POZNATKY A PŘÍNOSY</p>	<p>Bakalářská práce se zabývá seznámením s technologií hand trackingu ve virtuální realitě, analýzou současného stavu a následné implementace. Cílem je prozkoumání možností implementace a výběr varianty s největším potenciálem. Implementace je prováděna pomocí programu Unity pro VR brýle Oculus Quest 2.</p>
<p>KLÍČOVÁ SLOVA</p> <p>ZPRAVIDLA JEDNOSLOVNÉ POJMY, KTERÉ VYSTIHUJÍ PODSTATU PRÁCE</p>	<p>VR, Virtuální realita, hand tracking, Unity, Oculus Quest 2, Oculus Integration, HPTK,</p>

SUMMARY OF BACHELOR SHEET

AUTHOR	Surname Rajtmajer	Name Radek	
FIELD OF STUDY	B0715A270013 Strojní inženýrství		
SUPERVISOR	Surname (Inclusive of Degrees) Ing. Kubr	Name Jan	
INSTITUTION	ZČU - FST - KPV		
TYPE OF WORK	DIPLOMA	BACHELOR	Delete when not applicable
TITLE OF THE WORK	Use of hand tracking technology in virtual reality		

FACULTY	Mechanical Engineering	DEPARTMENT	KPV	SUBMITTED IN	2022
----------------	------------------------	-------------------	-----	---------------------	------

NUMBER OF PAGES (A4 and eq. A4)

TOTALLY	59	TEXT PART	45	GRAPHICAL PART	0
----------------	----	------------------	----	-----------------------	---

BRIEF DESCRIPTION TOPIC, GOAL, RESULTS AND CONTRIBUTIONS	The bachelor thesis deals with the introduction of hand tracking technology in virtual reality, analysis of the current state and subsequent implementation. The aim is to explore the implementation options and select one with the greatest potential. The implementation is carried out using the Unity software for the VR headset Oculus Quest 2.
KEY WORDS	VR, Virtual realita, hand tracking, Unity, Oculus Quest 2, Oculus Integration, HPTK,

Obsah

Úvod	14
1 Popis technologie hand trackingu a možností ovládání ve VR	15
1.1 Virtuální realita	15
1.1.1 Definice VR	15
1.1.2 Komponenty VR	16
1.2 VR zobrazovací zařízení	16
1.3 Motion tracking	17
1.3.1 Trackovací systém „Constellation“ (konstelace)	18
1.3.2 Trackovací systém Lighthouse	19
1.3.3 Trackovací systém SLAM / Inside-Out	20
1.4 Ovládání VR	21
1.5 Využití VR a hand trackingu	22
1.6 Softwary	22
2 Analýza současného stavu	25
2.1 Ovladače	25
2.1.1 Oculus Touch	25
2.1.2 Ovladač HTC Vive	26
2.1.3 Ovladač PlayStation Move	26
2.2 Hand tracking	27
2.2.1 Rukavice s markery	27
2.2.2 Hand tracking v Oculus Quest	28
2.2.3 Ultraleap hand tracking	29
2.3 Výběr VR brýlí	30
3 Analýza možností implementace	31
3.1 HurricaneVR – Rozšíření skriptu	33
3.2 VR Interaction Framework (VRIF)	33
3.3 Hand Tracking framework (HT)	36
3.4 HPTK	37
3.5 Oculus Integration	38
4 Popis realizace vlastní implementace	39

4.1	Vytvoření jednotlivých scén.....	39
4.1.1	Rozšíření skriptu OVRGrabber z Oculus Integration	39
4.1.2	VRIF.....	41
4.1.3	HT framework.....	42
4.1.4	HPTK	43
4.1.5	Oculus Integration.....	45
4.2	Vytvoření jednotného prostředí	49
4.3	Měření.....	50
4.4	Hodnocení.....	52
4.4.1	HPTK	52
4.4.2	HT framework.....	53
4.4.3	Oculus Integration.....	53
4.4.4	Finální výběr	55
	Závěr.....	57
	Seznam použitých zdrojů	58

Seznam Obrázků

Obr. 1-1: Virtuální realita [26]	15
Obr. 1-2: VR pokus s myší [1]	15
Obr. 1-3: Obrazovka telefonu při použití pro VR [3].....	16
Obr. 1-4: Oculus Quest [3]	17
Obr. 1-5: Koncept 3DoF a 6DoF [4]	17
Obr. 1-6: Rozmístění LED diod [5].....	18
Obr. 1-7: Rozmístění stanic (majáků) [27].....	19
Obr. 1-8: SLAM [5].....	20
Obr. 1-9: VR ovladač [6].....	21
Obr. 1-10: Prostředí Unreal Engine [28].....	23
Obr. 1-11: Prostředí Unity	23
Obr. 2-1: Oculus Touch [13]	25
Obr. 2-2: Ovladač HTC Vive [13].....	26
Obr. 2-3: Ovladač PlayStation Move [13].....	26
Obr. 2-4: HaptX Gloves DK2 [17]	27
Obr. 2-5: Pohled čtyř monochromatických kamer [16].....	28
Obr. 2-6: Postup vytváření 3D modelu [16].....	28
Obr. 2-7: VR headset s Gemini od Varjo [19].....	29
Obr. 2-8: Leap Motion Controller [19].....	29
Obr. 2-9: Vznik ohniska [19].....	29
Obr. 2-10: Oculus Quest 2 [20]	30
Obr. 3-1: Cesta k Unity Registry	31
Obr. 3-2: XR plugin management pro počítače.....	31
Obr. 3-3: Nastavení Oculus	31
Obr. 3-4: Oculus Integration.....	32
Obr. 3-5: Build Settings.....	32
Obr. 3-6: VRIF Import	34
Obr. 3-7: VRIF nastavení	34
Obr. 3-8: VRIF Layers	35
Obr. 3-9: VRIF Layer Collision Matrix	35
Obr. 3-10: VRIF Pipeline	35
Obr. 3-11: PlayerController.....	36
Obr. 3-12: HT framework import.....	36
Obr. 3-13: HPTK import	37
Obr. 3-14: Nastavení Physics	37
Obr. 3-15: Změna kolizí s HPTK	37
Obr. 3-16: Vrstva HPTK	37
Obr. 3-17: Příklad vytvořené pózy	38
Obr. 4-1: Hierarchie před skriptem	39

Obr. 4-2: HandsManager	39
Obr. 4-3: InteractableToolsSDKDriver	39
Obr. 4-4: Hand Tracking Grabber skript	40
Obr. 4-5: Hand Tracking Grabber v unity	40
Obr. 4-6: OVR Grabbable	40
Obr. 4-7: VRIF Integrations	41
Obr. 4-8: VRIF Hand Tracking	41
Obr. 4-9: GameDirector.....	42
Obr. 4-10: Hierarchie prstů.....	42
Obr. 4-11: My Right Hand Tracking.....	42
Obr. 4-12: Příklad Collideru.....	42
Obr. 4-13: My Grabbable Object.....	43
Obr. 4-14: Nastavení HPTK	44
Obr. 4-15: Nastavení OVRSkeletonTracker.....	44
Obr. 4-16: HPTK hierarchie	44
Obr. 4-17: Změna HPTK.....	44
Obr. 4-18: Přiřazení Hand prefabů	45
Obr. 4-19: InputOVR po vypnutí vizuálů.....	45
Obr. 4-20: Přiřazení CameraRig reference	45
Obr. 4-21: Reference ruky u HandGrabInteractor.....	46
Obr. 4-22: Přidání HandGrabInteractor do Interactor Drivers	46
Obr. 4-23: Nastavení Hand Grab Interactable	46
Obr. 4-24: Nastavení Transformable	46
Obr. 4-25: Objekt se všemi komponenty.....	46
Obr. 4-26: hand Pose Recorder	47
Obr. 4-27: Hand Pose Recordable	47
Obr. 4-28: Hand Pose Recordable po nahrání póz	47
Obr. 4-29: Hierarchie po nahrání póz.....	47
Obr. 4-30: Příklad nahrané pózy	48
Obr. 4-31: Příklad nastavení pózy	48
Obr. 4-32: Nastavení Hand Grab Interactable 2	48
Obr. 4-33: Nastavení Transformable 2	48
Obr. 4-34: LeftHandSynthetic 2	49
Obr. 4-35: Hierarchie s HandSynthetic	49
Obr. 4-36: LefthandSynthetic 1	49
Obr. 4-37: Prostředí	49
Obr. 4-38: Základní poloha	50
Obr. 4-39: Poloha při vypadnutí.....	50
Obr. 4-40: Úchop kuličky HPTK	52
Obr. 4-41: Úchop láhve HPTK.....	52
Obr. 4-42: Úchop láhve HT 2.....	53

Obr. 4-43: Úchop láhve HT 1	53
Obr. 4-44: Úchop kuličky HT	53
Obr. 4-45: Úchop láhve póza IO	54
Obr. 4-46: Úchop kuličky bez pózy OI 2	54
Obr. 4-47: Úchop kuličky póza OI 1	54
Obr. 4-48: Úchop láhve bez pózy OI 2.....	54

Seznam tabulek

Tab. 4-1: Tabulka s naměřenými úhly ve stupních pro aplikaci s HPTK	51
Tab. 4-2: Tabulka s naměřenými úhly ve stupních pro aplikaci s HT frameworkem	51
Tab. 4-3: Tabulka s naměřenými úhly ve stupních pro aplikaci s Oculus Integration	52
Tab. 4-4: Tabulka s bodovým ohodnocením jednotlivých implementací	56

Použité zkratky a pojmy

VR – Virtuální realita

3D – 3 dimensional – trojrozměrný

2D – 2 dimensional – dvojrozměrný

HMD – Head mounted display – brýle pro virtuální realitu

DoF – Degrees of freedom – stupně volnosti

Skript – Program zapsaný v podobě kódu

Prefab – Předem vytvořený objekt s komponenty

Motion tracking – Sledování pohybu

Úvod

V moderní době dochází k rychlému rozvoji nových technologií. Mezi tyto technologie patří i virtuální realita známá pod zkratkou VR. Tento pojem je velice rozšířen, a proto je možné se s ním setkat v mnoha oblastech života. Jednoduchým příkladem je zábavní průmysl, kde pomocí VR headsetů je možné dosáhnout jedinečných zážitků jinak těžko dosažitelných z pohodlí domova. Na druhou stranu se VR může použít v lékařství nebo v konstrukci, kde je možné vytvořit 3D modely orgánů, součástí nebo i celých staveb bez drahé výroby fyzického modelu. Dalším typickým příkladem je simulátor létání pro výcvik pilotů v armádě.

Jak již bylo zmíněno, VR se rychle rozvíjí a s tím přichází i nové možnosti ovládání. V současné době se totiž pro virtuální realitu používají hlavně VR ovladače. To se ovšem postupně mění s příchodem technologie hand trackingu. Ten se zbavuje nutnosti používání ovladačů a místo toho do virtuálního prostředí přenáší přímo ruce uživatele. To poskytuje výrazně větší rozmanitost možných pohybů. Samozřejmě existuje více způsobů fungování takového hand trackingu. Některé z nich dokonce přichází se zajímavými funkcemi, které například přináší haptickou zpětnou vazbu i do hand trackingu.

Hand tracking má dále různé způsoby implementace do samotných aplikací. Hlavní rozdíly se dají najít ve fungování interakcí s předměty ve virtuálním prostředí, a to hlavně v úchopu konkrétního předmětu. Úchopy mohou být založeny na různých principech. Jedním způsobem může být jednoduchý skript, který při provedení nějakého úkonu přiřadí chtěný předmět pod objekt virtuálních rukou v hierarchii. Samozřejmě se každý takový skript může lišit a poskytovat různé možnosti customizace. Další způsob může být založen na fyzice rukou. Takové řešení je ovšem méně časté.

Cíle bakalářské práce jsou:

1. Popis technologie hand trackingu a možnosti ovládání
2. Analýza současného stavu
3. Analýza možností implementace
4. Popis realizace vlastní implementace

1 Popis technologie hand trackingu a možností ovládání ve VR

Aby bylo možné představit hand tracking technologii, je nejdříve nutné se seznámit s pojmem virtuální realita. K použití VR je dále potřeba brýlí pro virtuální realitu. Ty by ovšem nefungovaly bez systému motion trackingu, který je zde také představen. Další důležitou součástí jsou možnosti ovládání. To vše je však zbytečné bez nějakého využití, a proto jsou zde naznačeny možnosti uplatnění. Na závěr kapitoly jsou prozkoumány dva hlavní softwary na vytváření VR aplikací.

1.1 Virtuální realita

Virtuální realita je přesně to, co si člověk pod tímto pojmem představí. Jedná se o umělé prostředí, které se snaží o vytvoření iluze skutečného světa. Lidé se v tomto prostředí můžou pohybovat a interagovat s různými objekty. Toto prostředí může mít mnoho praktických využití.



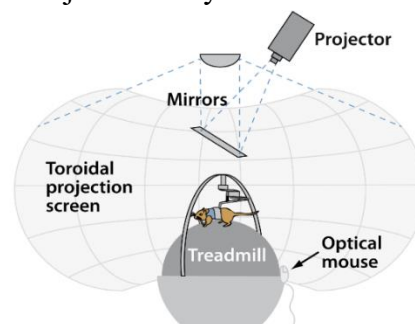
Obr. 1-1: Virtuální realita [26]

1.1.1 Definice VR

Definice VR je mnoho a každá z nich je trochu jiná. Zde je zvolena definice z knihy Virtual reality od Steven M. Lavallo: Jedná se o vyvolání cíleného chování v organismu pomocí umělé smyslové stimulace, během které má organismus žádné nebo minimální vědomí o skutečnosti. [1]

Tato definice se dá rozklíčovat pomocí 4 hlavních pojmů: [1]

1. Cílené chování: Organismus zažívá situaci (prožitek), která byla vytvořena uměle. Mezi tyto situace může patřit cokoli, např. chození, plavání, létání a jiné aktivity.
2. Organismus: V tomto případě se bude jednat o člověka (uživatele). Není to ovšem jediná možnost. V různých výzkumech bylo VR použito i na zvířatech. Jednoduchým příkladem je pokus z neurobiologie, ve kterém myš běhá po kouli, která se chová jako běžecský trenažér, a pomocí toho se pohybuje ve virtuálním bludišti.



Obr. 1-2: VR pokus s myší [1]

3. Umělá nervová stimulace: Pomocí technologie vytvářející VR prostředí je nahrazen jeden nebo více smyslových vjemů.
4. Vědomí: Organismus tento prožitek bere jako skutečný. Je takzvaně „ošálen“ a tím pádem je u něj vyvolán pocit přítomnosti v tomto umělém prostředí.

1.1.2 Komponenty VR

Pro začátek je důležité si představit součásti, které samotné VR vytváří. Existují dvě hlavní kategorie, Hardware a Software.

Hardware:

- Počítač nebo konzole
- VR headset neboli HMD (zobrazovací zařízení)
- Trackovací zařízení (senzory)
- Vstupní zařízení (ovladače, ruce)

Software:

- Programy pro 3D modelování
- Programy pro vytvoření simulace
- Softwary spojené s jednotlivými typy hardwaru (softwary na rozpoznání rukou nebo polohy v místnosti)

1.2 VR zobrazovací zařízení

Hlavním účelem VR je docílení reálného zážitku v umělé světě. To se dosahuje pomocí zobrazovacích zařízení nazývaných HMD nebo brýle pro virtuální realitu (VR headset). Jedná se o obrazovky, které si uživatel nasadí na hlavu. Tím pádem se s pohybem hlavy pohybují i obrazovky. VR headsety používají buď dva LCD displeje (jeden pro každé oko) nebo jen jeden, na který jsou posílány dva obrazy. Mezi oči a obrazovky jsou umístěny čočky, které slouží k zaostření obrazu pro každé oko. Ty vytvářejí stereoskopický 3D obraz natočením dvou 2D obrazů. Čočky vlastně napodobují způsob, jakým každé z lidských dvou očí vnímá svět trochu odlišně. VR brýle také potřebují dosahovat určitých parametrů jako jsou kvalita obrazu a počet snímků za vteřinu (FPS). Pokud by snímková frekvence nedosahovala hranice 60 FPS, uživatel by mohl při používání pociťovat dezorientaci a nevolnost. [2]

Existuje řada různých HMD. Každé má své výhody a nevýhody. Nejjednodušší verzí jsou headsety, které pro svou činnost nepotřebují víc než výkonný chytrý telefon. Pro použití je telefon se zapnutou aplikací vložen do příslušného slotu. Mezi takové brýle patří Google Daydream a Samsung Gear VR. [3]



Obr. 1-3: Obrazovka telefonu při použití pro VR [3]

Technicky vyspělejšími řešeními jsou VR headsety, které je nutné připojit k PC nebo konzoli s dostatečnou hardwarovou výbavou. Vzniklý obraz a obecně celý VR zážitek dosahuje vyšší kvality. Příkladem může být HTC Vive a Oculus Rift. [3]

Poslední verze VR headsetů mají tu výhodu, že pro jejich používání není potřeba připojení k externímu zařízení. Samotné brýle obsahují procesor, grafickou kartu, baterie, paměti atd. Mezi příklady patří Oculus Quest nebo HTC Vive Focus. [3]

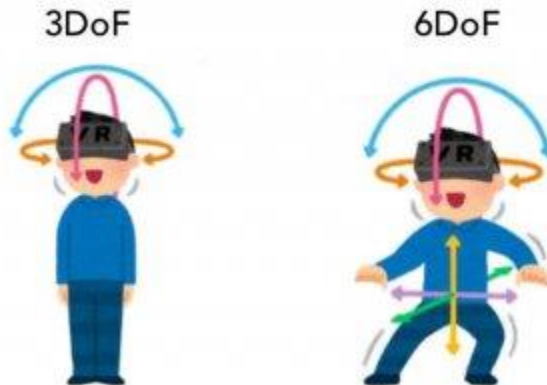


Obr. 1-4: Oculus Quest [3]

1.3 Motion tracking

Pro větší pocit přítomnosti ve VR vznikla řada systémů pro sledování pohybu, tzv. motion tracking. Motion tracking, který je důležitou součástí VR, je proces digitalizace pohybu pro použití v počítačových softwarech. Bez motion trackingu by byl uživatel ve VR omezen. Nemohl by se pohybovat ani rozhlížet.

Pro pohyb uživatele ve 3D prostředí je důležitý koncept 3DoF a 6DoF. Jednoduše řečeno uživatel může pohybovat hlavou rotací okolo tří os. To mu dává tři stupně volnosti (3Dof). Na tom následně staví 6DoF, které přidává pohyb po osách, nahoru a dolů, dopředu a dozadu, a doleva a doprava. Ve většině novějších VR headsetech se uživatel setká se šesti stupni volnosti. [4]



Obr. 1-5: Koncept 3DoF a 6DoF [4]

Pro dosažení přesného sledování pohybu byla vytvořena řada systémů. Každý trackovací systém přichází se svými výhodami a nevýhodami, a proto zde budou představeny tři typy. Tyto systémy je možné použít jak pro sledování HMD, tak pro sledování ovladačů.

1.3.1 Trackovací systém „Constellation“ (konstelace)

Tento systém byl vytvořen pro VR headset Oculus Rift. Při návrhu těchto brýlí byl Oculus postaven před výzvu. Snažil se totiž o dosažení stejné kvality trackování jako měly systémy za mnohem větší sumu peněz.[5]

Sledované zařízení je pokryto předem danou „konstelací“ infračervených LED diod, které jsou skryté pod vnějším plastem. Rozmístění je možné vidět na obrázku:



Obr. 1-6: Rozmístění LED diod [5]

Systém se skládá z řady senzorů, což jsou kamery s filtry, které vidí pouze infračervené světlo. Tyto senzory následně posílají snímky do počítače přes USB kabel s frekvencí 60 Hz. PC tyto snímky zpracuje, identifikuje polohu každé LED diody a tím i relativní polohu sledovaného zařízení. To je možné díky tomu, že software rozezná jednotlivé LED diody ze znalosti tvaru „konstelace“. Poté taky zná polohu diod z předchozího snímku, jejich směr zrychlení z akcelerometru, a rotaci z gyroskopu. Každá dioda navíc bliká svou specifickou frekvencí. Ta je pro lepší čtení LED diod při rychlém pohybu komunikována s čipem v senzorech. Díky tomu je zapínání senzorů se synchronizováno s probliknutím diod.[5]

Výhody systému: [5]

- Levná integrace
- Vysoká kvalita trackingu
- Funguje ve většině prostředí

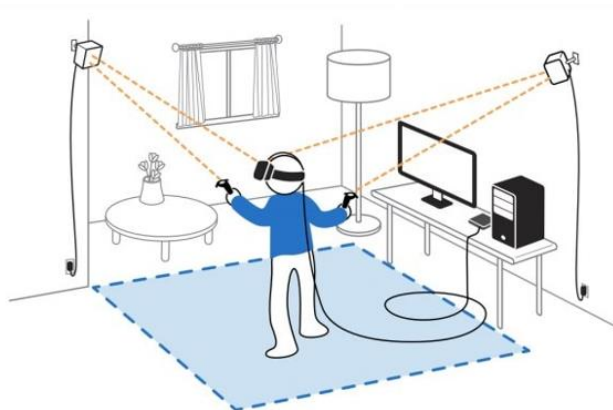
Nevýhody systému: [5]

- Každý senzor potřebuje kabelové připojení k PC
- Vysoká šířka pásma USB způsobuje problémy s řadou základních desek
- Sensory mají omezené vertikální zorné pole

1.3.2 Trackovací systém Lighthouse

Trackovací systém od Valve SteamVR se liší od své konkurence. Oproti většině systémů totiž nepoužívá kamery. Základem jsou dvě stanice (majáky), které umožňují sledování polohy v celé místnosti. [5]

Dvě stanice jsou umístěny do protilehlých horních rohů místnosti. Tyto stanice vyzařují širokoúhlé dvojrozměrné infračervené laserové paprsky přes celou místnost. To se provádí opakovaně vždy po jedné ose, tedy zleva doprava a pak shora dolů. Před každým pohybem je vyzářen silný infračervený záblesk světla. [5]



Obr. 1-7: Rozmístění stanic (majáků) [27]

Každé sledované zařízení obsahuje řadu infračervených fotodiód připojených k čipu. Čip slouží k měření času mezi infračerveným zábleskem a následným zasažením laseru v každé ose. Z toho pak určí svou polohu v místnosti. [5]

Výhody systému: [5]

- Není potřeba propojení s počítačem
- Vysoká kvalita trackingu
- Široký sledovaný objem

Nevýhody systému: [5]

- Relativně drahá výroba/integrace
- Vyžaduje montáž stanic na stěny (jinak dochází ke chvění)
- Reflexní povrchy způsobují chyby

1.3.3 Trackovací systém SLAM / Inside-Out

Tento systém se zaměřuje na odstranění nutnosti rozmístování externích senzorů a kamer v místnosti. HMD má v sobě zabudované kamery, které provádějí sledování „inside out (zvenitř ven)“ pomocí algoritmů počítačového vidění (computer vision). Jedním z těchto algoritmů je SLAM. To je zkratka pro Simultaneous Location And Mapping (Současné umístění a mapování). [5]

Algoritmus SLAM funguje na pozorování jedinečných statických prvků v místnosti. Následným porovnáním rotace a akcelerace, získaných z gyroskopu a akcelerátoru, k zdánlivému pohybu těchto prvků, lze určit polohu HMD. [5]



Obr. 1-8: SLAM [5]

Sledování ovladačů systému SLAM funguje podobně jako „Constellation“. Kamery umístěné na VR headsetu snímají LED diody, které vyzařují viditelné nebo infračervené světlo. Diody jsou skryté pod plastem ovladačů. [5]

Výhody systému: [5]

- Není potřeba externí hardware
- Nízká cena
- Jednoduchý setup

Nevýhody systému: [5]

- Nefunguje ve tmě
- Potíže se sledováním ovladače v určitých situacích

1.4 Ovládání VR

Výše byl uveden princip, jak je uživateli umožněn pohyb ve VR. Další důležitou součástí je forma ovládání. Uživateli musí být dovoleno interagovat s prostředím. Toho lze dosáhnout různými způsoby. Mezi ně patří VR ovladače a hand tracking. VR může uživatel ovládat i jinými méně rozšířenými metodami. Mezi ty patří například ovládání pomocí hlasu, které může být použito pro zajímavé rozšíření ostatních metod.

Momentálně nejběžnějším ovládacím prostředkem jsou ovladače. Slouží k registraci pohybů rukou a v některých případech i prstů. Tyto pohyby jsou zaznamenávány a přenášeny do virtuálního prostředí, kde na jejich základě vznikají určité interakce. Trackovací systémy jsou zmíněny v kapitole Motion tracking. Existuje mnoho typů ovladačů. Jednotlivé typy se liší v určitých vlastnostech, mezi které patří například preciznost trackingu, ergonomičnost, počet a typ tlačítek. K sadě tlačítek patří i trigger (spoušť) a páčka pro palec. Typickým příkladem využití ovládacích prvků je stisknutí triggeru pro zmáčknutí spouště virtuálního nástroje nebo použití páčky pro pohyb ve VR.[6]



Obr. 1-9: VR ovladač [6]

Hand tracking technologie je dalším vývojem ovládání ve VR. Jednoduše řečeno odpadá nutnost používání ovladačů. Ovladače jsou totiž nahrazeny přímo rukama uživatele. Ty jsou snímány pomocí kamer a přenášeny do virtuálního prostředí. Hand tracking má velký potenciál pro budoucnost VR a jejího ovládání. Je jen otázkou času, než nahradí ovladače ve většině použití.

Tato technologie s sebou totiž přináší mnoho výhod. Hlavní výhodou je, že ruce má člověk vždy u sebe. Nejen to, ale člověk je schopen pomocí vlastních rukou provádět delikátnější operace. Tyto operace vlastně nemají jinou limitaci, než je reálný pohyb rukou. Uživatel je na rozdíl od ovladačů schopný pohybovat každým prstem zvlášť. To vše vede k většímu pocitu imerze ve virtuálním prostředí.

Na druhou stranu zde vznikají jiné komplikace. Každý člověk je jiný. Liší se i jeho ruce, jejich velikost, barva i povrch. Navíc se uživatelé mohou vyskytovat v jiném prostředí. Z těchto důvodů vznikají různé obtíže při zaznamenávání pohybu rukou a následném vytváření jejich virtuálních dvojčat. Jednoduchým příkladem může být problém s trackováním rychlých pohybů. Dále může vznikat menší prodleva před prvotním načtením rukou a následným načítáním při vstoupení do zorného pole kamer. Sledování pohybu se také stává obtížné, pokud jsou ruce v zákrytu nebo

za překážkou. I proto je nutné provádět řadu měření a testů v různých prostředích a s různými lidmi pro dosažení co nejlepších výsledků.

V obecném případě se technologie hand trackingu dá zjednodušit na nějaký senzor, obvykle speciální typ kamery přímo v HMD, který zaznamenává obraz. Tento obraz je následně zpracován nějakým softwarem a podle získaných dat je vytvořen jednoduchý model rukou. Ten se většinou skládá z bodů, které reprezentují klouby a konečky prstů. Na ty je pak „navlečen“ 3D model. Celý proces se opakuje a tím vznikají pohyblivá virtuální dvojčata rukou uživatele, které může použít na interakci s objekty ve VR.

1.5 Využití VR a hand trackingu

S pojmem virtuální realita má většina lidí spojený zábavní průmysl. Ten ovšem není jediným možným využitím. VR se dá použít i v ostatních průmyslech. Příkladem může být zobrazení 3D modelu součástky, tréninkové aplikace, výzkum ergonomie pracoviště a další.

K tomu všemu lze také využít běžné VR ovladače. Problémem je to, že se držení ovladačů nepodobá realitě a uživatel je omezený jen na malé množství interakcí. Využití virtuální reality v průmyslu navíc bude pro většinu pracovníků prvním setkáním s VR, a proto je důležitá intuitivnost ovládání. S tou přichází právě hand tracking. Uživatel, jak již bylo zmíněno, používá k interakcím vlastní ruce. Díky tomu jsou pro něj pohyby ve VR více přirozené a je pro něj snadnější si zapamatovat dané úkony. [7]

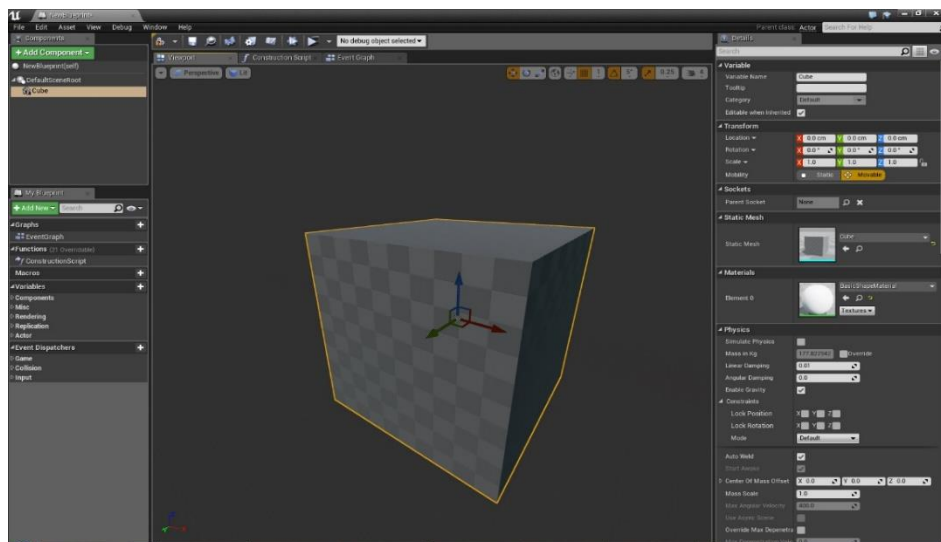
Prvním příkladem je zobrazování 3D modelů. Uživatel může s daným předmětem interagovat a prohlédnout si ho ve skutečné velikosti. To vše je možné bez výroby fyzického prototypu. Druhým příkladem jsou tréninkové aplikace. Nový pracovník nemusí být kvůli zaučení přítomný přímo na pracovišti. Místo toho je s prací seznámen ve virtuální kopii pracoviště. Díky tomu se nebude zpomalovat výroba, zmenší se počet potenciálních zmetků a na pracovníka bude působit celkově méně stresu. Díky hand trackingu se oproti ovladačům dále zkrátí potřebná doba na zaučení. Uživatel totiž bude vykonávat přímo daný pohyb prstů. Nebude pouze mačkat tlačítko na ovladači. Posledním příkladem je ergonomie pracovišť. Díky hand trackingu je možné provést přesnou ergonomickou studii pracovišť. [7]

Samozřejmě existuje spousta dalších využití, která mohou být specifická pro jednotlivá průmyslová odvětví.

1.6 Softwary

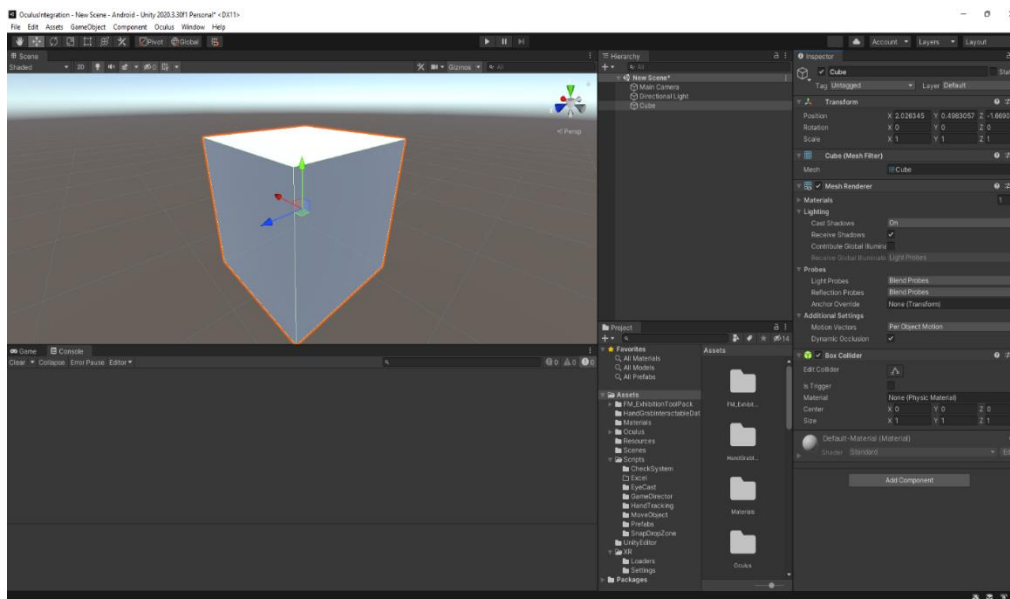
Všechny ovládací prvky jsou zbytečné, pokud není vytvořené virtuální prostředí. K tomu slouží softwary. Vzhledem k velké podobnosti mezi vývojem her a vývojem VR byla řada takzvaných herních enginů rychle adoptována pro použití ve VR. Hlavními představiteli jsou Unreal Engine a Unity. Dále lze použít Godot, Amazon Sumerian a další.

Prvním příkladem je Unreal Engine. Jedná se o populární herní engine vyvinutý společností Epic Games, který umožňuje vývoj napříč různými platformami od PC po konzole. Je dostupný pro zkušené programátory, kteří znají programovací jazyk C++, ale i pro amatéry, kteří mohou použít jejich předem vytvořené bloky kódu. Poskytuje také materiálové a animační nástroje, které umožňují rychlou tvorbu složitých scén.[8]



Obr. 1-10: Prostředí Unreal Engine [28]

Druhým příkladem je Unity. Jedná se o 2D a 3D herní engine, který existuje od roku 2005, od společnosti Unity Technologies. Používá programovací jazyk C#. Byl vytvořen s cílem poskytnout více vývojářům přístup k nástrojům pro vývoj her. V průběhu své životnosti se dramaticky změnil a rozšířil na řadu dalších platform, jako je např. PC, mobil, konzole a VR. I v dnešní době je hlavním cílem herního engine poskytnout co nejširší sadu nástrojů a zároveň co nejvíce usnadnit používání vývojářům všech úrovní. [9]



Obr. 1-11: Prostředí Unity

- Výběr softwaru [10] [11] [12]

Unity a Unreal Engine patří mezi špičku ve vývoji pro VR. Pro další postup je ale nutné vybrat vhodnější variantu pro průmyslové použití. Proto bylo potřeba porovnat oba softwary. To bylo provedeno v několika kategoriích. Vybraný software je použit v pozdějších kapitolách. U obou softwarů bylo provedeno testování na krátkém příkladu pro seznámení s prostředím.

První kategorií je obchod s assety/modely. Každé virtuální prostředí potřebuje objekty. Pro uživatele může být vytváření vlastních modelů překážkou, a proto oba softwary nabízejí obchod s 3D modely. Unreal knihovna se zaměřuje více na hry. Unity knihovna je ovšem větší a díky tomu poskytuje modely pro řadu průmyslových aplikací.

Dále jsou prozkoumány dokumentace a nabízené kurzy. Na stránkách obou softwarů jsou dostupné rozsáhlé knihovny, které obsahují informace o vývoji ve virtuální realitě, stejně tak i řadu různých kurzů. Tyto kurzy slouží pro představení prostředí obou aplikací a naučení základů práce s nimi. Unity ovšem nabízí těchto kurzů více a je obecně přístupnější pro nové vývojáře.

Třetí kategorií jsou programovací jazyky. Hlavním programovacím jazykem Unity je C#. Je tu ale možnost použít i jiné jazyky. V Unreal Engineu je používán jazyk C++ společně s předem vytvořenými bloky kódu. Zde záleží pouze na preferenci vývojáře.

Jako čtvrté byly prozkoumány hardwarové nároky. Oba softwary mají velice nízké minimální hardwarové požadavky. Nevyžadují více než operační systém Windows 7 a grafickou kartu s DirectX 10. Doporučený hardware se skládá z počítače s Windows 10 64-bit, čtyřjádrový procesor Intel nebo AMD, 8 GB RAM a výkonnější grafickou kartu s DirectX 11/12.

Poslední kategorií je cena. Unreal Engine je dostupný zdarma. Jedinou podmínkou je 5% provize ze všech produktů, které vytvoří zisk nad 3000 dolarů. Unity je dostupné zdarma pro studenty a osobní užití. Firmy si pak mohou vybrat z různých variací měsíčního a ročního předplatného.

Z porovnání je zřejmé, že Unity je vhodnější pro vývoj aplikací použitelných v průmyslu. To hlavně díky své větší knihovně modelů. Další výhodou je vyšší přístupnost novějším vývojářům. Případné zaučení by tedy probíhalo rychleji. Pro postup v této práci je použit právě software Unity.

2 Analýza současného stavu

V současné době existuje řada možností, jak VR ovládat. Proto zde jsou uvedeny příklady nepoužívanějších ovladačů, tak jako nejnovější technologie hand trackingu. Na závěr je nutno vybrat VR brýle, které jsou použity pro další postup.

2.1 Ovladače

Z ovladačů budou představeny 3 typy, které patří mezi špičku ve svém oboru. U ovladačů se hodnotí pohodlnost úchopu (ergonomičnost), tlačítka, výdrž baterie, a hlavně přesnost a spolehlivost sledování pohybu.

2.1.1 Oculus Touch

Oculus Touch jsou ovladače dodávané s VR headsetem od firmy Oculus VR. Nejnovější verze je dodávána k VR headsetu Oculus Quest 2.

Jedná se o pravý a levý ovladač vytvořený z odolného plastu, který je schopný odolat řadě nechtěných nárazů. I když na první pohled působí poměrně robustně, hlavně díky sledovacímu prstenci, je v ruce překvapivě lehký a pohodlný. Rozložení tlačítek je intuitivní. Na každém ovladači je možné najít 3 tlačítka společně s páčkou, trigger a tlačítko na rukojeti. Samotné trackování pohybu je přesné a citlivé. Funguje na principu konstelace LED diod, které jsou rozmístěny po ovladači, a snímány pomocí senzorů. V ovladači jsou dále k nalezení tzv. kapacitní senzory, které snímají polohu prstů. To umožňuje uživateli provádět jednoduchá gesta. Zážitku přidávají dobře zpracované vibrace, které ještě vylepšují pocit imerze. Ovladače běží na AA baterie, což přináší poměrně dlouho životnost. [13] [14]

Výhody:

- Pohodlnost
- Výborné trackování
- Zpětná vazba (vibrace)

Nevýhody:

- Běží na AA bateriích



Obr. 2-1: Oculus Touch [13]

2.1.2 Ovladač HTC Vive

I přesto, že ovladače pro VR headset HTC Vive, jsou na trhu již několik let, patří mezi špičku ve svém oboru.

HTC Vive ovladače jsou specifické svým multifunkčním trackpadem. Ten lze použít jako páčku, kolečko a tlačítko v různých situacích. Vedle toho má ovladač tlačítko na rukojeti a trigger. Rozložení tlačítek a celkový tvar ovladače je pohodlný. Ovladač je ale o trochu těžší než jiné ovladače. Samotný tracking je dobře udělaný a dá se považovat za hlavní výhodu. Jedná se o princip trackování pomocí dvou stanic. Na ovladači proto musí být několik senzorů, které sledují relativní pozici ovladače vůči daným stanicím. Baterie je vestavěná. [13] [15]

Výhody:

- Multifunkční trackpad
- Pohodlnost
- Přesné trackování

Nevýhody:

- Vyšší váha



Obr. 2-2: Ovladač HTC Vive [13]

2.1.3 Ovladač PlayStation Move

Ovladač přímo dělaný pro PlayStation je jediná dobrá možnost pro VR na konzolích.

Ovladače jsou typické svými barevnými senzory ve tvaru koule. Ty zajišťují, že sledování pohybu je přesné. Součástí je i zpětná vazba v podobě vibrací. Tlačítka svým vzhledem připomínají klasický PlayStation ovladač. Jedná se o kombinaci čtyř tlačítek okolo centrálního tlačítka pro pohyb a triggeru. Celkový design je poměrně jednoduchý. Přesto ale poskytuje ergonomický a pohodlný úchop. Nevýhodou je, že ovladač je použitelný pouze s PlayStationem, ke kterému je nutno dokoupit kameru. [13]

Výhody:

- Pohodlnost
- Jednoduchost
- Skvělé pro PlayStation VR

Nevýhody:

- Použitelné pouze s PlayStationem 4 nebo 5 společně s kamerou



Obr. 2-3: Ovladač PlayStation Move [13]

2.2 Hand tracking

Pro hand tracking byly také vybrány 3 typy produktů. Není zde potřeba hodnotit ergonomičnost a tlačítka, protože zde nic v ruce uživatel nadržuje. Naopak je nastíněno fungování jednotlivých systémů. Hand tracking je u větší části headsetů pouze jako dodatečná funkce, proto je každý headset dodáván i s vlastními ovladači.

2.2.1 Rukavice s markery

Rukavice s markery jsou takový mezistupeň mezi ovladači a hand trackingem. Stále je zde potřeba použití přídavného zařízení, tentokrát to je pár rukavic. Ty jsou posázeny markery. Markery jsou značky, mohou být v mnoha formách, pomocí kterých se zachycuje poloha rukavic. Slouží jako referenční body. Tyto body jsou následně zaznamenávány pomocí kamer. Příkladem může být kopule z několika kamer, která je postavena kolem uživatele, pro co největší přesnost.[16]

Pro zajímavost je možné uvést speciální případ takových rukavic. Jedná se o haptické rukavice, např. HaptX Gloves DK2. Tyto rukavice simulují dotek virtuálního předmětu, tak jako by to byl objekt reálný. Toho dosahuje pomocí desítek bodů hmatové odezvy.[17]



Obr. 2-4: HaptX Gloves DK2 [17]

Výhody:

- Vysoká kvalita trackingu
- Vysoká rozmanitost možných pohybů

Nevýhody:

- Stále je potřebný externí hardware
- V příkladu s kopulí – špatná přenosnost

2.2.2 Hand tracking v Oculus Quest

Systém hand trackingu použitý v Oculus Quest využívá čtyř již zabudovaných monochromatických kamer v HMD. Tyto kamery mají řadu výhod, a to menší velikost, nižší váhu, široké zorné pole, nízkou energetickou náročnost a nižší cenu. Je zde ale větší problém s počítačovým viděním (computer vision). [16]



Obr. 2-5: Pohled čtyř monochromatických kamer [16]

Ze čtyř 2D obrázků se musí vytvořit 3D virtuální dvojče rukou uživatele. Nejdříve je nutné ruce najít. Toho se docílí pomocí procesu známém jako deep learning (hluboké učení). To se využívá například u rozpoznávání obličejů na mobilních fotoaparátech. Jednoduše řečeno se software musí naučit rozeznávat ruce od okolí. Proto se softwaru ukáže tisíce a tisíce příkladů rukou v různých prostředích, dokud se je nenaučí odlišit rychlým a spolehlivým způsobem. [18]

Poté co jsou ruce nalezené, software obrázky zanalyzuje a odhadne umístění určitých klíčových bodů, opět pomocí deep learningu. Za klíčové body se považují klouby a koncečky prstů. Klíčové body se porovnají s předešlou pózou ruky a vytvoří se aktuální odhad pozice ruky. Ten se následně nahraje do uživatelské aplikace a přiřadí se k němu 3D model, který se zobrazí ve VR. [16]



Obr. 2-6: Postup vytváření 3D modelu [16]

Výhody:

- Není potřeba externí hardware
- Jednoduchá integrace – kamery už v HMD byly
- Vysoká rozmanitost možných pohybů

Nevýhody:

- Problémy s trackováním rychlých pohybů

2.2.3 Ultraleap hand tracking

Ultraleap vzniklo kombinací firem Leap Motion a Ultrahaptics. Přichází se svou verzí hand trackingu, která je založena na dvou kamerách a infračervených LED diodách.

LED diody osvětlí ruce uživatele pomocí infračerveného světla viditelného pouze zmíněnými kamerami. Blikání diod je synchronizováno se snímkovou frekvencí kamer. Sesbíraná data jsou následně posílána do počítače, kde z nich jsou za pomoci softwaru vytvořeny 3D dvojčata rukou. Ultraleap se pyšní svými modely, které modelují nejen dlaně a prsty, ale i klouby a kosti. Díky tomu dokážou přesně odhadnout polohu prstů i když jsou skryté za překážkou. [19]

Tuto technologii využívají v řadě produktů. Jedním z nich je Leap Motion Controller. Jedná se o optický modul, který lze zapojit do počítače. Dalším příkladem je VR headset vytvořený ve spolupráci se společností Varjo. Ten kombinuje HMD s Gemini, což je nejnovější verze Ultraleap hand trackingu. [19]



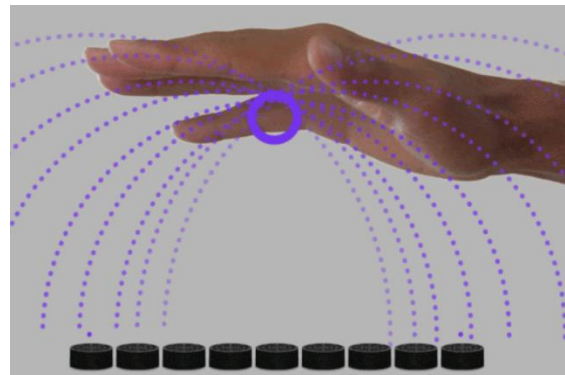
Obr. 2-8: Leap Motion Controller [19]



Obr. 2-7: VR headset s Gemini od Varjo [19]

Ultraleap také přichází s technologií, která přináší do hand trackingu hmatový vjem. Jednoduše řečeno přetváří ultrazvuk na virtuální dotyk. K tomu se používá několik malých reproduktorů vysílajících ultrazvuk, který je neslyšitelný pro lidské ucho. Tyto reproduktory jsou, za pomoci algoritmu, spouštěny s velmi specifickými časovými rozdíly. Ty způsobí, že ultrazvukové vlny dorazí do stejného bodu v prostoru ve stejnou dobu. Tento bod se nazývá ohnisko. V ohnisku vzniká tlakový bod, který je uživatel schopen cítit. Pak už jen záleží na rozmístění jednotlivých ohnisek, kterým se dosáhne požadovaného vjemu. [19]

Výhody a nevýhody jsou prakticky stejné jako u předchozího typu. Jediný rozdíl je možnost použití externího hardwaru pro dosažení dalších funkcí



Obr. 2-9: Vznik ohniska [19]

2.3 Výběr VR brýlí

Existuje několik možností, které umožňují provoz hand trackingu ve VR. Pro účely této práce byly prozkoumány tři možnosti. Těmi jsou VR headset Oculus Quest 2, VR headset Vive Focus 3 a řešení od společnosti Ultraleap.

- Oculus Quest 2

Jedná se o brýle pro virtuální realitu, které pro své fungování nepotřebují připojení k počítači. Headset je určen pro použití běžným uživatelem. Hand tracking funguje na principu, který byl představen v předchozí části. Samotné fungování trackingu je velmi dobré a patří mezi lepší ve svém oboru. Cena tohoto produktu se pohybuje mezi 10 000 Kč až 13 000 Kč v závislosti na velikosti uložení. [20]

- Vive Focus 3

Opět se jedná o samostatně fungující VR headset, který je více zaměřen na profesionální použití. Tyto brýle dosahují lepších zobrazovacích parametrů. To se také projevuje na ceně, která se pohybuje kolem 30 000 Kč. Hand tracking funguje na podobném principu jako předchozí možnost. Vive ale funkci hand trackingu nezvládl stejně dobře jako jeho konkurence. [20]

- Ultraleap

Prvním řešením od společnosti Ultraleap je zařízení Leap Motion controller, které se používá jako rozšíření pro implementaci hand trackingu. Dá se proto použít v kombinaci s různými VR headsety. Hand tracking funguje na principu infračervených LED diod. Dosahuje srovnatelných, nebo dokonce lepších výsledků než hand tracking od Oculusu. Samotné zařízení stojí okolo 3 000 Kč. Je ovšem nutné dokoupit samostatné brýle pro VR, což většinou posune celkovou cenu přes cenu Oculus Questu 2. Druhým řešením je pak VR headset Pico Neo 3. Tyto brýle mají vytvořený doplněk přímo od firmy Ultraleap a díky tomu se jedná o hlavního konkurenta Oculus. Společně s tímto doplňkem brýle stojí okolo 23 000 Kč. [19]

Po zvážení těchto možností byl pro účely této práce vybrán Oculus Quest 2. Jedná se o nejlepší možnost, co se týče poměru cena / výkon. Podobného výsledku dosahuje i Leap Motion Controller a Pico Neo 3. Oculus Quest 2 byl nakonec vybrán z důvodu jednoduchosti použití. Není potřeba připojování dalšího externího zařízení. Dále se jedná o nejlevnější variantu.



Obr. 2-10: Oculus Quest 2 [20]

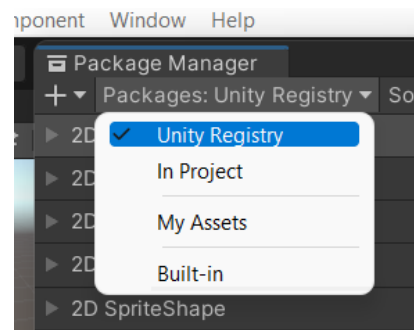
3 Analýza možností implementace

V této kapitole je prozkoumáno pět možností implementace hand trackingu a uchopení předmětů s jeho využitím. Všechny možnosti jsou provedeny ve vybraném programu Unity verze 2020.3.30f1 s rozšířením pro Android. Implementace jsou připravovány pro použití na zvoleném headsetu Oculus Quest 2. Proto jsou zvoleny assety podporující platformu Oculus. Pro každý projekt bude provedena stejná základní příprava potřebných assetů a celkové nastavení unity pro virtuální realitu.

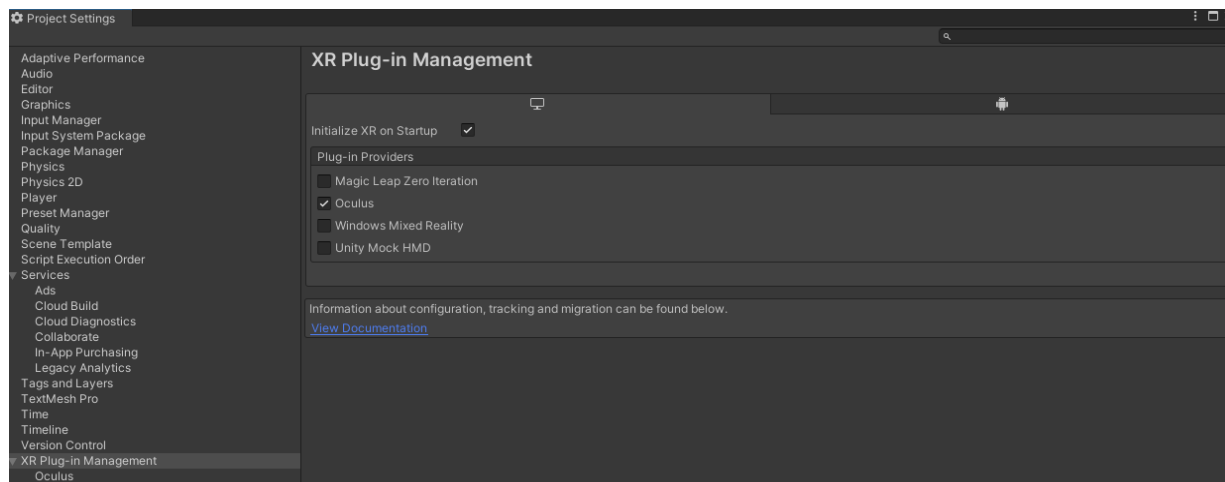
- Příprava Unity

Jako první jsou nainstalovány dvě základní rozšíření pro virtuální realitu. Jedná se o XR Plugin Management a Oculus XR Plugin. Ty jsou k nalezení ve Window > Package Manager > Unity Registry.

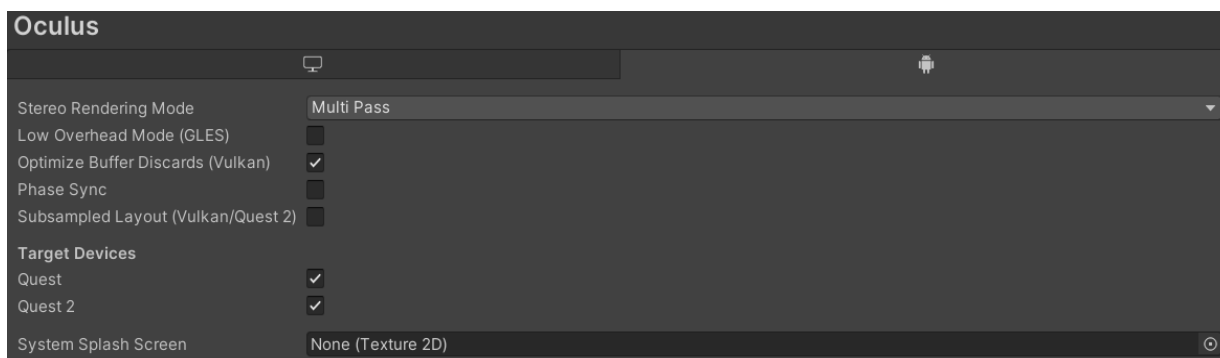
Po dokončení instalace je provedeno nastavení v Edit > Project settings > XR Plug-in Management. Zde je nutné zaškrtnout políčko vedle Oculus. To samé je provedeno na záložce Android, která je důležitá pro další postup.



Obr. 3-1: Cesta k Unity Registry



Obr. 3-2: XR plugin management pro počítače



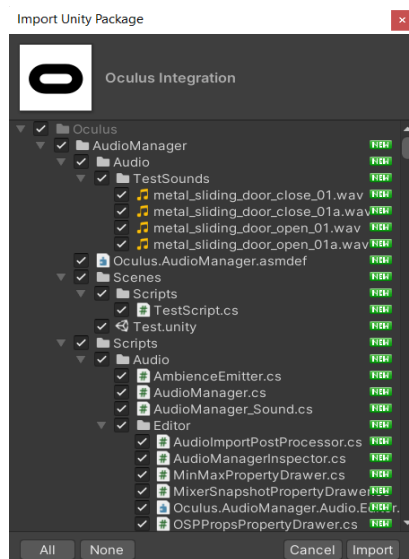
Obr. 3-3: Nastavení Oculus

Dále je zkontrolováno zaškrtnutí políčka Quest 2 na Edit > Project settings > Oculus > Android.

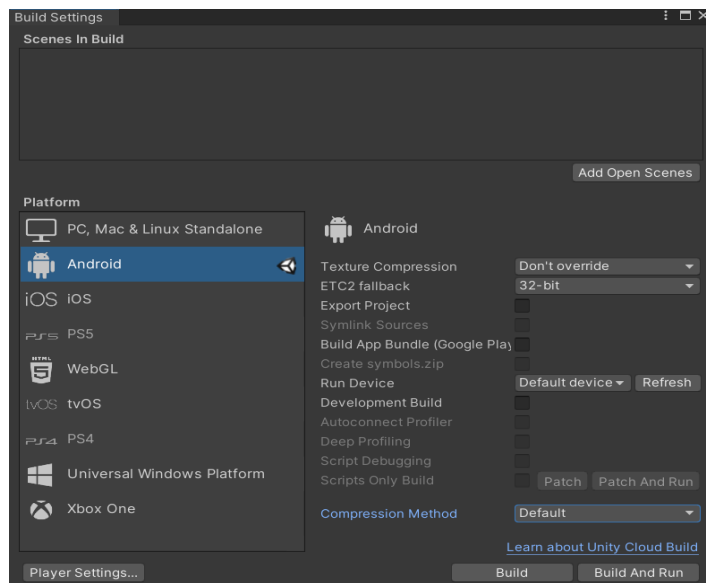
Tři z pěti řešení využívají prefabu OVRCameraRig. Ten je součástí assetu Oculus Integration, které je možné zadarmo získat na Unity Asset Store. Po přidání do unity je nutné asset nainportovat ve Window > Package Manager > My Assets. Po dokončení importování stačí potvrdit všechny vyskakovací okna (možnosti vlevo). Unity je poté restartováno.

Nyní je možné do scény předpřipravit prefab OVRCameraRig, který lze vyhledat v podokně Project. Tento prefab představuje novou kameru, a proto je odstraněn objekt Main Camera, která je součástí základní scény. V OVRCameraRig byl změněn Tracking Origin Type na Floor Level. Tím se zajistí, že výška kamery se bude odvíjet od vzdálenosti očí od země. K tomu je změněn Hand Tracking Support na Hands only, který je k nalezení pod Quest Features > General.

Na závěr je potřeba změnit platformu projektu na Android. Toho lze docílit otevřením File > Build Settings, zvolením možnosti Android a potvrzením tlačítkem Switch Platform. Ve stejném okně je změněno Compression Method na Default. Z tohoto místa budou také pomocí tlačítka Build vytvářeny aplikace, které budou následně nahrávány na headset pomocí aplikace SideQuest. Před spuštěním tvorby aplikace je důležité zvolit scénu, která bude součástí. Dále je nutné zkontrolovat, zda v adresáři umístění projektu, názvu aplikace a názvu projektu není diakritika a další speciální znaky. Ty mohou způsobovat errorry při vytváření aplikace. Dalším errorem při vytváření aplikace může být špatně nastavené renderování světla. Proto je nutné změnit Color Space na Linear v Project Settings > Player > Other Settings.



Obr. 3-4: Oculus Integration



Obr. 3-5: Build Settings

Tímto je unity připraveno na jednotlivé implementace hand trackingu.

3.1 HurricaneVR – Rozšíření skriptu

HurricaneVR je kompletní VR interakční framework s velkým důrazem na fyzikální interakce od firmy Cloudwalkin Games. Nabízí řešení mnoha problémů spoutaných s virtuální realitou. To ovšem pouze s využitím VR ovladačů. Proto by pro zkušeného programátora mohlo být lákavou možností implementace hand trackingu pomocí vlastního skriptu, nebo případným rozšířením již existujícího skriptu. [21]

Tato metoda je jednou ze složitějších možností, jak k problému přistoupit, ale na druhou stranu je možné docílit všeho, čeho by uživatel vyžadoval. K tomu je však potřebná určitá znalost programovacího jazyka C# a samozřejmě tento proces zabere více času než ostatní možnosti.

V této práci bude vytvořena jen malá ukázka postupu, jelikož jsou v dnešní době k dispozici snazší a rychlejší metody, jak docílit podobného výsledku. Je ovšem důležité tuto možnost zmínit, jelikož se jedná o základní způsob dosažení chtěného cíle. Je to také způsob, jakým byly vytvořeny všechny ostatní možnosti implementace. Tato metoda by mohla být zvolena v případě, pokud by uživatel chtěl docílit něčeho, co by jeho dosavadní řešení nenabízelo. Příkladem může být právě absence hand trackingu v HurricaneVR.

HurricaneVR podporuje všechny hlavní VR headsety, XR pluginy, verze Unity 2019 a novější, což umožňuje využití na mnoha zařízeních a platformách. Nejnovější verze 2.8.4.3 je momentálně dostupná za 79.99 dolarů na Unity Asset Store. [21]

3.2 VR Interaction Framework (VRIF)

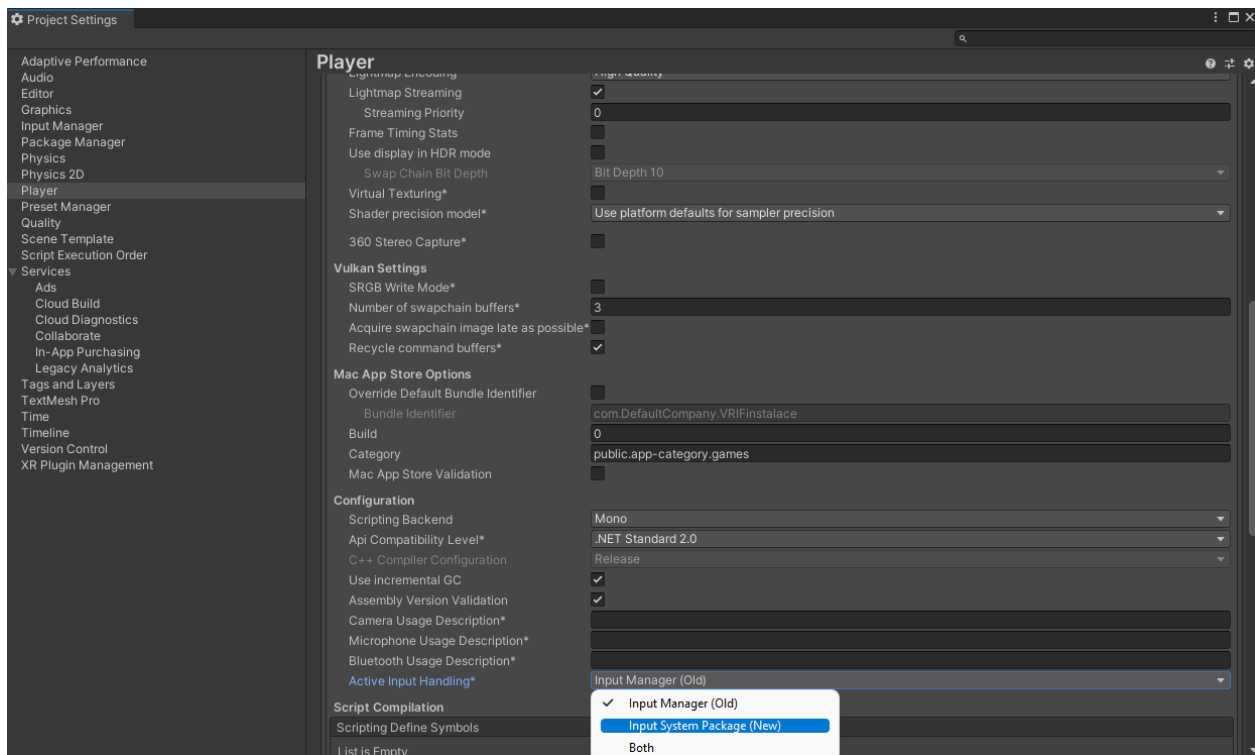
VR Interaction Framework je dalším vyzkoušeným VR frameworkem tentokrát od firmy Bearded Ninja Games. Jedná se o sbírku skriptů a prefabrikátů, které umožňují vývoj interakcí ve VR. Cílem je usnadnění vytváření vlastních interaktivních objektů a rychlé zvýšení produktivity. K dispozici je vysoká rozmanitost prefabrikátů běžných interakcí, od jednoduchých bloků po různé spínače, páky a zbraně. [22]

VR Interaction Framework podporuje všechny hlavní VR headsety kompatibilní s Unity, jako jsou Oculus Quest, Quest 2, HTC Vive, Valve Index atd. Nejnovější verze 1.73 je momentálně dostupná za 59.99 dolarů na Unity Asset Store. [23]

VRIF se velmi podobá HurricaneVR, rozdílem je to, že poskytuje vlastní formu implementace hand trackingu do VR. Dokonce obsahuje vlastní demo scénu, kde si lze vyzkoušet brání předmětů a 3D kreslení s využitím vlastních rukou. Hand tracking je ovšem pouze vedlejší produkt, a proto nedosahuje podobné kvality jako ostatní části assetu.

- Import VRIF

Samotné implementování VR Interaction Frameworku do Unity je jednoduchý proces. Po prvním zapnutí Unity je potřeba povolit nový Unity Input System. Změnu lze provést v nastavení projektu, konkrétně Edit > Project setting > Player > Other Setting > Active Input Handling > Input System Package (New). Pro změnu je potřeba restartovat Unity.

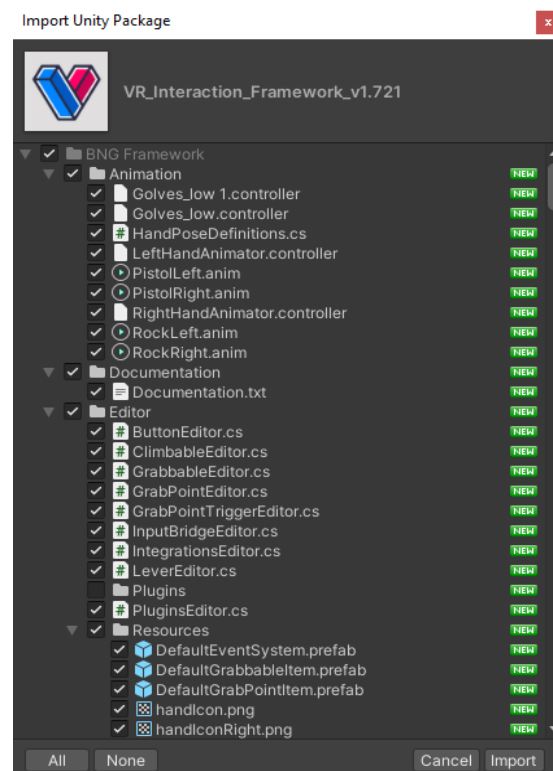


Obr. 3-7: VRIF nastavení

Poté už lze přistoupit k naimportování samotného assetu. To je možné provést více způsoby. V tomto případě je VRIF stažený externě a připravený ve složce. Pro importování se postupuje Assets > Import Package > Custom Package. V průzkumníku souborů se vybere příslušný soubor typu Unity Package File. Ve vyskakovacím okně je zvoleno Import a následně Install/Upgrade. To zajistí, že se nainstalují i všechny potřebné balíčky. Po dokončení načítání se objeví další okno.

Na konci seznamu jsou k nalezení změny nastavení projektu. Pokud by se provádělo importování, do již existujícího projektu, je vhodné zvážit přeskočení některých změn nastavení. V tomto případě se jedná o prázdný projekt, a proto není třeba nic měnit. Dále se potvrdí importování stiskem tlačítka.

Nyní lze provést kroky instalace a nastavení XR Management Pluginu a Oculus XR Pluginu.

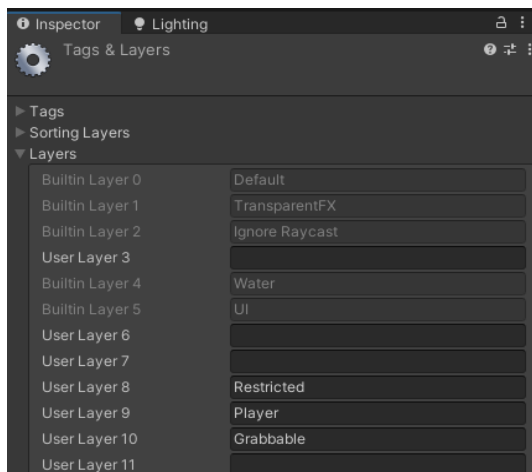


Obr. 3-6: VRIF Import

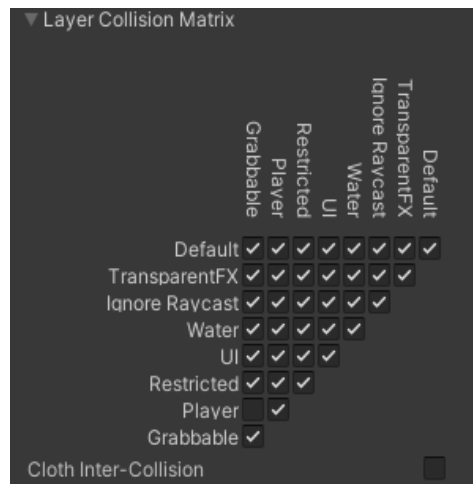
- Nastavení v již existujícím projektu [22]

V případě importování frameworku, do již existujícího projektu, je doporučeno provést následující nastavení. Nastavení vrstvy (layer) Player a Grabbable, příslušným objektům v hierarchii, umožní ignorovat kolize mezi interaktivními předměty a kolizním prvkem kapsle hráče. Vrstvu Restricted lze použít s komponentou PlayerTeleport k odfiltrování objektů, na které by se nemělo jít teleportovat.

Interakce mezi vrstvami se může upravit v matici kolizí, Edit > Project Settings > Physics.



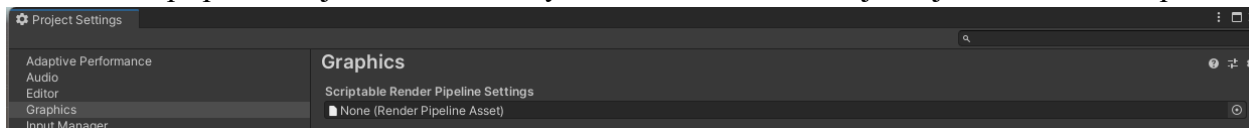
Obr. 3-8: VRIF Layers



Obr. 3-9: VRIF Layer Collision Matrix

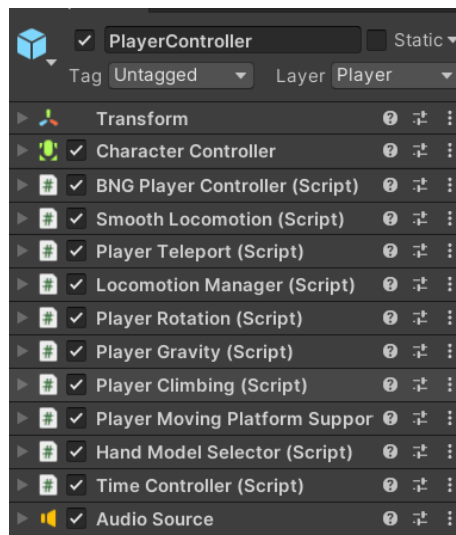
VRIF je považován za Project Template (šablonu projektu) a může přepsat nastavení projektu. To může potenciálně způsobit padání projektu, který používá Universal Render Pipeline. Pro opravu je třeba změnit pipeline na Edit > Project Settings > Graphics > Scriptable Render Pipeline Settings.

Dále je důležité nastavení MainCamera tagu na transformu, který má být sledován jako hlava hráče. Ve většině případů se jedná o CenterEyeAnchor. Toto řešení jako jedno z mála nepoužívá



Obr. 3-10: VRIF Pipeline

OVRCameraRig z Oculus Integration. Místo toho používá vlastní prefaby Rigů, které představují hráče ve scéně. Rig má podřízený objekt PlayerController, který má řadu vlastních skriptů. Ty umožňují různá nastavení hráče. BNG Player Controller slouží k základnímu nastavení hráče, kam patří například pohyb kamery s hráčem nebo také nastavení výšky hráče. Dále například Smooth Locomotion umožňuje nastavení základních pohybů jako je běh nebo skákání. Ostatní skripty dovolují nastavení teleportace, gravitace a dalších možností.



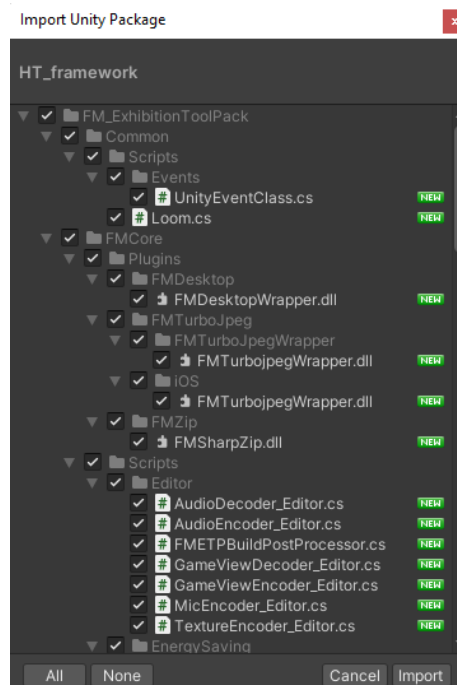
Obr. 3-11: PlayerController

Pod PlayerControllerem jsou další podřazené objekty, mezi které patří i objekty obou rukou.

3.3 Hand Tracking framework (HT)

Tento framework je stále ve vývoji a momentálně není dostupný pro širokou veřejnost. Byl proto poskytnut pouze pro ukázkou další možnosti implementace hand trackingu ve VR. Jeho výhodou je, že je vyvíjen přímo pro implementaci hand trackingu.

Framework byl poskytnut formou souboru. Z toho důvodu je importován v Assets > Import Package > Custom Package. V průzkumníku souborů je zvolen příslušný soubor. Při importování jsou odškrtnuty skripty SaveToExcel, Deafault Camera Y a Recenter_Camera (je možné je smazat i po importování). Tyto skripty momentálně způsobují errorry a nejsou potřebné pro další postup.



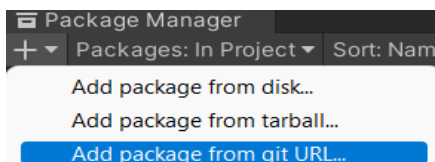
Obr. 3-12: HT framework import

3.4 HPTK

HPTK je asset dostupný zcela zdarma. Na Unity Asset Storu je ale možné si zakoupit rozšíření HPTK Posing & Snapping za 25 dolarů. Jedná se o poměrně rozdílný způsob zvládnutí řešeného problému. Místo skriptu, který by uchopil předmět přiblížením dvou prstů, se HPTK zabírá uchopením na základě fyziky rukou.

- HPTK import a nastavení [24]

HPTK je nainportováno pomocí odkazu, který je k dispozici na stránkách dokumentace assetu. Ten je zadán do vyskakovacího okna, které lze najít v Package manageru.

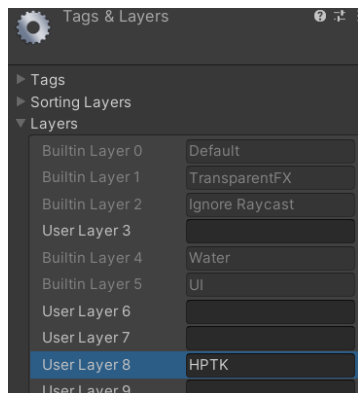


Obr. 3-13: HPTK import

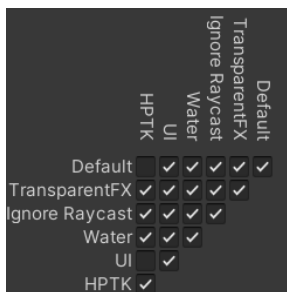
HPTK spoléhá, při simulaci fyziky rukou, na technologii nVidia PhysX a integrované konfigurovatelné klouby Unity. Proto je doporučeno toto nastavení v Project Settings:

- Physics
 - Default Solver Iterations – 25
 - Default Solver Velocity Iterations – 15
 - Enable Adaptive Force – enabled
 - Friction Type – One Directional Friction Type
 - Solver Type – Temporal Gauss Seidel
- Time
 - Fixed Timestep – 0.01

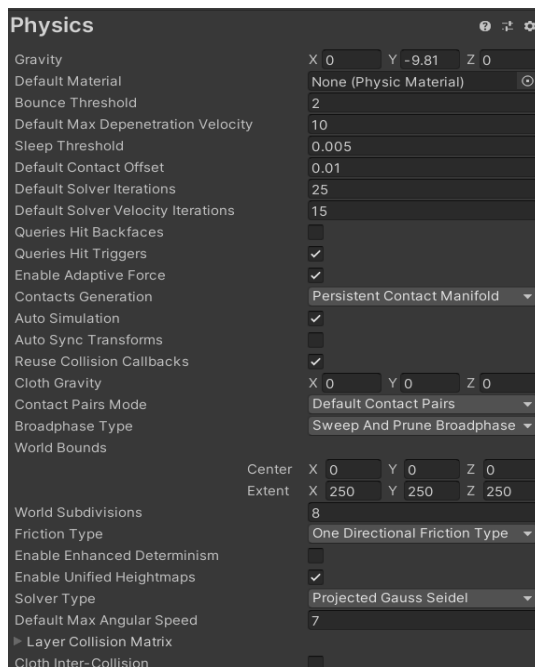
V dalším kroku je vytvořena nová vrstva HPTK. Ta je vytvořena ve slotu 8. Následně jsou v nastavení Physics změněny kolize vrstvy s vrstvami Default a UI.



Obr. 3-16: Vrstva HPTK



Obr. 3-15: Změna kolize s HPTK



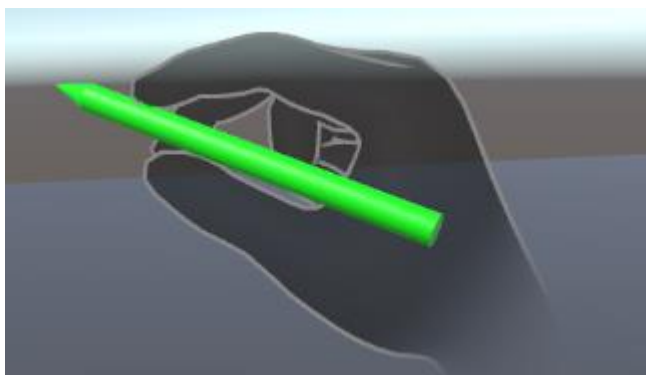
Obr. 3-14: Nastavení Physics

Pokud stále není v projektu nainstalován XR Plugin Management a nainportován Oculus Integration asset, tak to musí být provedeno teď. Oculus Integration je důležitý kvůli využití některých prefabů. Dále je otevřeno HPTK > Integration Manager a zvolena možnost Oculus. Tímto se nainportují poslední důležité skripty a prefabý pro zprovoznění hand trackingu.

3.5 Oculus Integration

Oculus Integration je dostupný zdarma na Unity Asset Storu. Importování tohoto assetu je již popsáno v první části Analýzy možností implementace. Je vyvíjen přímo firmou Oculus, a proto je základem pro většinu projektů zaměřených na její produkty. Jak již bylo zmíněno je využít u všech předchozích řešení, a to ve formě prefabu OVRCameraRig. Další důležitou součástí jsou prefabý rukou.

Oculus Integration nabízí kromě běžného braní předmětů i možnost implementace póz rukou, do kterých je virtuální ruka nastavena při uchopení daného předmětu. Pózy se dají vytvářet s využitím hand trackingu, postupným natáčením kloubů prefabu ruky, nebo kombinací zmíněných metod. Podobný systém je využíván u pózování ruky pro braní pomocí ovladačů u většího množství assetů. Příkladem může být například HurricaneVR nebo VR Interaction Framework, ten ovšem tuto formu nepodporuje pro hand tracking.



Obr. 3-17: Příklad vytvořené pózy

4 Popis realizace vlastní implementace

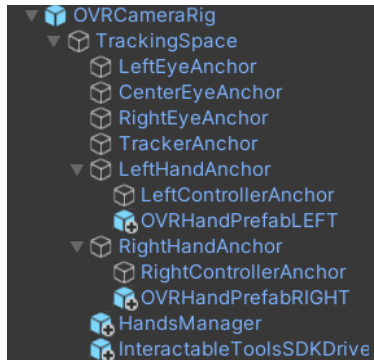
Po analýze jednotlivých možností implementace hand trackingu je potřeba vytvořit jednotlivé scény, ve kterých bude hand tracking fungovat. Dále je potřeba vytvořit jednotné prostředí pro všechny projekty. V tomto prostředí je provedeno testování hand trackingu, a nakonec je vybrána nejlepší možnost.

4.1 Vytvoření jednotlivých scén

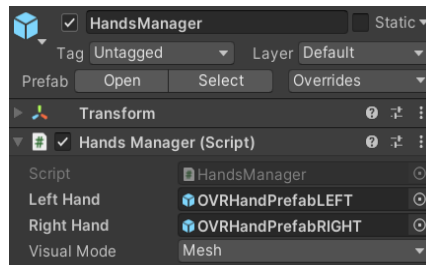
V této sekci jsou vytvořeny jednotlivé scény s hand trackingem společně s funkčním uchopením předmětů. Všechna řešení jsou představena krok po kroku, které byly podstoupeny pro zprovoznění jednotlivých scén. Postup je doplněn o obrázky, které pomáhají v orientaci v samotném Unity.

4.1.1 Rozšíření skriptu OVRGrabber z Oculus Integration

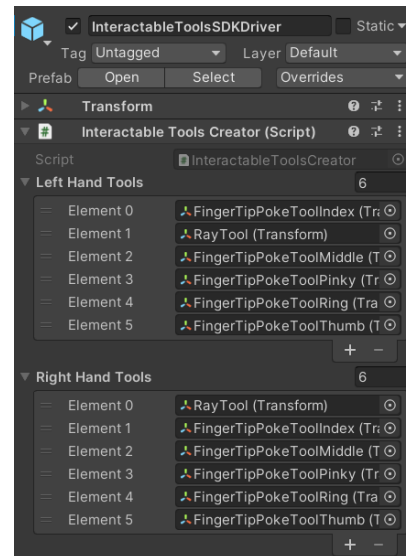
Scéna využívá OVRCameraRig. Pro zviditelnění rukou ve VR je potřeba přidat pod LeftHandAnchor a RightHandAnchor OVRHandPrefab (k názvu jsou přidány slova LEFT a RIGHT kvůli rozpoznatelnosti – není nutné). V podokně Project jsou vyhledány objekty HandsManager a InteractableToolsSDKDriver. Ty jsou přidány do scény jako děti OVRCameraRig. V HandsManageru jsou přiřazeny reference na hand prefaby. Dále je zde možné přepnutí vizuálu ruky na skeleton. Druhý objekt slouží k interakcím s předměty. V základu jsou přiřazeny konečky jednotlivých prstů a RayTool (paprsek vycházející z ruky pro interakce se vzdálenými objekty).



Obr. 4-1: Hierarchie před skriptem



Obr. 4-2: HandsManager



Obr. 4-3: InteractableToolsSDKDriver

Pro možnost uchopení předmětu bude napsán jednoduchý skript. K hand prefabum je přidán nový komponent, ale místo již existujícího skriptu je zvolen New script, který je pojmenován HandTrackingGrabber. Celý nový skript je k vidění na obrázku. Důležitým krokem je přidání řádky 4, která povolí využívání elementů z Oculus frameworku a přepsání MonoBehaviour na OVRGrabber v řádce 6, jelikož se novým skriptem pouze rozšiřuje skript používaný pro ovladače. Úlohou skriptu je zjistit, zda se k sobě přiblíží palec a v tomto případě ukazováček (lze změnit na další prsty). Pokud je tato podmínka splněna, je předmět uchopen. [25]

```

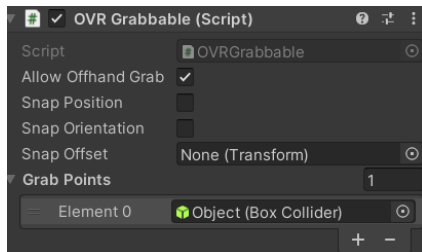
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4  using OculusSampleFramework;
5
6  @ UnityScript (2 asset references) | 0 references
7  public class HandTrackingGrabber : OVRGrabber
8  {
9      private OVRHand hand;
10     public float pinchThreshold = 0.7f;
11
12     @ Unity Message | 3 references
13     protected override void Start()
14     {
15         base.Start();
16         hand = GetComponent<OVRHand>();
17     }
18
19     @ Unity Message | 3 references
20     public override void Update()
21     {
22         base.Update();
23         CheckIndexPinch();
24     }
25
26     1 reference
27     void CheckIndexPinch()
28     {
29         float pinchStrength = GetComponent<OVRHand>().GetFingerPinchStrength(OVRHand.HandFinger.Index);
30         bool isPinching = pinchStrength > pinchThreshold;
31
32         if (!m_grabbedObj && isPinching && m_grabCandidates.Count > 0)
33             GrabBegin();
34         else if (m_grabbedObj && !isPinching)
35             GrabEnd();
36     }
37 }

```

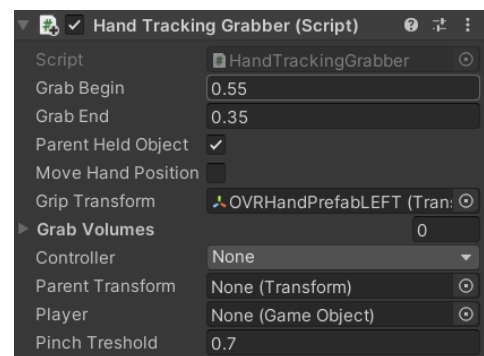
Obr. 4-4: Hand Tracking Grabber skript

Po dopsání skriptu je v unity zaškrtnuto políčko Parent Held Object a přiřazena reference na příslušný hand prefab v Grip Transform. Následně je nutné přidat Sphere Collider se zaškrtnutým Is Trigger a rádiusem 0,2 (čím větší tím se snáze předměty uchopí).

Předměty pro uchopení potřebují Rigidbody, Collider a OVR Grabbable skript, ve kterém je přiřazen collider předmětu do Grab Points.



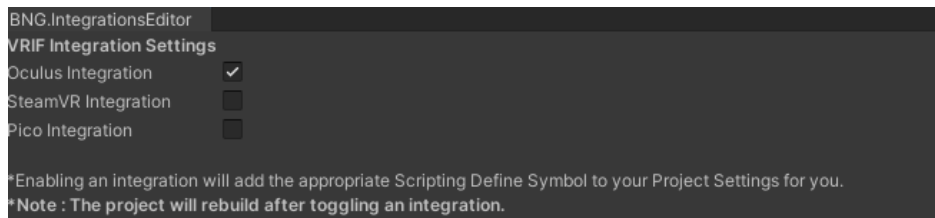
Obr. 4-6: OVR Grabbable



Obr. 4-5: Hand Tracking Grabber v unity

4.1.2 VRIF

V projektu s nainstalovaným VRIF je nejdříve povolena podpora Oculus Integration. Toho bylo dosaženo zaškrtnutím zaškrťovacího políčka ve VRIF (horní lišta) > VRIF Integrations.

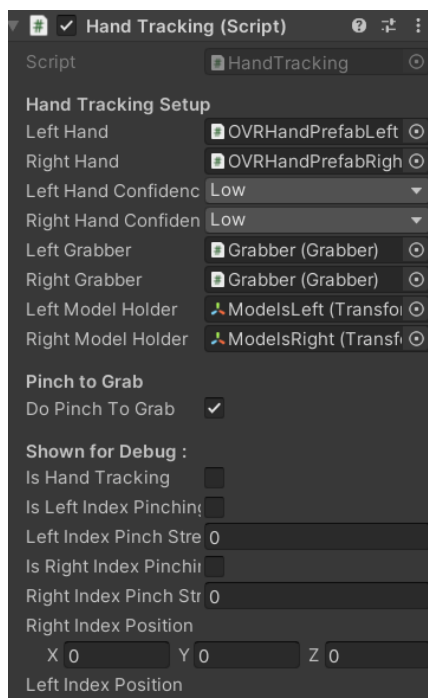


Obr. 4-7: VRIF Integrations

Následně je otevřen asset připravený ve složce Assets > BNG Framework > Integration > Oculus Integration, nazvaný Oculus Integration. Tím se naimportuje vše potřebné pro zprovoznění hand trackingu v tomto projektu.

Do scény je přidán nově vytvořený Oculus Camera Rig Advanced, který už v sobě má připravený Hand Tracking skript. Tento camera rig má již přiřazené hand prefaby. Dále je v něm přednastaven Hand Tracking Support na Hands only stejně tak jako Tracking Origin Type na Floor Level.

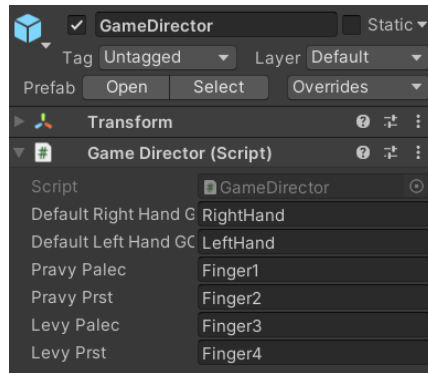
Na závěr pouze stačí přidat Rigidbody, Collider a Grabbable skript k předmětům, které mají být uchopitelné.



Obr. 4-8: VRIF Hand Tracking

4.1.3 HT framework

Do scény je jako první přidán GameDirector. Tento objekt obsahuje důležitý skript pro fungování hand trackingu. V GameDirectoru jsou přiřazeny názvy objektů rukou a prstů pomocí kterých bude možno uchopit předměty ve VR. Na obrázku jsou k vidění předem nastavené názvy, které jsou využity v tomto postupu.



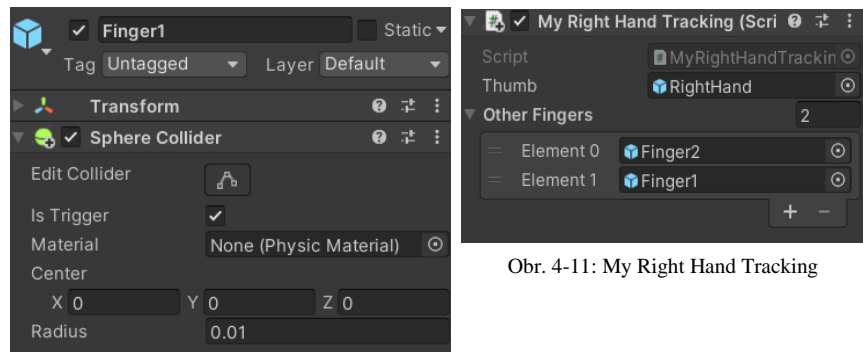
Obr. 4-9: GameDirector

Ve scéně je použit OVRCameraRig. Dále jsou pod LeftHandAnchor a RightHandAnchor přidány OVRCustomHandPrefab_L a OVRCustomHandPrefab_R, které jsou přejmenovány na LeftHand a RightHand podle GameDirectoru. V obou hand prefabech je následně použito tlačítko Auto Map Bones, které přiřadí jednotlivé objekty rukou na příslušná místa.

Poté jsou přiřazeny skripty My Left Hand Tracking a My Right Hand Tracking k náležitým rukou. Ve skriptu jsou nyní přiřazeny reference na ruku a dva koncečky prstů. Pro to jsou použity koncečky palce a ukazováčku. Oba prsty lze najít podle obrázku pod jménem b_1_thumb_null pro palec a b_1_index_null pro ukazováček. Pro pravou/levou ruku je přejmenován palec na Finger1/Finger3 a ukazováček na Finger2/Finger4. K jednotlivým prstům jsou nakonec přiřazeny Sphere Collider, ve kterých je nutné zaškrtnout Is Trigger a nastavit rádius.



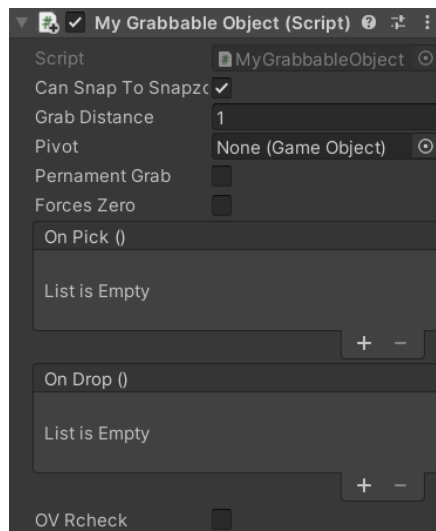
Obr. 4-10: Hierarchie prstů



Obr. 4-11: My Right Hand Tracking

Obr. 4-12: Příklad Collideru

Pro nastavení uchopitelných předmětů stačí přidat komponent RigidBody, Collider a skript My Grabbable Object.



Obr. 4-13: My Grabbable Object

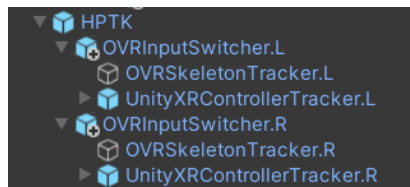
4.1.4 HPTK

Do scény stačí přidat prefab DefaultSetup.Oculus, jelikož v sobě již obsahuje OVRCameraRig a HPTK prefab. Oba tyto prefaby jsou dostupné samostatně. A je tedy možné dosáhnout podobného výsledku tímto způsobem: [24]

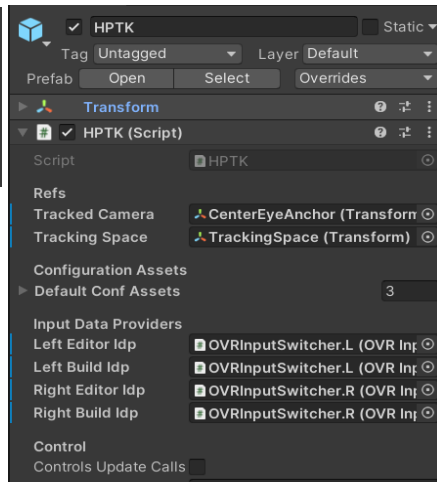
Do scény je přidán OVRCameraRig z Oculus Integration. Dále je pod LeftControllerAnchor a RightControllerAnchor přidán OVRHandPrefab (kvůli rozpoznatelnosti byly přejmenovány). U pravé ruky je zapotřebí změnit Hand Type, Skeleton Type a Mesh Type na možnost Hand Right.

V dalším kroku je do scény přidán objekt HPTK. Pod tento objekt jsou následně přiřazeny objekty OVRInputSwitcher.L a OVRInputSwitcher.R. V objektu HPTK je nutné přiřadit referenci na TrackingSpace, což je dítě OVRCameraRig, a na CenterEyeAnchor, což je dítě TrackingSpace. Dále je zde nutné přiřadit reference na OVRInputSwitchery v Input Data Providers. Následně jsou přiřazeny reference na hand prefaby v OVRSkeletonTracker.L, který je k dohledání pod OVRInputSwitcher.L, a to samé je provedeno pro OVRSkeletonTracker.R.

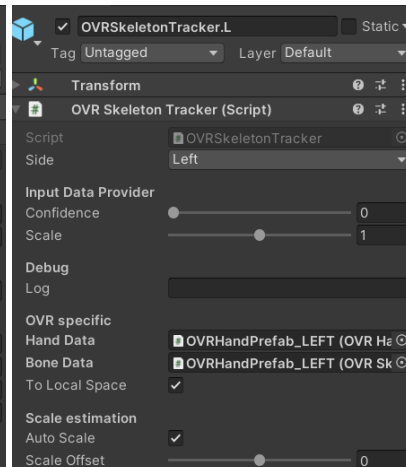
Jako poslední je do scény přidán DefaultAvatar.Standard, nebo DefaultAvatar.URP podle používané pipeline. Pro uchopitelné předměty stačí přidat pouze Collider, RigidBody, ve kterém je změněn Collision Detection na Continuous Dynamic, a změnit jejich vrstvu na HPTK. [24]



Obr. 4-16: HPTK hierarchie

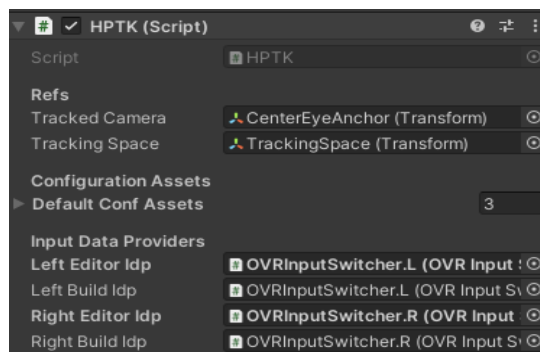


Obr. 4-14: Nastavení HPTK



Obr. 4-15: Nastavení OVRSkeletonTracker

Pro vyzkoušení hand trackingu první možnosti bez vytvoření aplikace je potřeba přiřadit OVRInputSwitchery místo PoseSkeletonTrackerů ve skriptu HPTK v objektu HPTK. [24]



Obr. 4-17: Změna HPTK

4.1.5 Oculus Integration

Stejně jako u předchozích implementací je zde provedeno vytvoření scény s hand tracking a uchopením. K tomu je však ukázán příklad implementace póz rukou.

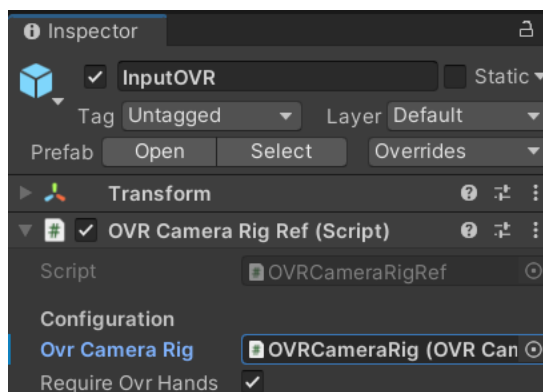
- Hand tracking a uchopení objektu [25]

Jako první jsou přidány hand prefaby pod OVRCameraRig. Proto je vyhledán OVRCustomHandPrefab_L a OVRCustomHandPrefab_R, které jsou následně přidány do scény pod LeftHandAnchor a RightHandAnchor. Díky tomuto se budou zobrazovat ruce při používání hand trackingu.



Obr. 4-18: Přiřazení Hand prefabů

Dalším krokem je umožnění interakcí ve VR. Proto je vyhledán a přidán do scény InputOVR prefab. V něm je pod skriptem OVR Camera Rig Ref přiřazen OVRCameraRig ze scény. Pokud by VR bylo zapnuté nyní, zobrazovaly by se dva překrývající modely rukou. To se děje, protože v InputOVR je již přidán jeden vizuál rukou a druhý byl přidán v předchozím kroku. Pro odstranění je vypnut prefab LeftHandVisual pod InputOVR > Hands > LeftHand a to samé je provedeno pro RightHandVisual. Toho je docíleno odškrtnutím políčka vedle názvu v podokně Inspector.

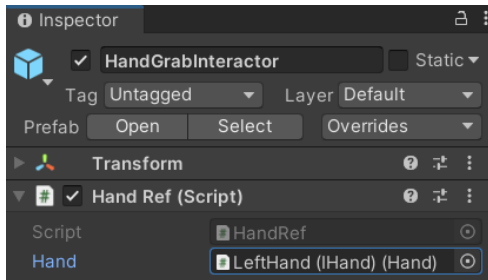


Obr. 4-20: Přiřazení CameraRig reference

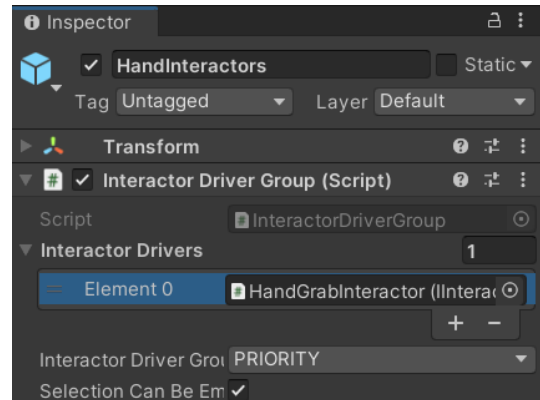


Obr. 4-19: InputOVR po vypnutí vizuálů

Pro pokračování je vyhledán HandGrabInteractor a přidán pod HandInteractors pod levou i pravou rukou. U HandGrabInteractor u levé ruky je přidána reference na levou ruku přetažením LeftHand do Hand Ref scriptu, a to samé je uděláno pro pravou ruku. Dále v HandInteractors je přidán HandGrabInteractor do Interactor Drivers opět u obou rukou. Tím je připraven hand tracking na bráni předmětů.

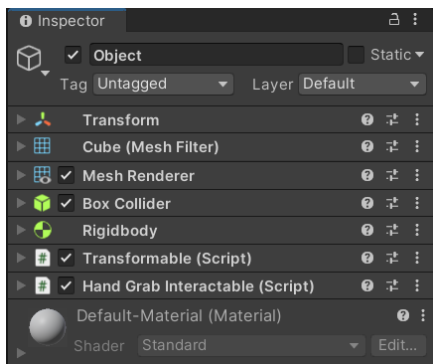


Obr. 4-21: Reference ruky u HandGrabInteractor

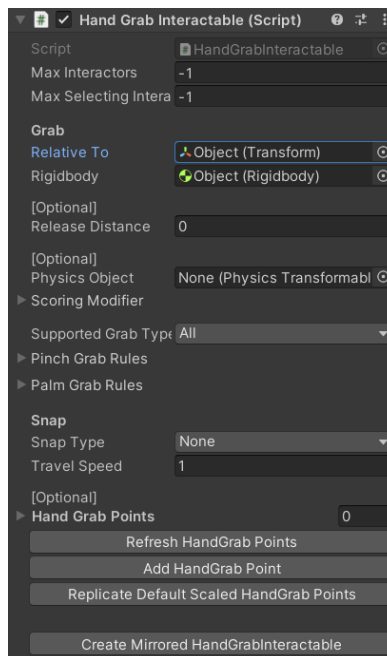


Obr. 4-22: Přidání HandGrabInteractor do Interactor Drivers

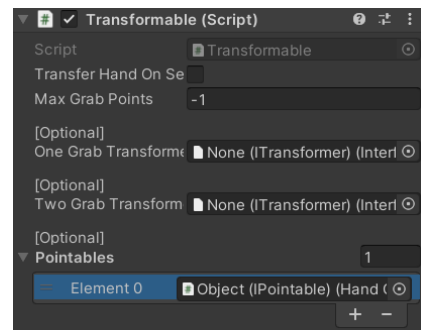
Jako další je připraven předmět, se kterým bude možno interagovat. Objekt je přidán do scény. Objektu jsou následně přiřazeny komponenty Rigidbody, jakýkoliv Collider, Transformable script a Hand Grab Interactable skript. V posledním skriptu je přiřazen Transform a Rigidbody objektu a změněn Snap Type na None. Dále je možné nastavit typ uchopení předmětu (pinch nebo palm) s následným nastavením vyžadovaných prstů k úchopu. Script Hand Grab Interactable je přesunut pod Pointables v Transformable skriptu. Nyní lze s objektem interagovat.



Obr. 4-25: Objekt se všemi komponenty



Obr. 4-23: Nastavení Hand Grab Interactable



Obr. 4-24: Nastavení Transformable

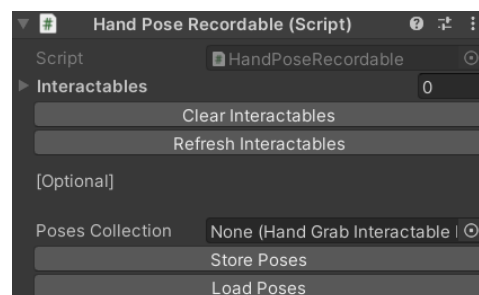
- Hand tracking uchopení s pózou [25]

Scéna s potřebnými objekty je již připravena z předchozí části. Do scény je přidán objekt Mug. Jedná se o prostý hrnek, na kterém bude ukázán proces nastavení pózy ruky při uchopení. Na objekt Mug jsou přidány komponenty Rigidbody, Mesh Collider a Transformable.

Pro nahrání pózy je do scény přidán prázdný objekt, který je pojmenován Hand Pose Recorder. Dále je k němu přiřazen stejnojmenný skript. Ke skriptu je dále přiřazena ruka, která bude nahrávána (Righthand), a Transform předmětu, pro který bude póza nahrávána (mug). Dále zde lze nastavit tlačítko pro zaznamenání pózy (mezerník). Přiřazení GhostProvider u Ghost Provider umožní prohlédnutí pózy po zaznamenání přímo ve VR. K objektu Mug je přidán Hand Pose Recordable.

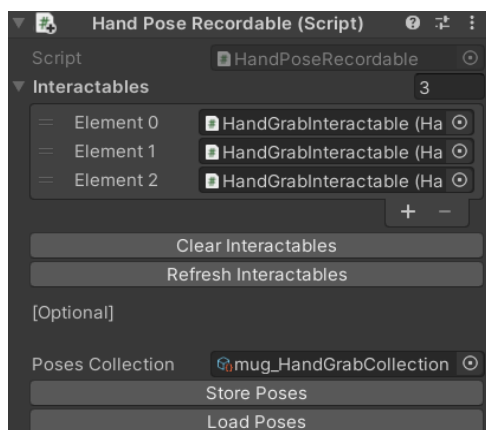


Obr. 4-26: hand Pose Recorder

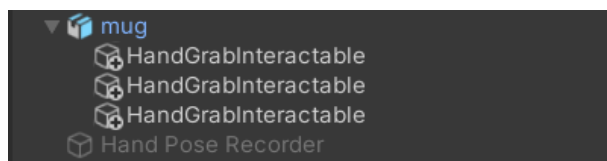


Obr. 4-27: Hand Pose Recordable

Nyní lze spustit scénu s připojeným headsetem a nahrát si jednotlivé pózy. Po zaznamenání jednotlivých póz, a ještě před vypnutím scény je důležité si pózy uložit pomocí tlačítka Store Poses v Hand Pose Recordable. Tímto se vytvoří soubor [Název objektu]_Hand Grab Collection. Ukončí se scéna a vzniklý soubor se přetáhne do Poses Collection, kde se následně nahraje pomocí tlačítka Load Poses. Nahrané pózy jsou k nalezení v Interactables a jako děti objektu Mug. Je doporučeno vypnout objekt Hand Pose Recorder po nahrání všech póz, jinak může dojít k dalšímu nechtěnému nahrávání.

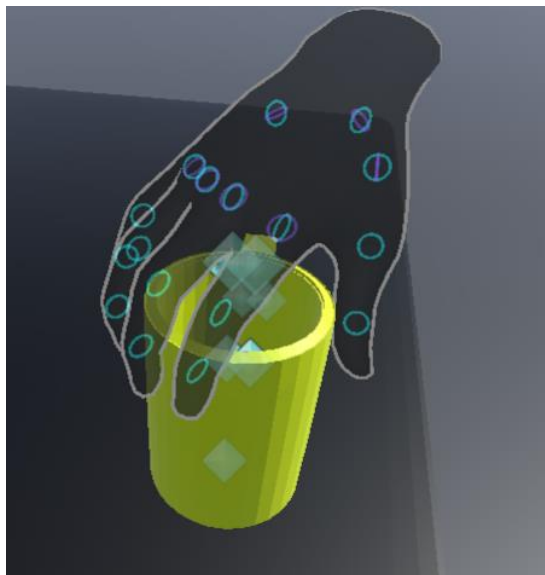


Obr. 4-28: Hand Pose Recordable po nahrání póz

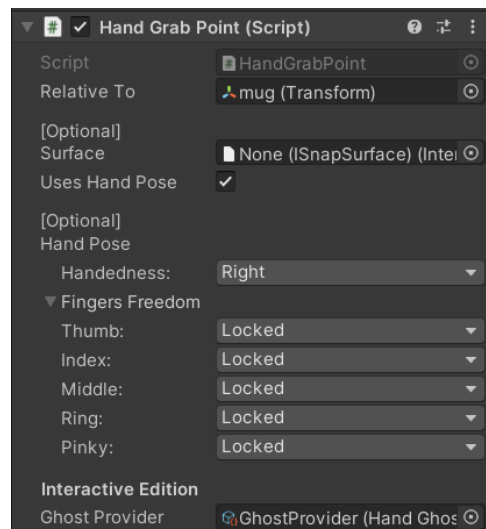


Obr. 4-29: Hierarchie po nahrání póz

Zvolením jednotlivých dětí v hierarchii mohou být příslušné pózy upraveny. Dále je možné nastavit pohyblivost prstů (v příkladu je poloha prstů zamknutá) a případně změnit o jakou ruku se jedná. Další možností je vytvoření duplikátu pózy pro druhou ruku pomocí tlačítka Create Mirrored HandGrabInteractable.



Obr. 4-30: Příklad nahrané pózy

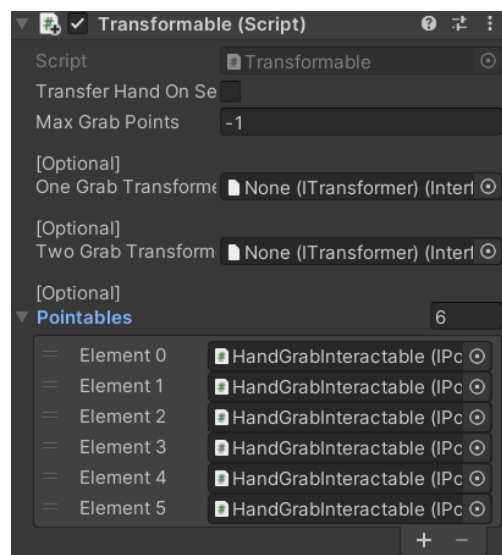


Obr. 4-31: Příklad nastavení pózy

Každá póza má přiřazený Hand Grab Interactable skript. Opět je tedy nutné přiřadit RigidBody objektu u všech póz. A následně přiřadit všechny pózy do Transformable skriptu objektu pod Pointables.



Obr. 4-32: Nastavení Hand Grab Interactable 2

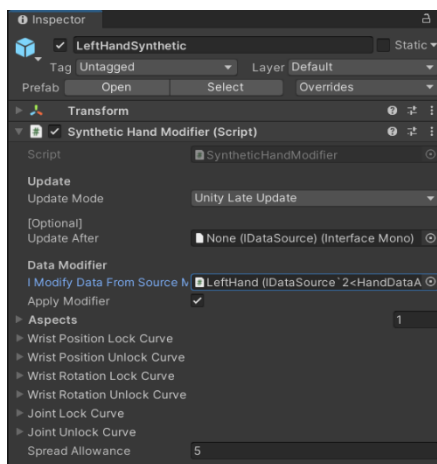


Obr. 4-33: Nastavení Transformable 2

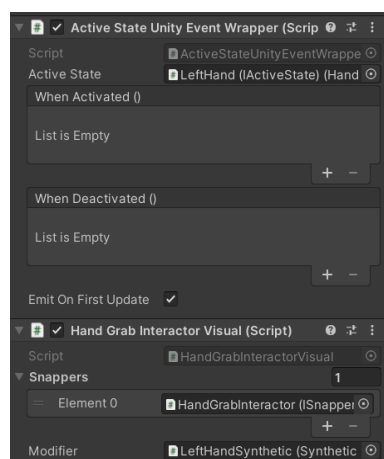
Aby se dané pózy mohly využít, je nutné umožnit zamknutí rukou při uchopení předmětu. Toho je docíleno přidáním LeftHandSynthetic pod InputOVR > Hands > LeftHand a RightHandSynthetic pod RightHand. V LeftHandSynthetic je přiřazena LeftHand pod I Modify Data From Source Mono a Active State. Dále je přiřazen HandGrabInteractor pod Snappers. To samé je provedeno pro pravou ruku.



Obr. 4-35: Hierarchie s HandSynthetic



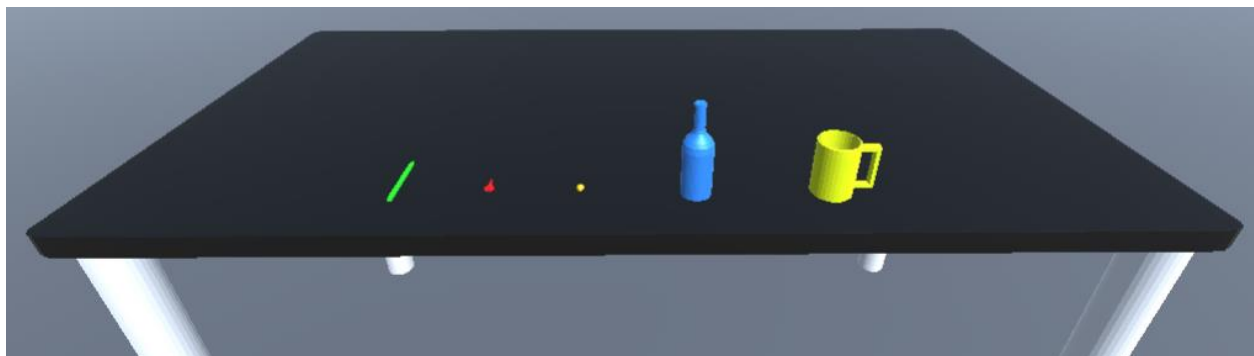
Obr. 4-34: LefthandSynthetic 1



Obr. 4-36: LeftHandSynthetic 2

4.2 Vytvoření jednotného prostředí

Nejdříve je potřeba vytvořit objekt, který bude působit jako země. Do scény je přidán nový objekt krychle, u které jsou změněny rozměry a přidána šedivá barva. Dále byl podobným způsobem vytvořen stůl. V dalším kroku bylo pomocí aplikace Blender vytvořeno pět jednoduchých modelů objektů, které by se mohly vyskytovat v běžném prostředí VR. Jedná se o tužku (zelená), klíč (červená), kulička (oranžová), láhev (modrá) a hrnek (žlutá). Tyto předměty byly dále zvoleny z důvodů reprezentace různých úchopů a velikostí předmětů. Je tedy možné je nahradit jinými podobnými příklady. Předměty vytvořené v Blenderu jsou následně naimportovány do Unity a to konkrétně v Assets > Import New Assets.



Obr. 4-37: Prostředí

Tato základní scéna je exportována za pomoci Assets > Export Package. Ve vyskočeném okně jsou zaškrtnuty všechny potřebné předměty. Tento nově vzniklý asset je následně naimportován do jednotlivých projektů a k předmětům jsou přidělené potřebné komponenty.

4.3 Měření

Z testování byly vyloučeny první dvě řešení hand trackingu, konkrétně rozšíření skriptu a VRIF. To bylo učiněno z důvodu jednoduchosti provedení těchto možností. Rozšíření skriptu je pouze ukázkou, že takové řešení je možné. Dovoluje pouze uchopit předměty stiskem palce a ukazováčku. I to je ovšem poměrně nespolehlivé, a to především u menších předmětů. VRIF funguje na podobném principu. I když je v tomto případě spolehlivější, stále se jedná pouze o prototyp. Navíc hand tracking není hlavním zaměřením tohoto assetu, který má ovšem spoustu jiných výhod. Testování je tedy provedeno pouze na zbylých třech možnostech, HT framework, HPTK a Oculus Integration.

Ve všech třech aplikacích bylo provedeno měření spolehlivosti trackování. Předmět byl uchopen v předpažené ruce sevřené do pěsti. To představovalo základní polohu (viz Obr. 3-38). Z předpažení byl předmět postupně zvedán do výpadku trackování (modely rukou zmizí, nebo se zaseknou v prostoru), nebo do vypadnutí předmětu z ruky. Toto měření bylo provedeno na dvou předmětech. Prvním byla láhev, zastupující větší předměty, které jsou ve VR prostředích častější. Druhým předmětem byla kulička, zastupující malé předměty. Oba předměty byly otestovány pětkrát v každé aplikaci. Celý proces testování byl proveden v odpoledních hodinách v místnosti s přirozeným světlem za oblačného dne.

Testování probíhalo před zafixovaným fotoaparátém. Testovaná osoba se postavila před bílou zeď a provedla již zmíněný pohyb s předpaženou rukou. Vždy při přerušení trackování, nebo vypadnutí předmětu byl pohyb zastaven. V tu chvíli byla pořízena fotografie (viz Obr. 3-39). Následně byl pomocí počítače odečten úhel rozdílu poloh. U výsledků je nutné počítat s jistou chybou, která může nastat drobnými pohyby hlavy a rukou.



Obr. 4-38: Základní poloha



Obr. 4-39: Poloha při vypadnutí

Stejným způsobem bylo otestováno držení předmětu pouze palcem a ukazováčkem. V tomto případě však docházelo k vypadnutí na hranici, nebo za hranici zorného pole. K tomu nejspíše docházelo z důvodu lepší viditelnosti prstů během zvedání předmětu a tím pádem i k lepšímu trackování ruky. Při pohybu do ostatních stran trackování vypadávalo až po opuštění zorného pole uživatele, a proto nebylo měření provedeno.

- HPTK

Jako první byla otestována aplikace s HPTK assetem. Při měření byly zjištěny tyto hodnoty:

Tab. 4-1: Tabulka s naměřenými úhly ve stupních pro aplikaci s HPTK

HPTK	
Láhev [°]	Kulička [°]
28	27
33	17
24	19
30	20
26	24

Z výsledků lze sledovat rozdíl mezi spolehlivostí při uchopení rozdílně velkých předmětů. Hlavním důvodem fenoménu je ovšem fungování tohoto braní předmětů, které je postaveno na fyzice rukou. U větších předmětů je tento způsob poměrně spolehlivý, přesto však dochází k občasnému vypadnutí předmětu. U menších předmětů k předčasnému vypadnutí dochází častěji. Výkyvy v naměřených hodnotách jsou způsobeny právě předčasným vypadáváním.

- HT framework

V tomto případě byly naměřeny hodnoty:

Tab. 4-2: Tabulka s naměřenými úhly ve stupních pro aplikaci s HT frameworkem

HT framework	
Láhev [°]	Kulička [°]
29	16
38	19
36	17
37	18
35	20

Z naměřených hodnot lze opět sledovat rozdíl mezi předměty. U láhve byly naměřeny nejlepší výsledky ze všech třech aplikací. K vypadnutí trackování docházelo, až na jednu výjimku, na hraně zorného pole, kdy předmět už nebyl z části vidět. U kuličky ovšem docházelo k vypadnutí předmětu již při malém pohybu, a proto byly naopak naměřeny nejhorší hodnoty.

- Oculus Integration

Na závěr byla změřena aplikace postavená na Oculus Integration. Výsledkem byly tyto hodnoty:

Tab. 4-3: Tabulka s naměřenými úhly ve stupních pro aplikaci s Oculus Integration

Oculus Integration s pózou		Oculus Integration bez pózy	
Láhev [°]	Kulička [°]	Láhev [°]	Kulička [°]
32	29	29	32
30	32	31	30
31	30	30	31
32	31	33	32
33	32	30	30

Z naměřených hodnot lze pozorovat malé, nebo skoro žádné rozdíly mezi lahví a kuličkou. Dále je také vidět, že nezáleží na použití uchopení s pózou, nebo bez ní. Ve všech případech došlo k vypadnutí trackování blízko hranice zorného pole

4.4 Hodnocení

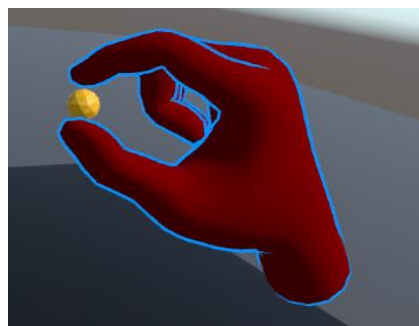
V této části jsou uvedeny výhody a nevýhody, které se objevily během procesu implementace a testování jednotlivých aplikací. Všechny poznatky jsou následně zpracovány do tabulky, kde jsou jednotlivé aplikace bodově ohodnoceny. Nakonec je vybrána aplikace, která dosáhla nejlepšího hodnocení.

4.4.1 HPTK

HPTK je rozhodně zajímavé svým jiným přístupem k celému problému. Tento přístup vede k většímu pocitu imerze ve virtuální realitě, a to hlavně z důvodu podobnosti úchopu předmětů k realitě. Uchopení probíhá nastavením ruky do podobné polohy jako kdyby byl držen skutečný předmět. Pokud se ruka sevře víc, nastává ve velkém počtu případů pokřivení modelu ruky.



Obr. 4-41: Úchop láhve HPTK



Obr. 4-40: Úchop kuličky HPTK

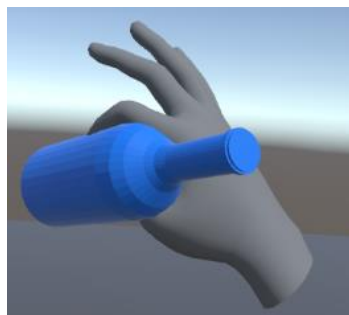
Tento způsob interakcí ale také vede k obtížím při snaze uchopit malý nebo plochý předmět ležící na desce. Dobrým příkladem může být sada tří předmětů použitých v prostředí na testování. Tužku i kuličku bylo možné uchopit jen za specifických okolností, správném pokrčení prstů a nastavením dostatečně malé váhy v Rigidbody. To se také projevilo při měření v předchozí kapitole. S klíčem nebylo možné interagovat skoro vůbec. Tyto problémy by mohly být řešené, podle dostupných

informací, v již zmíněném rozšíření HPTK Posing & Snapping. Pokud došlo k přerušení trackování rukou, tak byl předmět upuštěn.

Celý asset je přehledný a lze jej jednoduše implementovat. Jsou k němu dostupné stránky s dokumentací, ve které je možné dohledat potřebné informace a postupy. Podporuje několik vstupů pro hand tracking jako Oculus Quest 1/2, Hololens 2, Leap Motion a WebXR. Stejně tak podporuje ovladače, například Oculus Touch, WMR, Vive, OpenVR. Jedná se ale o méně používaný asset, což vede k menší komunitě a nedostatku návodů na případné problémy.

4.4.2 HT framework

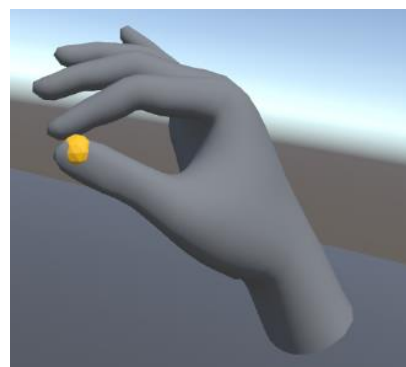
Tento asset opět přináší trochu jiný pohled na uchopení předmětů. Jedná se o takový mezistupeň HPTK a Oculus Integration, jelikož je možné uchopení ihned po vstoupení colliderů prstů do collideru předmětu. Díky tomu lze předmět uchopit pinchem nebo uchopením podobným realitě.



Obr. 4-43: Úchop láhve HT 1



Obr. 4-42: Úchop láhve HT 2



Obr. 4-44: Úchop kuličky HT

Předměty ovšem jdou uchopit pouze palcem a ukazováčkem. Občas se také může stát, že se jedním úchopem mohou posouvat dva objekty. S aktuálním nastavením se také vyskytuje problém s nejistým uchopením malých předmětů, jejichž collidery nejsou dostatečně objemné. Během měření byly sledovány velké výkyvy při brání různě velkých předmětů. Pokud došlo k přerušení trackování, předmět byl upuštěn (v některých případech zahazen).

Jedná se o neveřejný asset, který má o něco složitější implementaci. Je stále ve vývoji a tím pádem je možné chyby napravit a konzultovat s vývojáři.

4.4.3 Oculus Integration

Tento asset se ještě nedávno potýkal s jistými problémy s fungováním. To se také podepsalo na jeho nízkém hodnocení v Unity Asset Storu. Momentálně je ale v dobrém stavu. To se také dá dokázat jeho použitím i v ostatních implementacích. Jako jedna z mála možností nabízí také vytvoření póz pro hand tracking.

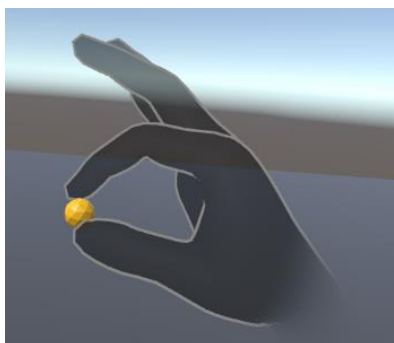
Uchopitelnost předmětů je velmi dobrá, a to i u předmětů s více nahanými pózami, kde je vždy použita právě chtěná póza. Při měření bylo dosaženo nadprůměrných výsledků u všech předmětů. U této jediné možnosti se nevyskytl problém s uchopením malých předmětů. Dále je i po přerušení trackování a opětovném návratu rukou do zorného pole předmět držen. Předměty je možné uchopit do jakékoliv kombinace palce a jiného prstu (až na malíček). Grabbable skript nabízí největší počet možných nastavení.



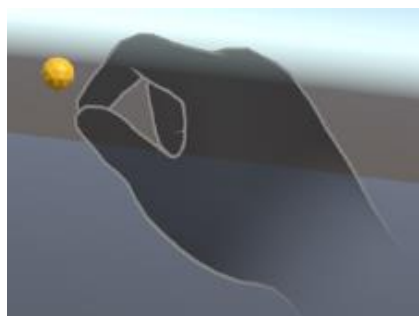
Obr. 4-45: Úchop láhve póza IO



Obr. 4-48: Úchop láhve bez pózy OI 2



Obr. 4-47: Úchop kuličky póza OI 1



Obr. 4-46: Úchop kuličky bez pózy OI 2

Jisté problémy mohou nastat při velmi výjimečné neuchopitelnosti předmětů bez póz. Dalším problematickým místem je snaha uchopit předmět s více nastavenými pózami oběma rukama najednou. Při tomto počínu dochází k odpoutání rukou s předmětem od hráče a vystřelením v náhodném směru. Při následovném puštění předmětu je vyvolána reakce, která většinou smete všechny předměty nacházející se před hráčem.

Celá implementace Oculus Integration je o něco složitější než předchozí postupy. Výhodou ovšem je vývoj právě firmou Oculus. Asset je pravidelně aktualizován a je spojen s velkou komunitou, která nabízí mnoho poraden a návodů.

4.4.4 Finální výběr

S přihlédnutím k jednotlivým hodnocením implementací byla vytvořena tabulka, která má za úkol ohodnotit několik atributů. Pro přesnější hodnocení byly aplikace otestovány na pěti dobrovolnících, kteří měli buďto malé, nebo žádné předchozí zkušenosti s VR. Hodnocení probíhá na škále od 1 do 5. Nejlepší volba bude dosahovat největšího počtu bodů. Jednotlivé hodnocené atributy jsou:

- **Měření spolehlivosti**

Zde jsou bodově ohodnoceny výsledky z měření. Nejlépe dopadl HT framework u láhve, a proto je ohodnocen 5 body. Za ním se umístil Oculus Integration pro oba předměty se 4 body. HPTK pro láhev dostalo pouze 3 body, jelikož docházelo k vypadnutí předmětu. Pokud ovšem předmět nevypadl, výsledky byly uspokojivé. Nejméně bodů dostaly zbylé dvě možnosti, HPTK a HT framework pro kuličku. I když se na první pohled výsledky trochu liší, důvody pro ně byly stejné, a proto jsou obě možnosti ohodnoceny 2 body.

- **Ovládání**

Tato kategorie je hodnocením intuitivnosti a jednoduchosti ovládání s využitím zpětné vazby od dobrovolníků. Všichni hodnotitelé preferovali aplikaci s Oculus Integration. Dosáhla tedy 5 bodů. HT framework pak následoval se 4 body. Jeho ovládání bylo intuitivní pro všechny uživatele. Problémem je ovšem již zmíněné zvedání menších předmětů, kde může dojít k vypadnutí. Posledním je HPTK se 2 body. Každý dobrovolník potřeboval více času na seznámení se s ovládáním oproti zbylým dvěma možnostem. Stejně tak uchopování předmětů dělalo ze začátku problémy. Tyto problémy byly ještě více zvýrazněny pro malé předměty. Dále se projevila nemožnost uchopení u některých předmětů.

- **Interaktivnost**

Zde se hodnotila celková míra interakcí opět s použitím zpětné vazby. Po seznámení s ovládáním byla nejlépe ohodnocena aplikace s HPTK. To je hlavně z důvodu fyziky rukou, která je klíčová pro brání předmětů. Model ruky interaguje s prostředím stejně jako předměty, které jsou uchopené. Celkový zážitek se více podobá skutečnému světu. Ohodnocen je tedy 5 body. O bod méně dostala aplikace s Oculus Integration, protože v tomto případě nejsou interakce modelu ruky s prostředím. S uchopeným předmětem je však možné posouvat ostatní předměty. 3 body pak dostal HT framework, který navíc měl problémy s interakcí mezi uchopeným lehčím předmětem a volně položeným těžším předmětem. Docházelo totiž k vytlačení drženého předmětu z úchopu, místo posouvání nedržení předmětu po desce stolu.

- **Jednoduchost implementace**

Jedná se o bodové ohodnocení časové náročnosti implementace assetu. Nejrychlejší implementaci má HPTK s 5 body. Druhý je HT framework se 4 body a třetí je Oculus Integration s 3 body. Časový rozdíl mezi implementacemi není dostatečně velký pro větší rozdíly v bodování. Implementace jsou popsány výše.

○ **Podpora assetu**

Poslední hodnocený atribut se zaměřuje na celkovou podporu. U assetů je zkoumána dostupnost návodů, dokumentací a velikost komunity. Výrazně nejlépe je na tom Oculus Integration. To hlavně kvůli tomu, že se jedná o asset přímo vyvíjený pro Oculus headsety. Na internetu je dostupná oficiální dokumentace i neoficiální fóra a návody. Dostává tedy 5 bodů. Druhý je HPTK se 3 body. Jedná se o veřejný asset. Problémem ovšem je jeho poměrně malý dosah. Na internetu je dostupná dokumentace. Neoficiální návody jsou ale více vzácné. Problémy se tedy musí většinou diskutovat přímo s vývojářem. Pouze 2 body získal HT framework, protože se jedná o neveřejný asset. Otázky a problémy se musí řešit přímo s vývojářem. Zde se ovšem jedná o pružnější konverzaci než u předchozích možností, jelikož je asset stále ve vývoji.

Výsledkem je tabulka s těmito hodnotami:

Tab. 4-4: Tabulka s bodovým ohodnocením jednotlivých implementací

	Měření spolehlivosti		Ovládání	Interaktivnost	Jednoduchost implementace	Podpora assetu	Celkem
	láhev	kulička					
HPTK	3	2	2	5	5	3	20
HT framework	5	2	4	3	4	2	20
Oculus Integration	4	4	5	4	3	5	25

Podle tabulky je jako nejlepší zvolen s 25 body Oculus Integration, který dosahuje nadprůměrných výsledků ve všech zvolených parametrech. O druhé místo se dělí s 20 body zbylé dvě možnosti.

Pořadí by se ovšem mohlo ještě změnit využitím rozšíření HPTK Posing & Snapping. Ze stejného důvodu je nejisté i budoucí umístění HT frameworku, který je stále vyvíjen. Tyhle skutečnosti by mohly sloužit jako námět pro další výzkum v budoucnosti.

Závěr

V bakalářské práci bylo nejdříve nutné představit technologii hand trackingu a další možnosti ovládání. Pro správné pochopení hand trackingu ve VR bylo v první části práce nutné si nejdříve definovat, co VR je. Na tuto definici bylo navázáno představením VR headsetů. Ty se dají považovat za první krok v ovládání samotného VR. Proto byly uvedeny tři různé příklady tzv. motion trackingu. Stejným způsobem se sleduje pohyb i VR ovladačů. Ty jsou nejběžnějším způsobem ovládání VR v dnešní době. Jejich nástupcem může být právě hand tracking. Jedná se stále o poměrně novou technologii, a proto bylo fungování nejnovějších systémů rozebráno v druhé kapitole Analýza současného stavu. V té byly také představeni momentální rivalové z řad VR ovladačů. Použití hand trackingu navíc zvyšuje stupeň zaučení uživatele.

V práci byl pro samotnou implementaci vybrán software Unity a brýle pro virtuální realitu Oculus Quest 2. Celkově bylo představeno pět možností implementace hand trackingu do virtuální reality. Z analýzy bylo zjištěno, že samotný proces zprovoznění hand trackingu je velmi podobný ve všech řešeních. V těch bylo využíváno základních prefabů z Oculus Integration assetu. Rozdíly se ukazovaly až v principu fungování úchopu předmětu ve virtuálním prostředí.

Jednotlivé projekty bylo nutné připravit pro implementaci naimportováním assetů potřebných pro dané řešení. Následně byly vytvořeny všechny scény, které obsahovaly jednotné prostředí. Pomocí těchto scén byly vytvořeny aplikace, které byly nainstalovány na zmíněný VR headset. Tyto aplikace byly využity pro další testování. Po prozkoumání fungování aplikací byly vyřazeny dvě z pěti možností. Zbylé aplikace byly vytvořeny pomocí HPTK assetu, HT frameworku a Oculus Integration assetu. Na těchto aplikacích bylo provedeno měření, ve kterém byly zjištěny pozice vypadnutí trackování. Měření bylo následně zpracováno do zhodnocení jednotlivých řešení v pěti kategoriích. Výsledkem bylo bodové ohodnocení možností implementace, kde nejlepší možnost získala nejvíce bodů.

Nejlepší aplikace byla vytvořena za použití assetu Oculus Integration. Hlavním důvodem byly velmi dobré výsledky z měření, intuitivní ovládání, které využívá všech prstů, možnost použití póz při uchopení předmětů a dobrá dostupnost návodů a dokumentace k assetu.

Vzhledem ke stálému vývoji hand tracking technologie je nutné podobný průzkum provádět opakovaně. Jednotliví výrobci stále budou zdokonalovat jejich řešení, vytvářet nové headsety a zařízení, které budou dosahovat lepších výsledků. Stejně tak se budou tvořit nové programy a assety pro implementace.

Seznam použitých zdrojů

- [1] *Virtual Reality - LaValle* [online]. [vid. 2021-11-16]. Dostupné z: <http://lavalle.pl/vr/>
- [2] *A New Reality: How VR Actually Works | by Zaynah Bhanji | Predict | Medium* [online]. [vid. 2021-11-26]. Dostupné z: <https://medium.com/predict/a-new-reality-how-vr-actually-works-663210bdf72>
- [3] *VR – Virtuální realita (VŠE, CO CHCETE VĚDĚT) | Alza.cz* [online]. [vid. 2021-11-26]. Dostupné z: <https://www.alza.cz/vr-virtualni-realita>
- [4] *A Brief Guide to VR Motion Tracking Technology - Antycip* [online]. [vid. 2021-11-26]. Dostupné z: <https://steantycip.com/blog/vr-motion-tracking/>
- [5] *How VR Positional Tracking Systems Work* [online]. [vid. 2021-11-27]. Dostupné z: <https://uploadvr.com/how-vr-tracking-works/>
- [6] *Virtual Reality Controllers: The Way of Interacting With the Virtual Worlds | Circuit Stream* [online]. [vid. 2021-11-26]. Dostupné z: <https://circuitstream.com/blog/vr-controllers-the-way-of-interacting-with-the-virtual-worlds/>
- [7] *How hand tracking in VR benefits your Industry* [online]. [vid. 2022-05-16]. Dostupné z: <https://blog.techviz.net/how-hand-tracking-in-vr-benefits-your-industry>
- [8] *What is Unreal Engine?* [online]. [vid. 2021-11-27]. Dostupné z: <https://conceptartempire.com/what-is-unreal-engine/>
- [9] *Making your Dream Games: What is Unity?* [online]. [vid. 2021-11-27]. Dostupné z: https://gamedevacademy.org/what-is-unity/#What_is_Unity
- [10] *Unity vs Unreal Engine - Which is Better For VR Development [2021 Updated] | Circuit Stream* [online]. [vid. 2021-12-01]. Dostupné z: <https://circuitstream.com/blog/unity-vs-unreal/>
- [11] *Unity Real-Time Development Platform | 3D, 2D VR & AR Engine* [online]. [vid. 2021-12-01]. Dostupné z: <https://unity.com/>
- [12] *The most powerful real-time 3D creation tool - Unreal Engine* [online]. [vid. 2021-12-01]. Dostupné z: <https://www.unrealengine.com/en-US/>
- [13] *Best VR Controller in 2021 - PC Guide* [online]. [vid. 2021-11-28]. Dostupné z: <https://www.pcguides.com/vr/guide/best-controller/>
- [14] *What is Oculus Touch?* [online]. [vid. 2021-12-03]. Dostupné z: <https://www.lifewire.com/oculus-touch-4159174>
- [15] *Exploring the magic behind the HTC Vive controller | VRHeads* [online]. [vid. 2021-12-03]. Dostupné z: <https://www.vrheads.com/exposing-magic-behind-htc-vive-controller>
- [16] *Hand Tracking Deep Dive: Technology, Design, and Experiences - YouTube* [online]. [vid. 2021-11-27]. Dostupné z: <https://www.youtube.com/watch?v=gpQePH-Ffbw>
- [17] *Haptic gloves for virtual reality and robotics | HaptX* [online]. [vid. 2021-11-27]. Dostupné z: <https://www.haptx.com/>

- z: <https://haptx.com/>
- [18] *How Oculus Quest Hand Tracking Technology Works* [online]. [vid. 2021-11-27]. Dostupné z: <https://www.online-tech-tips.com/gaming/how-oculus-quest-hand-tracking-technology-works/>
- [19] *Digital worlds that feel human | Ultraleap* [online]. [vid. 2021-11-28]. Dostupné z: <https://www.ultraleap.com/>
- [20] *Alza.cz – nakupujte bezpečně z pohodlí domova | Alza.cz* [online]. [vid. 2022-05-10]. Dostupné z: <https://www.alza.cz/>
- [21] *Hurricane VR - Physics Interaction Toolkit | Physics | Unity Asset Store* [online]. [vid. 2022-03-02]. Dostupné z: <https://assetstore.unity.com/packages/tools/physics/hurricane-vr-physics-interaction-toolkit-177300#description>
- [22] *Home | VRIF Wiki* [online]. [vid. 2022-03-04]. Dostupné z: <https://wiki.beardedninjagames.com/>
- [23] *VR Interaction Framework | Systems | Unity Asset Store* [online]. [vid. 2022-03-02]. Dostupné z: <https://assetstore.unity.com/packages/templates/systems/vr-interaction-framework-161066>
- [24] *Overview - HPTK* [online]. [vid. 2022-04-09]. Dostupné z: <https://jorge-jgnz94.gitbook.io/hptk/>
- [25] *Understand Package Components | Oculus Developers* [online]. [vid. 2022-04-12]. Dostupné z: <https://developer.oculus.com/documentation/unity/unity-utilities-overview/>
- [26] *Beat Saber opouští early access, dostane oficiální editor úrovní | GAMES.CZ* [online]. [vid. 2021-11-26]. Dostupné z: <https://games.tiscali.cz/novinky/beat-saber-opousti-early-access-dostane-oficialni-editor-urovni-328053>
- [27] *VR tracking: how does it really work? | our Blog | 4Experience.co* [online]. [vid. 2021-11-27]. Dostupné z: <https://4experience.co/vr-tracking-meet-degrees-of-freedom/>
- [28] *Entering values manually for an object location inside the Blueprint editor viewport doesn't work anymore. - UE4 AnswerHub* [online]. [vid. 2021-12-03]. Dostupné z: <https://answers.unrealengine.com/questions/231351/view.html>
- [29] OKITA, A. *Learning C# Programming with Unity 3D*. Second edition, Boca Raton, FL. USA: Routledge, 2019. 690 p. ISBN-13: 978-1138336810.
- [30] SUNG, K., SMITH, G. *Basic Math for Game Development with Unity 3D: A Beginner's Guide to Mathematical Foundations*. Bothel, WA. USA: Apress, 2019. 424 p. ISBN 978-1484254424.
- [31] LINOWES, J. *Unity Virtual Reality Projects: Learn Virtual Reality by developing more than 10 engaging projects with Unity 2018*. 2nd Edition, Birmingham, UK: Packt Publishing, 2018. 492 p. ISBN 978-1788478809.
- [32] *Oficiální Unity3D návody dostupné na <https://learn.unity.com/>*