



**ČESKÉ VYSOKÉ  
UČENÍ TECHNICKÉ  
V PRAZE**

**F3**

**Fakulta elektrotechnická  
Katedra počítačové grafiky a interakce**

**Bakalářská práce**

# **VR simulace poruch orientace v prostoru**

**Roland Fritsch**

**Otevřená informatika – Počítačové hry a grafika**

**Srpen 2021**

**Vedoucí práce: Ing. David Sedláček, Ph.D.**



## Poděkování / Prohlášení

Chtěl bych poděkovat svému panu vedoucímu za vedení v této bakalářské práci. Dále všem kantorům, kteří mě během studia učili a já v této práci použil znalosti z jejich předmětů. A v neposlední řadě bych rád poděkoval své rodině, která mi poskytla prostor a vhodné pracovní podmínky v naší domácnosti během současné epidemiologické situace.

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze dne 13. 8. 2021

.....

## Abstrakt / Abstract

Tato práce se zabývá realizací poruch orientace v prostoru ve virtuální realitě. Cílem je uživateli přinést balíček nástrojů v herním engine Unity, pomocí kterého bude schopen vytvořit simulace konkrétních poruch. Simulace by následně měly pomoci lidem, kteří pracují s pacienty s danými omezeními, aby si lépe dokázali poruchy představit. Práce v rámci analýzy zkoumá několik metod, díky kterým by mělo být možné poruchy simulovat ve VR. V dalších kapitolách podle nich navrhuje jednotlivé nástroje, které se následně snaží zrealizovat.

**Klíčová slova:** VR, virtuální realita, simulace, poruchy orientace v prostoru, topografická dezorientace, Unity, SteamVR, bakalářská práce.

This thesis deals with realization of spatial disorders of orientation in virtual reality. The main goal is to bring to a package of tools in Unity to a user by which they will be able to create simulations of specific disorders. Simulations should then help people who work with patients suffering with particular restrictions to visualize these disorders better. The chapter of analysis examines several methods which should help simulate these disorders in VR. In the next chapters the thesis designs and then implements individual tools in accordance with these methods.

**Keywords:** VR, virtual reality, simulation, spatial orientation disorders, topographical disorientation, Unity, SteamVR, bachelor thesis.

**Title translation:** VR simulation of orientation disorders in space (Bachelor thesis)

## / Obsah

<b>1 Úvod</b> .....	1
1.1 Motivace .....	1
<b>2 Analýza</b> .....	2
2.0.1 neeuklidovský prostor .....	2
2.0.2 Levels of detail .....	3
2.0.3 Redirected walking .....	4
2.0.4 Podobné práce .....	5
<b>3 Návrh řešení</b> .....	6
3.0.1 Návrhy simulací .....	8
<b>4 Implementace</b> .....	10
4.0.1 Simulace poruch .....	11
<b>5 Závěr</b> .....	16
<b>Literatura</b> .....	17
<b>A Zadání práce</b> .....	19
<b>B Zadání poruch orientace v prostoru</b> .....	20
<b>C Sestavení projektu</b> .....	23
<b>D Uživatelská příručka</b> .....	24
D.1 Ovládání .....	24
D.2 Předpřipravené modely .....	25
D.3 Skripty .....	26
D.4 Shadery .....	27

## Tabulky / Obrázky

<b>4.1.</b> Testování .....	15
<b>2.1.</b> Ukázka euklidovského a neuklidovského prostoru, Joshuabowman, , Pbroks13, ed. Noneuclid.png. In: Wikimedia Commons [online]. [cit. 2021-01-17]. Dostupné z: <a href="https://commons.wikimedia.org/w/index.php?curid=4373343">https://commons.wikimedia.org/w/index.php?curid=4373343</a> .....	2
<b>2.2.</b> Nekonečné schody, Sakurambo. Penrose stairs. In: Wikipedia [online]. [cit. 2021-01-17]. Dostupné z: <a href="https://commons.wikimedia.org/w/index.php?curid=794844">https://commons.wikimedia.org/w/index.php?curid=794844</a> .....	3
<b>2.3.</b> Princip použití různé úrovně detailu, Using LODs for distant objects [online]. In: . [cit. 2021-01-17]. Dostupné z: <a href="http://mkrus.free.fr/CG/LODS/xrds/">http://mkrus.free.fr/CG/LODS/xrds/</a> .....	4
<b>3.1.</b> Ukázka orlího pohledu ze série Assassins creed, RAF-FAELE, Rennan. Virtual Reality Immersive user interface for first person view games. In: Research Gate [online]. Berlin: ResearchGate, c2008-2021, 2017 [cit. 2021-01-17]. Dostupné z: <a href="https://www.researchgate.net/figure/Assassins-Creed-game-eagle-vision-72_fig24_318572588">https://www.researchgate.net/figure/Assassins-Creed-game-eagle-vision-72_fig24_318572588</a> .....	7
<b>4.1.</b> Ukázka zvýraznění modelu pomocí fresnel efektu .....	10
<b>4.2.</b> Ukázka pohyblivých ukazatelů.....	11
<b>4.3.</b> Mapa pro Landmark Agnosia. Červeně je zvýrazněn start, cíl a pozice portálů, žlutě trasa .....	12
<b>4.4.</b> Mapa pro Anterograde disorientation or topographical	

amnesia. Červeně je zvýrazněn start a cíl, pozice portálů je červeně a zeleně, žlutě je zvýrazněna cesta ..... 13

**4.5.** Mapa pro Egocentric disorientation. Růžovou plochou je zobrazena startovní pozice ..... 13

**4.6.** Mapa pro Heading disorientation. Uživatel musí obejít dům a najít model auta, který se přesune ..... 14

**4.7.** Ukázka aplikace hemiprostorového neglektu. V levé části nejsou vidět okna a dveře budov. V pravé ano. Ve středu je mizící auto ..... 14





# Kapitola 1

## Úvod

Virtuální realita (VR) nabízí nevídané možnosti a je úžasným nástrojem. Běžní lidé si VR většinou spojí s hraním her, možná promítáním filmů atp. Ve všech těchto využitích se uživatel může téměř doslova ponořit a vžít se do okamžiků, které se zásluhou VR kolem něj odehrávají. A právě kvůli vysoké míře imerze neboli vnoření se se VR nepoužívá jen v zábavním průmyslu, ale i v mnoha jiných odvětvích, kde je možné díky tomuto nástroji realizovat veliké množství simulací.

### 1.1 Motivace

Někteří lidé v naší společnosti trpí zhoršenou orientací v prostoru či přímo jejími poruchami. Aby lékař mohl pacientům s tímto typem omezení řádně pomoci, musí se dokázat vcítit do jejich problému a dobře si představit, s čím se potýkají. Jenže něco takového může být velice složité. Informace, které považují za zcela automatické, zde najednou nejsou. Je přece zcela zřejmé, že když půjdu proti psovi a zahnu pár metrů před ním doprava, a pokud pes mezitím neodešel jinam, tak se bude nacházet po mé levé straně. Například ale člověk s egocentrickou disorientací toto nejspíše určit nedokáže(více v příloze B). A právě tato práce by měla vést – k lepšímu pochopení, jak lidé s poruchami orientace v prostoru vnímají svět kolem sebe. Budu se snažit vytvořit nástroje, díky kterým bude možné dané poruchy co nejlépe simulovat.

Projekt bude koncipovaný jako balíček nástrojů v herním enginu Unity<sup>1</sup>.

---

<sup>1</sup> <https://unity.com/>

# Kapitola 2

## Analýza

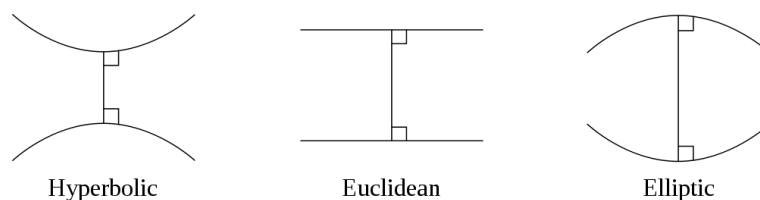
Dostal se přede mě zajímavý problém – virtuální realita se snaží, co nejlépe stimulovat naše smysly, abychom se mohli pohodlně ponořit do světa, který vytvořila kolem nás. Snaží se omezit diskomfortní pocity, jež může vyvolávat. Například různé formy dezorientace. Ale právě tu prostorovou se musím pokusit vyvolat.

V rámci simulace B se uživatel bude snažit dojít ze startu (bod A) do cíle cesty (bod B) na předpřipravené mapě, která bude upravena určitými nástroji, aby během tohoto průchodu měl uživatel pocit, že trpí daným omezením. Simulace bude mít tři části, tři průchody mapou:

- Poprvé bude uživateli představena cesta z bodu B do bodu A za pomoci nahrávky, která ho provede trasou.
- Při druhém průchodu bude mít za úkol se vrátit z bodu A do bodu B. Bude mít ideální podmínky pro orientaci, bez omezujících efektů a navíc budou zvýrazněny orientační body.
- Při posledním průchodu absolvuje stejnou trasu jako ve druhé fázi, ale tentokrát, místo zvýraznění klíčových aspektů cesty, budou aplikována omezení charakteristická pro danou poruchu.

Z popisu této simulace mi vystává jeden hlavní problém. Řekněme, že budeme mít třeba mapu města. Uživatel bude muset projít cestu, na různých křižovatkách zahýbat. Dejme tomu, že ve druhé fázi by bylo zvýrazněné zaparkované velké červené auto, za kterým by uživatel měl zahrnout doleva. Pro třetí průchod ho nechám nějak zmizet nebo ho nějakým způsobem pozměním (např. změna barvy, poznávací značky...). Vzhledem k tomu, že všechny tři průchody se odehrávají bezprostředně po sobě, tak s největší pravděpodobností většina uživatelů sice zaznamená chybějící orientační bod, ale i tak si nejspíše budou pamatovat, že někde v těch místech měli zahýbat doleva. Na myšlenku, jak „zdravého“ člověka v takovéto situaci lépe zmást, mě přivedl následující pojem.

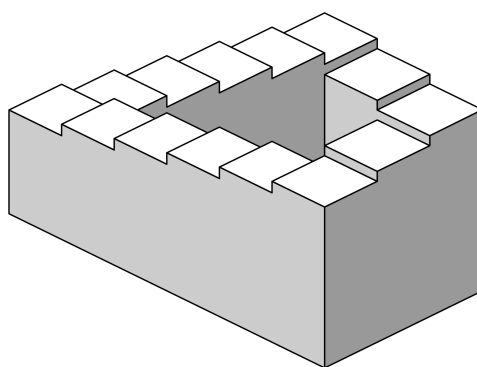
### 2.0.1 neeuklidovský prostor



**Obrázek 2.1.** Ukázka euklidovského a neeuklidovského prostoru, Joshuabowman, , Pbroks13, ed. Noneuclid.png. In: Wikimedia Commons [online]. [cit. 2021-01-17]. Dostupné z: <https://commons.wikimedia.org/w/index.php?curid=4373343>

„Eukleidovský prostor je matematický výraz pro člověku nejbližší, intuitivní představu prostoru.“ [1] Platí zde určité vlastnosti, např. že dvě rovnoběžné přímky se nikdy neprotnou, nebo že všechny úhly trojúhelníku mají dohromady vždy 180 stupňů (viz. 2.1).

A právě toto v neeuklidovském prostoru platit nemusí. Prostor může být nějak zakřivený, zdánlivě stejné vzdálenosti se mohou lišit. V důsledku toho se kružnice může jevit jako čtverec, tunel se může zdát velice krátký, ale při průchodu je značně delší, nebo schody mohou samy na sebe navazovat (obr. 2.2). Když se mi také nějak podaří „zakřivit“ realitu, mělo by být jednodušší uživatele zmást, kde se v prostoru nachází. Toto „zakřivení“ ale musí být dostatečně nenápadné, aby uživatel nabyl dojmu, že se skutečně spletl, a nenabyl dojmu, že aplikace „podvádí“. Např. když je přede mnou dům, já zavřu oči a po otevření zde již dům není, tak mi nejspíše dojde, že zde něco není v pořádku. V současné době již existuje několik her, které dokáží hráče patřičně zmást<sup>1</sup>. Používají k tomu perspektivu, pozici hráče vůči objektům ve scéně, více různých kamer, mezi kterými se uživateli přepíná obraz, jež právě vidí, či hráčovo náhlé přemístění v prostoru na jiné místo, aniž by změnu pozice uživatel zaznamenal. Některé z těchto technik se pokusím aplikovat ve své práci.



**Obrázek 2.2.** Nekonečné schody, Sakurambo. Penrose stairs. In: Wikipedia [online]. [cit. 2021-01-17]. Dostupné z: <https://commons.wikimedia.org/w/index.php?curid=794844>

Některé poruchy se projevují opomíjením detailů a podobné objekty mohou splývat (viz příloha B). Je několik způsobů jak objekt pozměnit, v mém případě zjednodušit. Lze změnit tvar objektu nebo jeho vizuální stránku. V Unity se objekty nejčastěji skládají z trojúhelníkové sítě (mesh<sup>2</sup>), jež je tvořena ploškami těchto trojúhelníků, hranami, které je vymezují, a vrcholy, jež udávají jejich polohu. Za pomoci vertex shaderů<sup>3</sup> lze měnit vlastnosti této meshe. Dále jsou objekty reprezentovány materiálem. Jeho vlastnosti ovlivňují, jak se objekt např. chová vůči světlu, jestli ho odráží nebo pohlcuje, zda je nějakým způsobem průhledný, jestli vrhá stín, a především jakou má barvu. Většinu těchto vlastností lze buď nastavit manuálně anebo mohou být reprezentovány za pomoci textury. Nejčastěji se používají dvourozměrné textury, které je možné chápat jako obrázek, který se na objektu zobrazí. Všechny tyto vlastnosti lze ovlivnit konkrétním shaderem, úpravou textury či materiálu, anebo jeho jiných vlastností.

## ■ 2.0.2 Levels of detail

Inspiraci pro řešení problémů pro úpravu objektu bych mohl nalézt v technice *levels of detail* (LOD), kde se pro lepší frekvenci renderování scény dle určitých pravidel mění úroveň detailu vykreslovaných objektů. Když mám objekt, o kterém vím, že se vzdálenost mezi ním a kamerou bude měnit, tak pro případy, kdy je od kamery dále,

<sup>1</sup> video s příklady neeuklidovských her: <https://www.youtube.com/watch?v=1FEIUcXCEvI>

<sup>2</sup> Mesh v Unity: <https://docs.unity3d.com/ScriptReference/Mesh.html>

<sup>3</sup> <https://en.wikipedia.org/wiki/Shader>



**Obrázek 2.3.** Princip použití různé úrovně detailu, Using LODs for distant objects [online].  
In: . [cit. 2021-01-17]. Dostupné z: <http://mkrus.free.fr/CG/LODS/xrds/>

nepotřebuji použít tak detailní model. Bude stačit model s menším množstvím ploch, protože na danou vzdálenost si nejspíše nevšimnu vzniklých nepřesností (viz. 2.3). LOD se dá rozdělit na tři typy:

- **Discrete levels of detail** obsahuje několik různě složitých modelů, mezi kterými se přepíná podle předdefinované vzdálenosti. Může se ale stát, že přechod je příliš viditelný a nastane tzv. *pop up efekt*, náhle „vystane“ daný objekt z pozadí. Lidé mohou být na tyto náhlé změny citliví. Při použití diskrétního LOD by se měly vzdálenosti vhodně nastavit tak, aby tento efekt byl minimální.
- **Continuos levels of detail** využívá nějaké předdefinované struktury, díky kterým je možné vyprodukovat jakoukoliv úroveň detailu během renderování. Model nikdy nemá více ploch, než je třeba, a přechod je mnohem plynulejší – minimalizuje se tím pop up efekt.

### ■ 2.0.3 Redirected walking

Tato technika umožňuje uživateli procházet virtuální prostor větší než je skutečný fyzický prostor kolem něj. Dosahuje se toho za pomoci rotací a translací, jež se liší svou velikostí ve virtuálním a skutečném prostoru. Získaný rozdíl, např. v rotaci uživatele, může napomoci k jeho natočení zpět do středu fyzického prostoru, aby z něj omylem nevyšel. Tyto rozdíly (např. při dotáčení či nedotáčení uživatele) ale nesmí být znatelné [7].

Metoda využívá faktu, že lidé (i roboti) v prostředí bez vizuálních nápověd nejsou schopni udržet rovný směr chůze a začnou se stáčet a pohybovat v kruzích. Ve skutečném světě totiž nelze dosáhnout perfektní symetrie (každý člověk má jinak silnou levou a pravou nohu, jednu lehce delší, rozdílné sensorické vnímání). Jedna strana bude mít vždy tendenci dominovat a nakonec bude člověk chodit v kruzích, aniž by si to uvědomil [8].

V knize *Virtual reality* [8]. se píše, že na dosažení daného efektu je třeba čtvercová plocha se stranami dlouhými aspoň třicet metrů. Ve článku z Tasmánské univerzity [9]. jsem se dočetl o testování několika algoritmů pro redirected walking, které se snaží uživatele udržet ve vyznačeném fyzickém prostoru a, při vystoupení z něj, ho co nejlépe navést zpět. Při použití modifikovaného algoritmu *Small circle* průměrně stačila kruhová plocha o průměru 17,3 m. Byly zde ale předem známé akce uživatele. Při náhodných akcí uživatele se s jinými algoritmy dostali průměrně na kruhovou plochu o průměru

24,5 resp. 28,4 m. Bohužel já nebo budoucí uživatelé mého balíčku nejspíše nebudeme mít k dispozici dostatečně velkou plochu a tak jsem se detaily těchto algoritmů dále nezabýval. Dále se zde píše, že lze aplikovat rotaci až o 5 stupňů za sekundu aniž by si toho uživatel všimnul, což jistě vyzkouším.

#### ■ 2.0.4 Podobné práce

Najít práce, které by se zabývaly podobnou tematikou (Simulací poruch orientace v prostoru, aby zdravím lidem navodili pocit, že danou poruchou trpí.) pro mě nebylo jednoduché. Většina prací se snažila zjistit zda VR může pomoci lidem trpícími danými omezeními při rehabilitaci.

##### **Virtual reality in neurologic rehabilitation of spatial disorientation**

V této studii 22 participantů (11 trpělo poruchou orientace v prostoru. Průměrný věk všech účastníků byl 66 let.) mělo za úkol během pěti sezení projít různé trasy ve virtuálním prostředí nejprve s instrukcemi kdy kam mají zahrnout a poté sami. Když udělali chybu, byli upozorněni a opraveni. Návlek se opakoval dokud nebyli schopni trasu projít bez problémů. Obě skupiny zaznamenaly zlepšení svých orientačních schopností dle standartních neuropsychologických testů. Více zde <sup>1</sup>

##### **Effect of Spatial Disorientation in a Virtual Environment on Gait and Vital Features in Patients with Dementia: Pilot Single-Blind Randomized Control Trial**

Zde se snažili zjistit zda jejich systém dokáže rozpoznat poruchu orientace v prostoru u uživatele. Měli zde tři skupiny participantů. První dvě skupiny měli průměrný věk přes 70 let s tím rozdílem, že členové první skupiny trpěli demencí. Třetí skupina byla složena z lidí do 40 let věku a právě přístup k nim mě zaujal. Všechny tři skupiny měli za úkol projít trasu, nejprve s vedením a nápovědami a následně bez nich. Třetí skupina ale to ale měla stížena pozměněním orientačních bodů, jejich přesunutím, či přidáním nových objektů. Něčeho podobného nejspíše využiji při své simulaci. Více zde <sup>2</sup>

<sup>1</sup> [https://jneuroengrehab.biomedcentral.com/articles/10.1186/1743-0003-10-17?fbclid=IwAR0RbSKSgMx0qwYcSnv\\_NvRvkkxgsyw9IPZihErfrUX5mQFjYcomH6ZyVbY](https://jneuroengrehab.biomedcentral.com/articles/10.1186/1743-0003-10-17?fbclid=IwAR0RbSKSgMx0qwYcSnv_NvRvkkxgsyw9IPZihErfrUX5mQFjYcomH6ZyVbY)

<sup>2</sup> <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7582144/?fbclid=IwAR3s6BBmrBaXix3RWOCpWRK5kZj-EBBm0cOw1hwifwnGNXCcVH71TWLgoco>

# Kapitola 3

## Návrh řešení

Pro jednoduchost a i na doporučení svého vedoucího jsem se rozhodl nejprve simulaci naimplementovat jako hru z pohledu první osoby (FPS) a následně, po odzkoušení základních mechanik v této formě, ji převést do virtuální reality. Na počítači se podobně jako většina FPS aplikací bude ovládat za pomoci myši a klávesnice, pro VR použiji HTC Vive. V Unity budu používat Universal render pipeline (URP). Oproti „klasické“ nabízí možnost používat Shader graph, což je mnohem uživatelsky přívětivější způsob tvoření shaderů než jejich psaní v kódu. Navíc by se časem URP měla stát běžně používanou a podporovanou pipelineou v Unity. High definition render pipeline jsem se rozhodl nepoužít, protože, i když také nabízí možnost tvořit shadery pomocí grafů, cílí na projekty ve vysoké grafické kvalitě na nejnovějších výkonných počítačích a konzolách, což můj domácí notebook nesplňuje. Zároveň mi přijde, že by bylo s HDRP složitější pracovat než s URP. K podpoře VR budu používat balíček SteamVR. V této kapitole budu za uživatele uvažovat osobu, jež pracuje s mým balíčkem, ne osobu, která bude používat finální aplikaci.

**První průchod** z bodu B do bodu A budu realizovat za pomoci komponenty Nav-Mesh [2]. Uživatel na mapu umístí objekt hráče a *empty objects*, kterými vyznačí trasu. Poslední tako určený bod se považuje za cíl. Objekt hráče bude obsahovat komponentu NavMesh agent, díky které najde cestu ze své výchozí pozice přes všechny body až do cíle. Toto bude lepší používat v FPS verzi, protože ve VR verzi by to mohlo způsobit motion sickness. Alternativou bude možnost dorazit do cíle za pomoci šipek a ukazatelů.

**Druhý průchod** začne krátkým zatměním, během kterého se načtou všechny změny. Ve SteamVR by na to měla být třída SteamVR\_Fade. Zvýrazní se:

- Orientační body, přesněji objekty. Jejich materiály budou mít shader se zabudovaným fresnel efektem, díky kterému by měli zářit hrany objektů, a tím by mělo dojít k jejich zvýraznění. Pokusím se, aby záře hran navíc pulzovala, čímž by se mělo docílit ještě většího výsledného efektu.
- Směr cesty a trochu i geometrie budov, podobně jako v různých stopařských módech v některých hrách (viz 3.1).
- Zvýraznění cíle za pomoci nějakého ukazatele, či šipky, jež bude vždy čelem k hráči.

Do **posledního průchodu**, kdy se hráč opět přesune do bodu A, se nepřejde automaticky, ale až po zmáčknutí klávesy, aby se předešlo neočekávanému přesunu, který by mohl vyvolat diskomfort. Opět bude použit fading efekt ze SteamVR. K dosažení dezorientace bude moct uživatel použít dva hlavní způsoby:

- Modifikace klíčových objektů (těch, jež byly zvýrazněny ve fázi 2). Bude zde skript, ve kterém půjde nastavit konkrétní změnu. Verze, že by uživatel jen nahrál model a balíček by nějak dokázal rozeznat jakou změnu aplikovat a jak ji aplikovat, mi připadá, že by značně omezila šíři přijatelného vstupu od uživatele a snížila by variabilitu, co se volby modifikace týče. Veliký podíl na vizuální stránce modelu má jeho materiál, a tak zde bude několik jednoduchých možností, jak ho modifikovat. Bude možné nahrát jinou barvu, texturu či rovnou celý materiál. Dále by bylo možné



**Obrázek 3.1.** Ukázka orlího pohledu ze série Assassins creed, RAFFAELE, Rennan. Virtual Reality Immersive user interface for first person view games. In: Research Gate [online]. Berlin: ResearchGate, c2008-2021, 2017 [cit. 2021-01-17]. Dostupné z: [https://www.researchgate.net/figure/Assassins-Creed-game-eagle-vision-72\\_fig24\\_318572588](https://www.researchgate.net/figure/Assassins-Creed-game-eagle-vision-72_fig24_318572588)

v rámci samotného objektu změnit části jeho transform komponenty, přesněji pozici, rotaci nebo velikost. Poslední možnou změnou by bylo udělat objekt neaktivním (neviditelným), a případně místo něj zviditelnit jiný objekt, který byl do té doby neaktivní.

- Přemístění uživatele, aniž by si toho všimnul, do jiné části mapy za pomoci **teleportu**. Tím dosáhnou neeuklidovského prostoru a mělo by to bezpečně vyústit ve zmatení hráče. Portály [3] budou duální. Tzn. když vejdu do portálu č. 1 a budu přemístěn do oblasti portálu č. 2, tak při následném průchodu budu přenesen zpět do oblasti portálu 1 a neobjevím se v nějaké nové oblasti s portálem 3. Budou dva typy portálu:
  - Naivní – ten pouze po průchodu hráče určitou oblastí zahájí teleportaci na nové místo. To znamená, že celé okolí obou portálů musí být zcela identické, aby hráč nezaznamenal změnu polohy.
  - Normální – portál krom plochy, jež bude sloužit jako collider pro indikaci přítomnosti hráče a následné vyhodnocení možného přesunu, bude mít i druhou plochu, na kterou se bude promítat prostor, do něhož se má hráč přesunout. Promítací plocha bude obsahovat render texture. Obraz na ní promítaný bude snímat kamera, jež bude mít vůči výstupnímu portálu stejnou pozici a rotaci jako hráčova kamera vůči vstupnímu portálu. Zároveň bude potřeba na renderovací plochu nepromítat celý obraz snímaný kamerou, ale pouze část, jež může hráč skrze portál vidět. To by mohlo být realizovatelné pomocí shaderu a správné změny UV souřadnic. Každý portál bude mít svou vlastní kameru. U tohoto typu teleportu bude stačit, aby byla identická pouze oblast, ve které se budou všechny plochy obou portálů nacházet. Zbytek by měl být bez problému promítnut na renderovací plochu.

V obou případech bude potřeba, aby obě oblasti zaujímali stejnou polohu vůči zdrojům světla kvůli dopadu stínů. Zároveň pro použití teleportů bude třeba uzavřenější scény, omezující pohyb jen určitými směry (bludiště, městské ulice, chodby, tunely...) bez větších otevřených ploch, umožňujících poměrně volný pohyb (parky, lesy...), aby se nestalo, že se hráči podaří teleport obejít.

V simulaci bude možné umístit popisky, které budou mít za úkol poradit hráči, v jaké fázi se zrovna nachází a co má dělat. V FPS verzi se hráč bude moct pohybovat pomocí kláves WASD a myši. Joystick bude udávat směr pohybu vůči aktuální poloze levého

controlleru. To znamená, kam směřuje ovladač tam bude „předek“ hráče. To dovolí volné rozhlížení. Toto řešení může způsobovat diskomfort lidem méně zvyklým na VR, takže uvidím zda ho nakonec zahrnu ve finální verzi. Spíše použiji možnost pohybu pomocí teleportace na místo dopadu paprsku. Pro pohodlnější používání balíčku uživatelem se pokusím používat třídy dědicí od třídy EditorWindow, která umožňuje změnit vizuální stránku a chování zadávání proměnných v inspektoru v Unity.

### ■ 3.0.1 Návrhy simulací

Zde vycházím hlavně z přílohy B, kde jsou popsány jednotlivé poruchy.

- **Landmark agnosia** – Simulace se bude odehrávat ve městě, kde na každé křižovatce budou rozdílné orientační objekty (budova, auto...), aby si uživatel lépe pamatoval jak odbočil ve druhé fázi bez omezení. Ve třetí fázi podobné orientační objekty splynou dohromady (stejná textura, barva, nahrazení různých objektů stejného typu jedním...), aby pro něj bylo těžší se rozhodovat kudy se vydat. Pro případ, že by si uživatel nezávisle na klíčových objektech pamatoval trasu, zde bude teleport, jenž ho přenesení do jiné části trasy, aby se narušila posloupnost zapamatovaných instrukcí kudy se vydat.
- **Anterograde disorientation or topographical amnesia** – Tuto simulaci bych chtěl realizovat obdobně jako první s tím rozdílem, že by se uživatel při chůzi ve třetí fázi mohl mírně natáčet nebo když se rozejde se nárazově otočit třeba o 90 nebo 180 stupňů, aby se vydal špatným směrem. Zároveň by zde bylo více portálů, takže by se uživatel v nich při průchodu mohl zacyklit a nikdy nedokázat dosáhnout cíle.
- **Egocentric disorientation** – Zde by simulace začínala v reálném světě (byla by pouze dostupná ve VR verzi). Uživatel by si zapamatoval svou pozici vůči nějakému předmětu, tak aby ho viděl (např. by stál přímo před monitorem, oknem atp.). Teď ten předmět vidí a ví jakým směrem se od něj nachází. Nasadí si headset a předmět momentálně nevidí. Musí projít určitou trasu a ukázat, kde si myslí že se předmět vzhledem k němu nachází. Bez omezení stále ve stejném směru, ale ve třetí fázi bych chtěl využít poznatku o metodě redirected walkinga tím uživatele otočit, aniž by si toho všimnul. První způsob jaký mne napadá je, že by měl ujít určitou vzdálenost v jednom směru (jak ve virtuálním, tak reálném světě) a následně se otočit a vrátit se zpět na původní pozici. Během pohybu by byl konstantně otáčen o pár stupňů, čímž by se mělo dosáhnout následného zmatení. Druhý způsob bude nejspíše jednodušší. Místo reálného pohybu by se uživatel pohyboval pomocí teleportace a při každém přesunu by byl mírně natočen. Procházel by bludiště, kde by bylo mnoho zatáček, takže by se musel reálně otáčet, čímž by se vzniklý rozdíl v rotacích měl vyrovnat. V obou případech bude problém s kabelem, který vede z headsetu a v prvním navíc i dostupný prostor. Obojí budu muset vyzkoušet a otestovat.
- **Heading disorientation** – U této poruchy mám problém ji dobře odlišit od předešlé, ale simulace nejspíše bude spočívat v průchodu trasy, kde bude důležitá vzájemná poloha objektů a při aplikaci omezení bude jejich vzájemná poloha změněna.
- **Hemiprostorový neglekt** – U této simulace nebudou přímo omezení, která by znemožnila průchod správné trasy. Budou opomíjeny detaily na jedné straně. Objekty, které budu chtít takto ovlivnit budou mít materiál se speciálním shaderem. Ten bude zjišťovat souřadnice na obrazovce každého fragmentu a pokud se bude nacházet v opomíjené polovině bude pozměněn. Buď bude možné použít jinou barvu, texturu, normálovou mapu pro fragmenty v dané části obrazovky, anebo se změní jeho alpha složka, aby byla menší nebo rovna hodnotě v proměně `alpha clipping`, čímž se fragment zahodí. Mezi polovinami obrazovky budu v určitém rozmezí hodnoty



---

interpolovat, abych dosáhnul plynulého přechodu. Ve VR verzi, místo rozdělení obrazovky na dvě části, zkusím oku se „zdravou“ hemisférou zobrazovat správný svět a oku s „poškozenou“ hemisférou svět pozměněný.

## Kapitola 4

### Implementace

Nejprve jsem se většinu návrhů pokusil naprogramovat a odzkoušet ve FPS verzi, následně je převést do VR. Zde se ale objevilo několik problémů. Většinu jsem ale dokázal vyřešit či obejít. Zároveň jsem se potýkal i s občasnými konflikty mezi URP a SteamVR. Pro hráče jsem zvolil komponentu CharacterController a ve VR verzi používám prefab ze SteamVR CameraRig s různými úpravami. Simulaci řídí třída RouteController, která přepíná mezi fázemi, a kontroluje makra potřebná pro ostatní objekty ve hře.

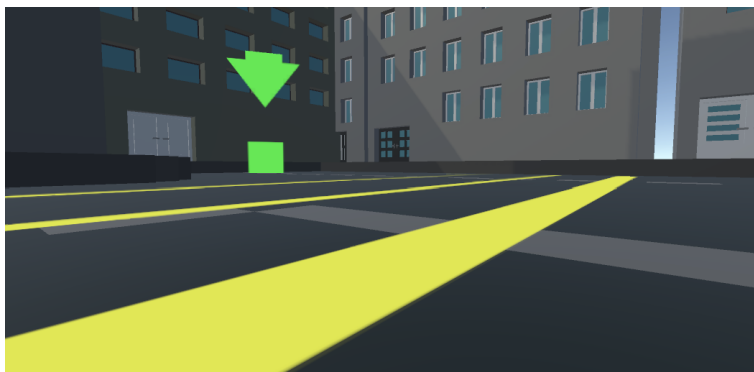
**První fáze** byla vesměs bez problémů. NavMesh agent funguje jak má. Přidal jsem možnost místo NavMesh agenta použít podobně jako v druhé fázi ukazatele pro pohodlnější ovládání ve VR.



**Obrázek 4.1.** Ukázka zvýraznění modelu pomocí fresnel efektu

Ve **druhé fázi** jsem použil shader graph pro tvorbu pbr shaderu, který zvýrazní objekty fresnel efektem (viz 4.1). Tento shader obsahuje vlastnosti pro běžné nastavení materiálu (base color, metallic...). Původně se měl fresnel efekt v dané fázi vypnout či zapnout, ale nastavování proměnné v shaderu mi v tomto případě pořádně nefungovalo. Nakonec tedy místo zapínání efektu měním materiály s rozdílnými shadery. Pro zvýraznění cesty a cíle, kudy se má hráč vydat, jsem použil objekt plane (plochu) se speciálním shaderem, který v závislosti na čase mění UV souřadnici  $y$  vložené textury (v mém případě např. šipka). Textura je černobílá a používá se pro transparentnost. Výsledný efekt vypadá např. jako plovoucí čáry prostorem indikující směr 4.2.

Ve **třetí fázi** jsem pro modifikaci objektu napsal skript, který po přidání jako komponenty k tomuto objektu, umožní vybrat a nastavit jednu z možností (změna barvy, polohy...). Zde jsem použil třídu EditorWindow pro vytvoření vlastní podoby, aby se uživateli vždy zobrazovala jen vybraná možnost. Největší problém, který jsem ale nedokázal vyřešit, jsem měl s promítáním prostoru za portálem na renderovací plochu. V FPS vše fungovalo, ale při přechodu na VR, kvůli jinému renderování jsem nedokázal docílit stejného výsledku. Původně pro FPS verzi jsem ve vertex shaderu díky funkci `UnityObjectToClipPos()` a následnému použití funkce `ComputeScreenPos()`



**Obrázek 4.2.** Ukázka pohyblivých ukazatelů

získal UV souřadnice renderovací plochy na obrazovce. Následně ve fragment shaderu jsem souřadnice  $x$  a  $y$  vydělil souřadnicí  $w$  (která značí vzdálenost od objektu), čímž jsem je „promítnul na obrazovku“ a následně je použil jako souřadnice pro výslednou texturu. Stejného efektu jsem dosáhnul i v shader graphu za použití Screen position node.

Ve VR se obraz renderuje [5] zvlášť pro každé oko, buďto do dvou samostatných textur (multipass stereo) nebo do jedné vedle sebe (singlepass stereo), což je značně výhodnější, protože se stejný proces nemusí dělat zvlášť pro každé oko. Ani s jedním přístupem se mi nepodařilo dosáhnout kýženého výsledku. Pro singlepass by pro to dle dokumentace měly existovat různé funkce, ale asi mi unikají nějaké souvislosti, kvůli kterým jsem je nejspíše použil špatně. I když se mi například podařilo sjednotit pohled očí za pomoci `unity_StereoEyeIndex` (v multipassu), i tak vyrenderovaný obraz neseděl. Při rotaci hlavy doleva či doprava se i promítaná textura posouvala (více než by měla). Zkoušel jsem různé korekce UV souřadnic. Daný nežádáný posun jsem se např. snažil v shaderu odečítat, ale výsledek zůstal stejný. Z toho důvodu v balíčku je k dispozici jak normální portál pro FPS, tak naivní pro VR, který nevytváří obraz, ale pouze uživatele přenese. Proto bohužel musí být obě destinace totožné.

Další problém jsem měl se třídou `SteamVR_Fade`. Ukázalo se, že není kompatibilní s URP. Problém jsem obešel za pomoci černě zbarvené UI komponenty `image`, která pomocí funkce `CrossFadeAlpha()` [4] mění transparentnost komponenty. Canvas s touto komponentou je vždy umístěn před kamerou `CameraRigu`.

#### ■ 4.0.1 Simulace poruch

- **Landmark agnosia** – Zde jsem použil mapu města 4.3, rozděleného do čtvercových bloků. Barevně jsou zde odlišeny důležité budovy. Ty, kde uživatel bude zahybat jsou navíc ve druhé fázi zvýrazněny. Ve třetí budovy stejného typu barevně splnou. Navíc je zde jsou dvě auta, podle kterých se může orientovat uživatel, ale ta jsou ve třetí fázi přesunuta na jiné místo. Během cesty hráč projde portálem, který ho přesune o blok blíže k cíli. Podle budov se nyní orientovat nedá, takže pokud by si uživatel pamatoval naučenou trasu, tohle by ho mělo zmást.
- **Anterograde disorientation or topographical amnesia** – Zde jsem použil stejnou mapu, ale přidal jsem navíc jednu dvojici portálů 4.4. Cesta se navržená tak, že uživatel ve třetí fázi skrze ně několikrát projde, což by ho mělo dezorientovat.
- **Egocentric disorientation** – 4.5 Pohyb pomocí normální chůze jsem zavrhnul vzhledem k omezenému reálnému prostoru (při testování jsem často do něčeho narazil). Navíc někdy byla limitací i délka kabelu. Uživatel se tedy přesunuje pomocí telepor-



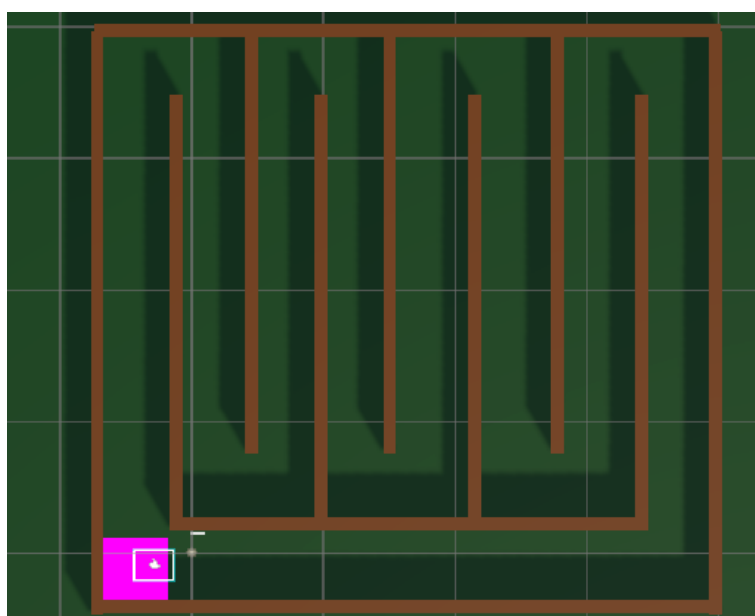
**Obrázek 4.3.** Mapa pro Landmark Agnosia. Červeně je zvýrazněn start, cíl a pozice portálů, žlutě trasa

tace. Mapa je jednosměrné bludiště s mnoha zatáčkama, které hráče přivede zase zpět na začátek. Místo tří fází simulace jsem použil jen dvě. Uživatel si musí zapamatovat svou reálnou polohu. Cestu projde bez omezení. Po kontrole polohy, že je stále stejná, se přepne do druhé fáze. Zde při každém přesunu se natočí o 3 stupně. Zkoušel jsem až pět, ale to již bylo moc výrazné. Uživatelova reakce je, že jen mírně natočí hlavu. Jakmile dorazí k zatáčce, tak při otáčení rotaci opět dorovná. Trasu se dá projít až na 30 přesunů, což činí rozdíl 90 stupňů. Při testování pocitový rozdíl oproti výchozí poloze (např. vzhledem ke kabelu headsetu) byl zhruba 15-25 stupňů, ale skutečný rozdíl oproti výchozí pozici byl až 3x větší. Nicméně je lepší během průchodu si kabel nadzvedávat, aby se do něj člověk nezamotal.

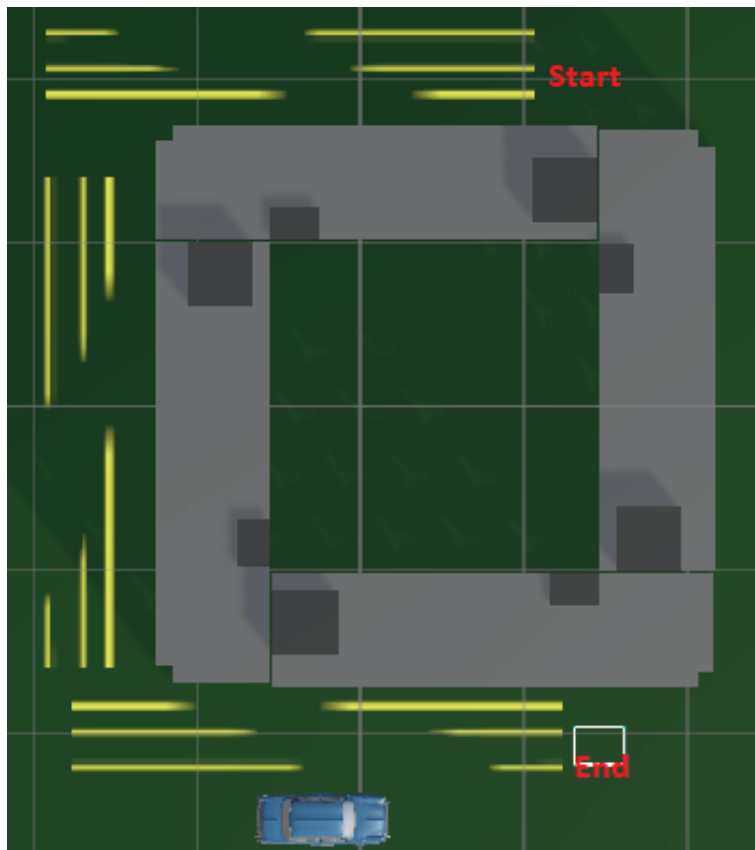
- **Heading disorientation** – 4.6 Zde jsem měl asi největší problém, jak danou poruchu simulovat. Nakonec jsem použil mapu s budovou, kterou je třeba obejít. Když se třikrát zahne za roh, nachází se zde auto. Následně je ale umístěno o roh dál, což může vyvolat jisté zmatení. S touto simulací jsem nejméně spokojen.
- **Hemiprostorový neglekt** – Tady jsem opět použil mapu města, tentokrát se dvěma bloky. Kolem nich projíždějí auta. K jejich pohybu jsem použil kvadratické beziérové křivky. První dvě fáze jsou stejné jako v předešlých simulacích. Ve třetí fázi k nějaké změně, která by mohla ztížit dokončení cesty přímo nedojde, ale napsal jsem shader, který omezí vnímání jedné strany. Buď (a jak jsem to použil i zde) na straně, kde působí omezení, není daný objekt vůbec vidět, anebo se zde zobrazí jiná textura (při použití detailní textury by se zobrazovala méně detailní). To samé lze nastavit



**Obrázek 4.4.** Mapa pro Anterograde disorientation or topographical amnesia. Červeně je zvýrazněn start a cíl, pozice portálů je červeně a zeleně, žlutě je zvýrazněna cesta



**Obrázek 4.5.** Mapa pro Egocentric disorientation. Růžovou plochou je zobrazena startovní pozice



**Obrázek 4.6.** Mapa pro Heading disorientation. Uživatel musí obejít dům a najít model auta, který se přesune



**Obrázek 4.7.** Ukázka aplikace hemiprostorového neglektu. V levé části nejsou vidět okna a dveře budov. V pravé ano. Ve středu je mizící auto

i pro normálovou mapu a metalickou texturu. Ve VR jsem místo toho chtěl renderovat rozdílný obraz pro každé oko, ale to nemělo dobrý výsledek. Při testování jsem spíše vnímal obraz, který vidělo mé dominantní oko a způsobovalo to bolest hlavy. Takže jsem to zavrhnul a udělal stejně jako v FPS verzi. Na této mapě se konkrétně zneviditelnují okna a dveře budov a projíždějící auta 4.7.

**Testování** bohužel za současných podmínek bylo poněkud komplikované. FPS verzi jsem přeposlal několika svým přátelům a rodiným příslušníkům. VR verzi jsem zkoušel jen sám v budově školy. Simulace zkoušelo celkem 8 lidí. Ptal jsem se jich zda našli

porucha	počet co našel cíl	počet co pocítoval dezorientaci
Landmark agnosia	7	8
topographical amnesia	5	8
Heading disorientation	8	2
Hemiprostorový neglekt	8	0

**Tabulka 4.1.** Testování jednotlivých poruch v FPS verzi.

cíl (dokud to nevzdali) a zda se cítili být dezorientováni 4.1. Egocentric disorientation jsem testoval pouze sám.

## Kapitola 5

### Závěr

Účelem tohoto projektu bylo pokusit se vytvořit nástroje k nasimulování několika poruch orientace v prostoru (více v B) ve virtuální realitě. Po dohodě s vedoucím (a protože jsem si to již takhle částečně připravoval, kvůli práci z domova) jsem se to snažil udělat i ve verzi pro počítač. Účelem nebylo přímo vytvořit přímo aplikaci, ale uživateli poskytnout nástroje, se kterými by ji mělo být možné vytvořit. Hodně záleží jak případný uživatel se těchto nástrojů chopí a hlavně jak si danou simulaci naplánuje a připraví. Myslím si, že nejvíce problémů jsem měl s *Heading disorientation*, ale naopak jsem celkem spokojen např. se simulací *Hemiprostorového neglektu*. Simulace jsou ceckem jednoduché, ale věřím, že jsem na nich zvládnul ukázat své nástroje.

Práci jsem bohužel nedokázal věnovat tolik času a péče, abych s ní byl zcela spokojen a ani má komunikace s vedoucím nebyla zcela adekvátní. Na druhou stranu jsem si podle mě vybral sice zajímavé téma, ale zároveň poněkud složitější než jsem čekal. Pokud má práce bude přijata, budu uvažovat zda ji v rámci diplomové práce nerozšířit o propracovanější simulace a nedostat ji do stavu, s jakým bych byl spokojen.



## Literatura

- [1] Eukleidovský prostor. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2021-01-16]. Dostupné z: [https://cs.wikipedia.org/wiki/Eukleidovsk%C3%BD\\_prostor](https://cs.wikipedia.org/wiki/Eukleidovsk%C3%BD_prostor)
- [2] Navigation Mesh. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2021-01-16]. Dostupné z: [https://en.wikipedia.org/wiki/Navigation\\_mesh](https://en.wikipedia.org/wiki/Navigation_mesh)
- [3] Coding Adventure: Portals. In: Youtube [online]. [cit. 2021-01-16]. Dostupné z: <https://www.youtube.com/watch?v=cWpFZbjtSQg>
- [4] Graphic.CrossFadeAlpha. Unity Manual [online]. Unity Technologies, c2020 [cit. 2021-01-16]. Dostupné z: <https://docs.unity3d.com/2018.1/Documentation/ScriptReference/UI.Graphic.CrossFadeAlpha.html>
- [5] Single Pass Stereo rendering (Double-Wide rendering). Unity Manual [online]. Unity Technologies, c2020 [cit. 2021-01-16]. Dostupné z: <https://docs.unity3d.com/Manual/SinglePassStereoRendering.html>
- [6] Building a NavMesh. Unity Manual [online]. Unity Technologies, c2020 [cit. 2021-01-16]. Dostupné z: <https://docs.unity3d.com/Manual/nav-BuildingNavMesh.html>
- [7] JERALD, Jason. The VR Book: Human-Centered Design for Virtual Reality. 2015. New York: y. Association for Computing Machinery and Morgan & Claypool, 2015. ISBN 9781970001143.
- [8] LAVALLE, Steven M. Virtual Reality. Cambridge: Cambridge University Press, 2016.
- [9] FIELD, Tom a Peter VAMPLEW. Generalised Algorithms for Redirected Walking in Virtual Environments. Core [online]. Tasmania, 2004 [cit. 2021-8-12]. Dostupné z: [https://core.ac.uk/download/pdf/33303849.pdf?fbclid=IwAR2Kwf0ckKJb4fz0-86n\\_ni95SxFjGI158nFnbK--BWoZOV6T5nArkTpvZI](https://core.ac.uk/download/pdf/33303849.pdf?fbclid=IwAR2Kwf0ckKJb4fz0-86n_ni95SxFjGI158nFnbK--BWoZOV6T5nArkTpvZI)





## **Příloha A**

### **Zadání práce**

Seznamte se s poruchami orientace v prostoru a konzultujte jejich projevy s doc. Fajnerovou z Národního Institutu Duševního Zdraví (NUDZ). Navrhněte a implementujte knihovnu pro vývojové prostředí Unity, která umožní simulaci těchto poruch v obecné herní scéně. Předpokládejte, že knihovnu bude používat uživatel (programátor) znalý tvorby prostředí v Unity. Funkcionalitu knihovny demonstруйте v herním světě připomínajícím část města nebo vesnice, jejíž základní koncept konzultujte s vedoucím práce. Ukázkovou aplikaci zprovozněte jak v PC verzi (z pohledu první osoby), tak ve VR verzi. Obě verze demonstrační aplikace otestujte s uživateli a v případě příznivé epidemiologické situace také s odborníky z NUDZ.

## Příloha B

### Zadání poruch orientace v prostoru

**Topographical Disorientation (TD)**, also known as *Topographical agnosia (Topographical agnosia)*, is a cognitive disorder that results in the individual being unable to orient in one's surroundings due to some neurological cause, such as brain lesion (resulting from a stroke or part of a progressive illness, hemispatial neglect, dementia or Alzheimer's disease) or developmental disorder (Aguirre and D'Esposito M, 1999).

Ve všech případech by byl podstatou simulace následující postup (specifická podoba se bude lišit pro každý z níže popisovaných typů poruch):

- následování nějaké trasy (navigace participanta podle šipek nebo asi raději simulovaný průchod městem v podobě nahrávky)
- Následně probíhá simulace navigace od cílové pozice k výchozímu bodu za ideálních podmínek. Na zdůraznění toho, co zdravý subjekt k navigaci používá nebo by mohl použít se během pohybu na trase postupně zvýrazňují různorodé prvky (orientační body, geometrie ulic, směr odbočování vůči vlastní pozici hráče a podobně)
- Úlohou participanta je následně stejnou trasu (od cílové pozice k výchozímu bodu) zopakovat s nějakým typem omezení (daný typ omezení si vybírá z nabídky/ seznamu). Podoba prostředí musí být přizpůsobena typu poruchy, aby byla orientace v tomto prostředí adekvátně stížená, a simulovala tak průchod městem s nějakým prostorovým omezením (pokud to nelze dosáhnout, tak může být simulovaná navigace pouhým přehráním nahrávky průchodu městem, tak aby byla změna patrná a omezení dané poruchou jasně vystihnuté).

**Landmark agnosia** is characterised by the inability to use prominent, salient environmental features (such as landmarks) for the purposes of orientation, usually due to lesions in the lingual gyrus (Aguirre and D'Esposito M, 1999; Whiteley and Warrington, 1987). Patients with landmark agnosia can distinguish between classes of buildings, but are unable to identify specific buildings, such as their own house or famous landmarks. However, they are able to draw detailed maps and visualize places that were familiar to them before the illness. Nevertheless, they are able to navigate using spatial information and specific details of landmarks such as house number or its color.

Zde si představuji, že navigující osoba nebude schopna využívat orientační body (budovy, mosty, přírodní dominanty). To znamená, že když narazí na budovu/orientační bod podobného typu, tak se nedokáže podle něj zorientovat a může se splést. Přesto by měl být schopen využívat geometrii prostředí, takže např. u stejného typu budovy odbočí odpovídajícím směrem). V případě simulace by se tak mohlo třeba jednat o to, že procházíme prostředím, které obsahuje spoustu podobných/zaměnitelných budov (které byly původně něčím odlišitelné) a dalších prvků, které se v prostředí opakují a stěžují tak orientaci v prostoru a schopnost se rozhodnout na křižovatce dle těchto orientačních značek. Prostředí bude ochuzené o detaily (nebo se musí záměrně opakovat). Během simulované navigace se tedy zvýrazňují i chybné (nesprávné) budovy/objekty, které si osoba plete s těmi cílovými.

**Anterograde disorientation or topographical amnesia** is marked by the inability to orient in new environments, usually due to lesions in the parahippocampus and medial temporal lobe (Habib and Sirigu, 1987). These patients are able to navigate through and draw maps only of environments learned at least 6 months before the brain damage (Ross, 1980; Teng and Squire, 1999). This indicates that the hippocampus and surrounding structures of MTL are not needed for the retrieval of spatial maps learned prior to the injury; however, they are essential for the formation of long-term declarative memories, including spatial memories (Teng and Squire, 1999). The study by Bohbot et al. (1998) confirmed the role of the right hippocampus in visuo-spatial memory tasks (object location, Rey-Osterrieth Figure with and without delay) and the left hippocampus for verbal memory tasks (Rey Auditory Verbal Learning Task with delay). However, as only patients with lesions to the right parahippocampal cortex were impaired on the hidden goal task (MWM) with a 30 min delay, showing that parahippocampal cortex itself may play an important role in spatial memory.

Zde by se mělo jednat o naprostou neschopnost se zorientovat v novém prostředí a vrátit se na původní lokaci nebo zopakovat předchozí trasu díky narušené paměti. Zde asi opět lze simulovat buďto významným ochuzením a zmatením prostředí, ve kterém se orientovat nelze (podobnost lokací, zmatečnost doplňujících prvků, nejasné značení ulic s jejich přehazováním atd.). Prostě naprostá dezorientace, kde žádný prvek není rozpoznatelný anebo působí zmatečně.

**Egocentric disorientation** is defined as inability to represent the location of objects with respect to the own position (self) that usually occurs due to lesions in the posterior parietal lobe. While these patients are unable to accurately reach for visual objects or state the relationship between two objects (above, below, left, right, nearer or farther), they experience no difficulty recognizing or naming people or objects (Wilson et al., 2005). These patients are not able to point to locations of targets defined by visual, proprioceptive, or audio input. A frequent demonstration of this disability is that, although the patients can point to a visualized object, they are no longer able to do so with their eyes closed, or if the object is not visible (in distance).

Zde bude asi simulace složitější, jelikož je navigace přímo vázaná na egocentrický systém vlastního těla. Dokážu si ale představit, že by ve vzorové simulaci dostala osoba egocentrické instrukce, jak se pohybovat v prostředí, které neobsahuje orientační body (například park, les nebo bludiště) a tedy se může orientovat jenom dle pokynů odbočit vlevo, vpravo, jít rovně atd. Zde bude klíčové, aby během navigace byla vidět postava hráče (simulující pacienta) a během navigace se zdůraznil typ informace, kterou osoba nedokáže použít (třeba, že nedokáže říct, že se budova od něj nachází po pravé ruce a pak při návratu tedy logicky pom pravé ruce).

**Heading disorientation** is defined by the inability to represent direction of orientation with respect to external environmental cues. These patients are able to determine their location using landmarks, but are unable to determine which direction to proceed from those landmarks in order to reach their destination, usually due to lesions in the posterior cingulate cortex. They are also impaired in map drawing tasks and are unable to describe routes between familiar locations (Aguirre, D'Esposito M, 1999). Also focal brain damage to the right retrosplenial region due to a cerebral hemorrhage (described in three case studies) may cause a loss in sense of direction (Takahashi et al, 1997). These patients showed normal visual perception, were able to identify, determine and remember locations of visible objects, but were unable to recall direction from selective familiar landmarks.



## Příloha C

### Sestavení projektu

Pro správné fungování balíčku je potřeba Unity 2020.1.14f1 (64-bit). Při vytváření nového projektu v záložce *Templates* je nutné vybrat možnost *Universal render pipeline*. Dále by měl být v počítači program Steam s nainstalovaným SteamVR. Doporučuji používat HTC Vive. Po načtení projektu v Unity je nutné ujistit se, že v záložce *Window* → *Package Manager* jsou tyto balíčky<sup>1</sup>:

- Test Framework
- Timeline
- Unity Collaborate
- Unity UI
- Universal RP
- Visual Studio Editor
- Open VR XR Plugin

#### **V projektu použité balíčky**

- SteamVR Plugin
- City Voxel Pack<sup>2</sup>

---

<sup>1</sup> většina by se zde již měla nacházet

<sup>2</sup> použito na ukázkovou mapu

## Příloha D

### Uživatelská příručka

Všechny hlavní části balíčku jsou ve složkách, které se nacházejí v adresáři *Assets*. Balíček obsahuje pět scén ve složce (*Scenes* → *Demo*), kde je ukázáno použití jednotlivých komponent. Každá scéna se jmenuje iniciály poruchy, kterou představuje. Modely budov, silnic a chodníků pochází z balíčku *City Voxel Pack*. Dále je zde můj vlastní model auta s texturami a materiály.

Pro správně fungování, zejména první fáze simulace, je třeba sestavit **Navigation Mesh** [6]. Pomocí *Window* → *AI* → *Navigation* se otevře okno *Navigation*. Zde je potřeba v záložce *Object* vybrat všechny objekty, po kterých má být hráči umožněno se pohybovat (ve fázi 1) a nastavit atribut *Navigation Area* na *Walkable*. Potom vybrat všechny „překážky“, nastavit je na *Not Walkable* a navíc zaškrtnout možnost *Navigation Static*. Následně v záložce *Bake* zmáčknout *bake* a chvíli počkat. Ve složce s aktuální scénou by se měl vytvořit adresář s vygenerovanou mapou pro fázi 1. Alternativou je dané komponenty odstranit a první fázi projít ručně pomocí ukazatelů.

#### D.1 Ovládání

- v **FPS** módu se hráč ovládá:
  - Klávesami **WASD**, kde každá klávesa udává směr pohybu dle rozpoložení na klávesnici
  - **Myší** se rozhlíží
  - **Mezerníkem** spustí fázi 1 (když se používá *NavMesh*) a 3
  - **Esc** vypne simulaci
- ve **VR** módu se hráč ovládá:
  - **Triggerem pravého ovladače**. Po stisknutí se zobrazí paprsek a po uvolnění triggeru se hráč přesune.
  - **Grip buttonem pravého ovladače** se spouští fáze 3
  - **Interact UI pravého ovladače** vypne simulaci



## D.2 Předpřipravené modely

Ve složce *Prefabs* Se nacházejí tři složky, do kterých jsou rozděleny všechny předpřipravené modely.

### ■ Needed

Zde se nacházejí modely nutné pro správné fungování aplikace.

### ■ Player

Tento model představuje hráče, obsahuje jak komponenty pro FPS, tak VR používání. Je zde neaktivní objekt *Cylinder*, když se nastaví jako aktivní, bude sloužit pro vizualizaci hráče na mapě. Doporučuji pro VR vypnout. U komponenty *Player Controller* je možné nastavit rychlost hráče (Speed) pro fázi 2 a 3. V komponentě *Nav Mesh Agent* se obdobně může nastavit rychlost a další vlastnosti pro fázi 1. Model obsahuje objekt *GroundCheck*, ten slouží pro kontrolu, zda hráč po aplikování gravitace dosednul na zem. Pro správné fungování se musí všem objektům představující zem nastavit *Layer* (vrstvu) na stejnou hodnotu jako proměnou *Ground Mask Player Controller*. V ukázkové scéně se používá vrstva *Ground*. Je nutné vždy mít aktivní pouze jednu z komponent *PlayerCamera* a *[CameraRig]*. *PlayerCamera* pro FPS mód a *[CameraRig]* pro VR. Po přidání hráče do scény je dobré přesunout ho na místo, na kterém by měl po spuštění začínat ve fázi 1 a ve fázi 2 a 3 končit. Na pravém ovladači v komponentě *[CameraRig]* se nachází skript pro pohyb ve VR módu.

### ■ RuteController

Toto je hlavní řídicí objekt celé scény. Přepíná mezi fázemi a nastavují se zde globální makra pro fungování ostatních objektů. Je potřeba sem ze scény přidat objekt hráče do proměnné *Player Object* a do proměnné *Fade* objekt *Canvas Fade* (viz níže). Vlastnost *VR* slouží k určení, zda se scéna bude nacházet v módu VR nebo FPS. Vlastnost *Skip* se používá během testování scény. Když je zaškrtnutá, po zapnutí se hráč okamžitě přenesení do fáze 3. Dále jsou zde proměnné pro používání *NavMesh* a omezení pro *hemiprostorový neglekt*.

Pole *Points* slouží k nastavení trasy při používání *NavMesh*. Nastavením velikosti (v základu na hodnotě 2) se určí, přes kolik kontrolních bodů musí hráč projít. Pod nastavením velikosti je seznam aktuálně nastavených bodů. Hráč je projde od shora dolů. *RuteController* obsahuje 2 body, pojmenované *Start* a *End*. Pro přidání nových bodů je nutné:

- Zvýšit velikost na požadovaný počet.
- Pravým tlačítkem myši kliknout na *RuteController* ve stromu scény (měl by se nacházet v levé části obrazovky) a vybrat možnost *Create Empty*.
- Nový bod přidat do pole *Points* přetažením na požadované místo v pořadí bodů.
- Ve scéně přesunout bod na požadované místo.

Hodnotou *Radius* se určuje, jak daleko může být hráč od své výchozí pozice z fáze 1, aby se ve fázi 2 dostal do cíle.

### ■ Canvas Fade

Tento objekt slouží k zatemnění obrazovky během přechodů mezi fázemi. Po přidání do scény je potřeba do komponenty *Canvas* přidat kameru z objektu *[CameraRig]*, nacházející se v prefabu *Player*. Ve scriptu *Fade* se může nastavit čas, jak

dlouho tento efekt bude trvat. Funguje pouze ve VR. Je potřeba tento objekt přidat do *RuteController*.

#### ■ Line

Toto je model křivky pro pohyb ve VR pomocí teleportu. Je potřeba ho přidat do scény.

#### ■ Portals

Zde se nachází prefab *NaivePortal*. Po přidání do scény je třeba propojit ho s druhým portálem. To se provede přetažením druhého portálu do vlastnosti *Other Portal* v komponentě *Naive Portal*. Zároveň je potřeba sem obdobným způsobem přidat objekt *Player*. Pro nejlepší výsledný efekt doporučuji umístit oba portály do dvou identických částí mapy a vyzkoušet. Je silně doporučeno upravit rozměry portálu tak, aby překrýval celou oblast a nedal se v nějakém místě obejít. Vhodným způsobem „instalace“ je umístit jeden portál a následně celou oblast zkopírovat a přesunout jinam. Zároveň je zde normální *Portal*, který navíc zobrazuje plochu kam se hráč přesune. Ten ale doporučuji používat jen v FPS verzi.

#### ■ Support

Zde se nachází prefaby, které mají napomoci orientaci hráče během simulace. Všechny tři prefaby, které se zde nacházejí, obsahují komponentu *Show Script*. Její vlastnosti budou popsány níže.

#### ■ TextDisplayer

Tento prefab slouží k zobrazení popisků během simulace. Rozměr, pozice a rotace se upravuje přímo v *TextDisplayer*. Vlastnosti textu se upravují v objektu *Text* a pozadí v *Panelu*. Defaultně se objevuje ve fázi 3.

#### ■ Direction

Slouží k naznačení směru a zvýraznění základní geometrie (např. ulic ve scéně připomínající město). Objeví se ve fázi 2.

#### ■ FinishPointer

Funguje velice podobně jako *Direction* s tím rozdílem, že je vždy natočen směrem k hráči, a po dosažení přednastaveného radiusu kolem cíle zmizí. Měl by se umístit na výchozí pozici hráče, aby mu ve fázi 2 označoval cíl.

## D.3 Skripty

Ve složce *Scripts* se nacházejí další dvě složky. V adresáři *NotUser* jsou skripty, které zajišťují chod aplikace, ale uživatel s nimi nemusí přímo přijít do styku. Naopak ve složce *User* jsou skripty, které krom toho, že je již některé objekty používají, navíc může uživatel přidat ke svým objektům ve scéně, aby pozměnil jejich chování.

#### ■ ShowScript

Tento skript způsobí, že cílový objekt se zobrazí jen v předem určených fázích simulace. Je používán prefaby ve složce *Support*. Ale je možné ho použít i pro vlastní objekt. Aby byl správně použit, je třeba ho přidat jako komponentu k nějakému

jinému objektu. Cílový objekt (který tím má být ovlivněn) je třeba přesunout do objektu se skriptem, aby se stal jeho potomkem. V seznamu potomků musí být nejvýše (první).

U této komponenty je možné nastavit několik vlastností:

- *Change On Stage* – když tato možnost není zaškrtnutá, změní se viditelnost objektu v určené fázi po průchodu hráče jeho startovní pozicí. Když je zaškrtnutá, objekt se objeví jen v určené fázi.
- *Hide At Start* – když *Change On Stage* není zaškrtnutá, ovlivňuje viditelnost objektu ve chvíli, co hráč projde svou startovní pozicí.
- *Target Stages* – Určuje fáze, ve kterých bude změněna viditelnost objektu.
- *Flip Stages* – prohazuje směr animace objektu v daných fázích.
- *Face To Player* – zaškrtnutí způsobí, že se objekt bude vždy natáčet směrem k hráči.

#### ■ ObjectModifier

Tento skript aplikuje na objekt ve fázi 3 předdefinovanou změnu. Je možné vybrat změnu základní barvy, textury nebo celého materiálu. Dále změnu velikosti, rotace či pozice anebo prohození s jiným objektem. Nový objekt musí být přítomný ve scéně, a pokud nemá být dříve viditelný, měl by být nastavený jako neaktivní. Pokud se pole pro nový objekt nechá prázdné, starý objekt pouze zmizí.

#### ■ HightLighting

Po připojení tohoto skriptu objektu, se daný objekt ve fázi 2 zvýrazní a ve fázi 3 může aktivovat efekt *hemiprostorového neglektu*.

## ■ D.4 Shadery

Ve složce *Materials* se nacházejí materiály, které používají prefaby a některé objekty použité v ukázkové scéně. Jedná se buď o materiály se shadery, které nabízí přímo Unity, anebo se shadery, které se nacházejí ve složce *Shaders* a jsou připraveny speciálně pro tento balíček.

#### ■ Fresnell Effect

Tento shader umožňuje obdobné nastavení vlastností materiálu jako běžný shader, ale navíc umožňuje nastavit zvýraznění hran objektu za pomoci záře a nastavit tomuto efektu barvu. Může se používat společně se skriptem *HightLighting*.

#### ■ DirectionShader1

Materiál s tímto shaderem způsobí, že nastavená (černobílá) textura upraví transparentnost objektu (bílá – viditelné, černá – průhledné). Zároveň se textura neustále dokola posouvá od horního okraje k dolnímu. Lze nastavit i barvu viditelných částí. Tento shader je použit pro různé ukazatele a naváděcí prvky.

#### ■ CameraCutoffShader

Tento shader se používá pro portál, který renderuje obraz. Ořezává texturu tak, aby bylo vidět jen to, co má být vidět skrze portál a ne celý obraz kamery.