

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

3D MONITOR

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

JAN SZKANDERA

BRNO 2009



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

3D MONITOR

3D DISPLAY

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

JAN SZKANDERA

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. MICHAL HRADIŠ

BRNO 2009

Abstrakt

Tato bakalářská práce se zabývá návrhem a realizací systému, který umožní obraz scény zobrazovaný na ploše vnímat prostorově. Prostorové vnímání 2D obrazové informace je umožněno jednak stereopromítáním a jednak tím, že se obraz mění v závislosti na poloze pozorovatele. Tato práce se zabývá hlavně druhým z těchto problémů.

Abstract

This Bachelor's thesis goal is to design and realize system, which allows user to perceive 2D visual information as three-dimensional. 3D visual preception of 2D image is possible thanks to stereoprojection and projection dependency of user location. The thesis mainly aims at second of these problems.

Klíčová slova

počítačová grafika, projekční matice, změna souřadného systému, virtuální realita, Nest of Birds

Keywords

computer graphics, projection matrix, change of coordinates, virtual reality, Nest of Birds

Citace

Jan Szkandera: 3D monitor, bakalářská práce, Brno, FIT VUT v Brně, 2009

3D monitor

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením Ing. Michala Hradiše. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Jan Szkandera
20. května 2009

Poděkování

Děkuji mému vedoucímu Ing. Michalu Hradišovi za pomoc a čas, který mi věnoval při řešení této práce. Také děkuji své rodině, která mi byla v době řešení projektu oporou.

© Jan Szkandera, 2009.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

Úvod	2
1 Úvod do problematiky	3
1.1 Virtuální realita	3
1.2 Pojem 3D monitor	3
1.3 Současný stav	4
2 Mimoosová projekce	5
2.1 Základní pojmy	5
2.2 Rozdíly oproti osové projekci	6
2.3 Výpočet projekční matice	7
3 Změna souřadného systému	11
4 Návrh aplikace	14
4.1 Využití mimoosové projekce	14
4.2 Volba souřadného systému scény	15
4.3 Vykreslovací cyklus	16
5 Implementace	18
5.1 Získání pozice pozorovatele	18
5.2 Stereoprojekce	20
5.3 Modul pro práci s Nest of Birds	20
5.4 Aplikace v OpenGL	21
5.5 FPS Aplikace	22
6 Testování	25
6.1 Testování aplikace v OpenGL	25
6.2 Testování FPS aplikace	25
Závěr	27
A Obsah DVD	29
B Manuál k práci s Nest of Birds	30
C Manuál k aplikaci v OpenGL	31
D Manuál k FPS aplikaci	32

Úvod

Extenzivní rozvoj informačních technologií dal možnost vývoji zobrazovacích zařízení pro virtuální realitu. Už na konci šedesátých let minulého století vznikala jednoduchá zařízení, která umožňovala člověku pozorovat třírozměrný prostor vytvořený počítačem. Cílem této práce je návrh a realizace systému, který pomocí dostupných zobrazovacích přístrojů, jako je monitor, umožní člověku vnímat obrazy objektů vytvořených počítačem třírozměrně.

V první kapitole stručně popíšu pojem virtuální realita a vymezím pojem 3D monitor ve smyslu této práce. V závěru kapitoly shrnu poznatky o současných systémech, které se podobají 3D monitoru.

Druhá kapitola se zabývá mimoosovou projekcí (z angl. *off-axis projection*). Účelem mimoosové projekce je matematicky popsat, jak zobrazit objekty v prostoru na plochu, když pohledový jehlan (viz. 2.1) není osově souměrný.

Úkolem třetí kapitoly je seznámit čtenáře s převodem souřadnic mezi dvěma souřadnými systémy.

V další čtvrté kapitole je možno nalézt, obecný návrh 3D monitoru.

V páté kapitole popíšu přístroj Nest of Birds a soustavu projektorů – tato zařízení jsem využil při implementaci aplikací, které jsem vytvořil v praktické části mé práce. V dalších podkapitolách pak shrnu samotnou implementaci těchto aplikací.

Poslední šestá kapitola obsahuje popis a vyhodnocení testů, které jsem provedl s navrženým a realizovaným systémem.

Kapitola 1

Úvod do problematiky

Úkolem této kapitoly je seznámit čtenáře s pojmy jako jsou virtuální realita nebo 3D monitor. Dále zde shrnu současný stav vývoje zařízení, které slouží k zobrazování scény v závislosti na poloze pozorovatele.

1.1 Virtuální realita

Virtuální realita je počítačová simulace určitého prostředí, umožňující člověku interakci s tímto prostředím. Interakce člověka se simulovaným prostředím by měla navodit realistický vjem tohoto prostředí. Pojem virtuální realita je spojován především s vyvoláním realistického zrakového vjemu, kdy pohled na simulovaný prostor závisí na poloze pozorovatele. Kromě tohoto vjemu, je často uživateli interagujícím s prostředím umožněno působit i na simulované objekty v prostředí, pomocí různých druhů zařízení vybavených senzory pro určení polohy.

Zařízení používaná v prostředích virtuální reality mohou být rozdělena na vstupní a výstupní. Zatímco pomocí vstupních zařízení uživatel takového prostředí ovlivňuje objekty ve virtuální realitě, výstupními zařízeními je ovlivňován sám uživatel.

Vstupními zařízeními jsou například rukavice či jiné části oblečení, vybavené senzory pro určení polohy. Tato zařízení jsou spojována s virtuální realitou nejčastěji, ale mohou to být i jiné přístroje závislé na účelu simulace například volant simulátoru automobilu nebo druh zbraně.

Výstupní zařízení jsou, jak jsem se již zmínil, nejčastěji zařízení pro zobrazování simulovaného prostředí. Ovlivňovat však můžeme i jiné smysly uživatele virtuální reality – především sluch (prostorový zvuk) nebo hmat (vibrace zbraně během střelení).

1.2 Pojem 3D monitor

Pojem 3D monitor není nijak definovaný. Často tak bývají nazývány speciálně upravené LCD displeje umožňujících stereoprojekci, a tím 3D vjem objektů zobrazených na takovém displeji. Smyslem mé bakalářské práce není návrh nového přístroje, ale pomocí již vyrobených technologií docílit stejného efektu – tedy vytvořit uživateli realistický prostorový vjem objektů zobrazených na obrazovce 3D monitoru.

Kromě toho, že 3D monitor (ve smyslu této práce) používá dostupných technologií, je jeho další vlastností to, že zobrazuje objekty prostorově v závislosti na poloze pozorovatele. Pokud se uživatel bude před 3D monitorem pohybovat, měl by mít pocit, že se pohybuje

před reálnými prostorovými předměty, které budou ve skutečnosti pouze zobrazeny na ploše 3D monitoru.

Plochou 3D monitoru můžeme rozumět například projekční plátno.

V kontextu této práce je 3D monitor systém zobrazující 3D objekty na ploše tak, aby uživatel vnímal tyto objekty prostorově, přičemž prostorový vjem je zachován i po změně pozice pozorovatele před zobrazovací plochou.

1.3 Současný stav

V současné době existuje celá řada zařízení a technik, které řeší stereoskopické zobrazení, například anaglyfické zobrazení nebo světelně polarizovaný obraz. Problém zobrazení obrazu pro každé oko zvláště je tedy poměrně dobře vyřešen.

Stereoproječní techniky ovšem samy o sobě neřeší problém zobrazení závislém na poloze pozorovatele. Systém, který řeší i tento problém je například CAVE (zkratka z Cave Automatic Virtual Environment). CAVE je tvořen krychlí velikosti menšího pokoje, jejíž stěny jsou tvořeny plátny, na která je promítán obraz z projektorů. Projektory jsou synchronizovány se speciálními LCD brýlemi, umožňujícími zakrýt pohled oka (obecně známy pod anglickým výrazem LCD Shutter Glasses). Obraz je tedy v krátkých časových intervalech promítán na projekční plátna střídavě pro levé a pro pravé oko. Brýle pak přepínají pohledy očí podle toho, pro které oko je obraz právě promítán. Poloha pozorovatele je snímána elektromagnetickými senzory a projektované obrazy na všech plátnech se mění v závislosti na poloze pozorovatele.

Další podobné zařízení je například ImmersaDeskTM, které pro zobrazování používá projektor nebo zařízení Varrier, které obklopuje uživatele množstvím displejů, čítajícím až několik desítek takových panelů.

Výše uvedené systémy jsou v současnosti využívány hlavně ke studijním účelům. Komerční uplatnění nalézají jako prezentační nebo herní zařízení.

Zařízení používaná v praxi můžete vidět na obrázku 1.1.



Obrázek 1.1: Vpravo zařízení CAVE, uprostřed ImmersaDeskTM a vlevo zařízení Varrier. (Zdroje obrázků: CAVE – howstaffworks.com, ImmersaDeskTM – invisu.ca, Varrier – evl.uic.edu)

Kapitola 2

Mimoosová projekce

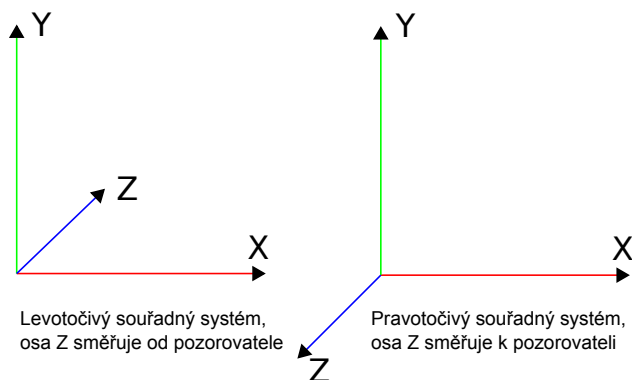
V této kapitole nejprve vysvětlím základní pojmy promítání a perspektivy. V další části vysvětlím postup při výpočtu projekční matice.

Poznatky důležité k pochopení mimoosové projekce jsem čerpal z článků Virtual Window Simulation [7] a hlavně Generalized Perspective Projection [4]. Kvůli dobré názornosti obrázků z článku [4], jsem podle nich vytvářel i obrázky použité v této kapitole.

2.1 Základní pojmy

V této podkapitole stručně vysvětlím některé pojmy, které je vhodné znát k pochopení dalšího výkladu.

Pravotočivá versus levotočivá soustava souřadnic – viz. obrázek 2.1. Prostor v počítačové grafice obvykle popisujeme pomocí tří souřadnic. Tyto souřadnice jsou vztaženy ke třem, na sebe kolmým osám X, Y a Z. Pokud zvolíme směr os X a Y, pak jsou dva možné způsoby určení směru osy Z. Pakliže $(v_x \times v_y)/|v_x \times v_y| = v_z/|v_z|$, kde v_x , v_y , v_z jsou bázové vektory báze souřadného systému, pak je souřadný systém pravotočivý. Naopak levotočivý systém má vlastnost $(v_y \times v_x)/|v_y \times v_x| = v_z/|v_z|$.



Obrázek 2.1: Levotočivá a pravotočivá soustava souřadnic.

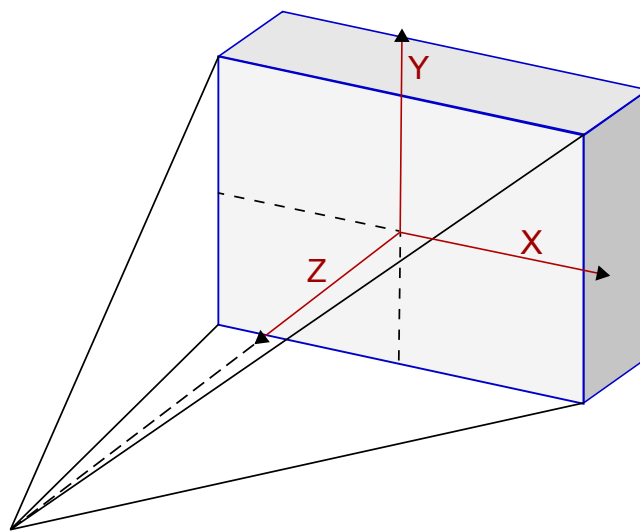
Projekční matice nebo také matice kamery popisuje mapování bodu v 3D prostoru na bod ve 2D obraze zobrazovaném na projekční ploše. U perspektivní projekce můžeme změnou projekční matice měnit tvar pohledového jehlanu.

Pohledový jehlan. V 3D počítačové grafice je pohledový jehlan oblast v modelovaném prostoru, která může být zobrazena na promítací ploše – je to zorné pole pomyslné kamery. Přesný tvar tohoto prostoru závisí na použité čočce kamery, ale typicky je to pravoúhlý komolý jehlan. Základny, které jsou kolmé na směr pohledu, se nazývají bližší ořezávací rovina a vzdálenější ořezávací rovina. Objekty, které jsou umístěny blíže ke kameře než bližší a za vzdálenější ořezávací rovinou, nejsou zobrazeny. Často je vzdálenější ořezávací rovina umístěna nekonečně daleko od kamery, takže všechny objekty uvnitř jehlanu jsou vyobrazeny bez ohledu na jejich vzdálenost od kamery. [9, překlad]

Pro úplnost je třeba uvést, že hrany jehlanu procházejí vrcholy zobrazovací plochy (např. obrazovky monitoru), která je obvykle obdélníkového tvaru.

2.2 Rozdíly oproti osově projekci

U 3D počítačových her se předpokládá, že se uživatel dívá kolmo do středu plochy monitoru a jeho poloha se nemění. Vrchol pohledového jehlanu (viz. podkapitola 2.1), tedy poloha oka pozorovatele, leží na ose vycházející ze středu zobrazovací plochy zobrazovacího zařízení kolmo na tuto plochu – viz. 2.2. Pro pojmenování takového zobrazení se v anglické literatuře vžilo slovní spojení *on-axis projection*, česky lze toto sousloví přeložit jako osová projekce. Pohledový jehlan je u 3D her pevně nastaven a je symetrický. Nemění-li se pohledový jehlan, není třeba měnit ani projekční matici, která ho definuje.

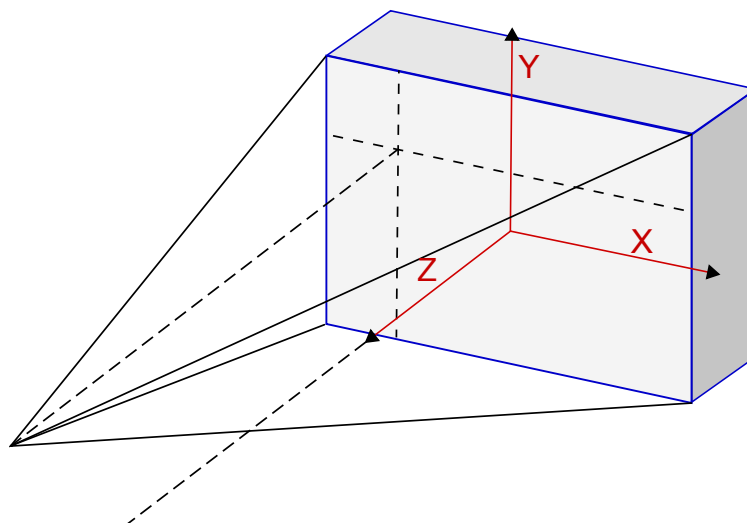


Obrázek 2.2: Pohledový jehlan osově projekce.

Naproti tomu u mimoosové projekce (v anglicky psané literatuře *off-axis projection*) se předpokládá obecně jiná poloha pozorovatele než pouze na ose zobrazovacího zařízení, jak jsem ji popsal v předchozím odstavci – viz. obrázek 2.3.

Z pohledu deskriptivní geometrie jde v obou případech o středová promítání, ale vzájemná poloha středu promítání a průmětny se u osově a mimoosové projekce liší.

Ve zbytku práce budou mít pojmy průmětna, monitor a projekční plocha často stejný význam.



Obrázek 2.3: Pohledový jehlan mimoosové projekce.

2.3 Výpočet projekční matice

Pro snadnější pochopení této podkapitoly by čtenář měl znát geometrický význam vektorových počtů, především by měl mít představu o geometrickém významu takových pojmů jako vektorový součin, skalární součin nebo normalizace vektoru.

Výpočet základní projekční matice

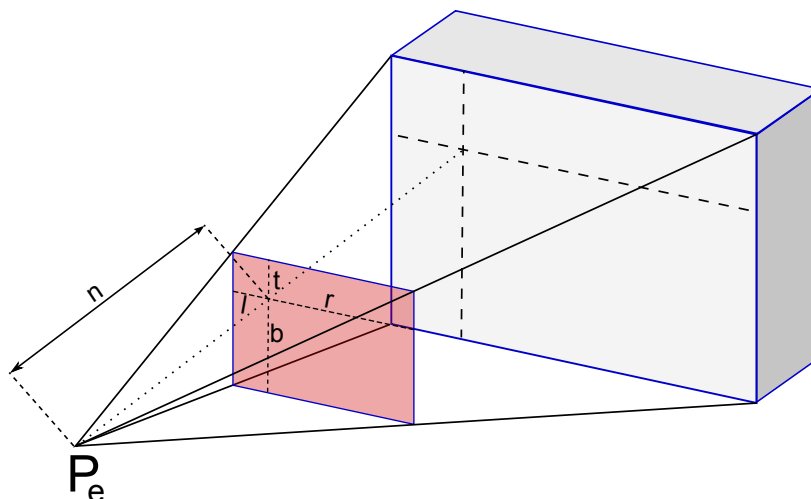
Jak již bylo popsáno výše pokud se pozorovatel před zobrazovací plochou nepohybuje je pohledový jehlan stále stejný. Pokud ovšem pozorovatel mění svou polohu, mění se i pohledový jehlan a s každou změnou polohy pozorovatele je třeba spočítat novou projekční matici. Výstupem výpočtů, které v této podkapitole budu provádět, bude projekční matice spočtená pro konkrétní vztah polohy středu promítání k poloze průmětny.

Pro hlubší studium mimoosové projekční matice doporučuji článek [2].

Nyní už k samotnému výpočtu. Na začátku definuji výsledek práce, kterou budu popisovat v následujícím textu. Hledaná matice v pravotočivém souřadném systému (např. OpenGL) vypadá následovně:

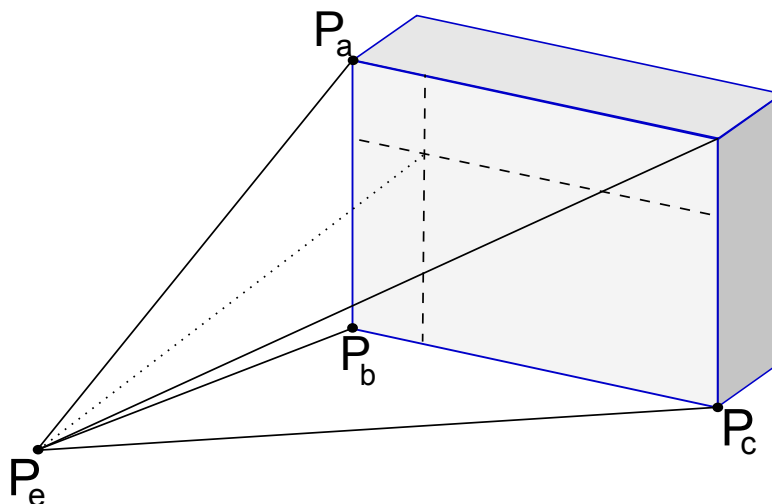
$$P = \begin{pmatrix} \frac{2n}{r-l} & 0 & \frac{r+l}{r-l} & 0 \\ 0 & \frac{2n}{t-b} & \frac{t+b}{t-b} & 0 \\ 0 & 0 & \frac{-(f+n)}{f-n} & \frac{-2fn}{f-n} \\ 0 & 0 & -1 & 0 \end{pmatrix}, \quad (2.1)$$

kde l , r , t , b jsou postupně levý, pravý, horní a dolní rozměr pohledového jehlanu vztahený k bližší zobrazovací rovině, jak je to znázorněno na obrázku 2.4. Písmenky n a f jsou označeny po řadě vzdálenosti od vrcholu pohledového jehlanu k bližší a vzdálenější ořezávací rovině. V levotočivém souřadném systému je třeba pozměnit některá znaménka, v tuto chvíli se tímto problémem nebudeme zabývat. Nicméně tato matice se omezuje pouze na případ, kdy je zobrazovací plocha rovnoběžná s rovinou, kterou určují osy X a Y . Později popíšu i případ, kdy zobrazovací plocha s rovinou XY rovnoběžná nebude.



Obrázek 2.4: Rozměry pohledového jehlanu.

Problém nalezení projekční matice se nyní zúžil na nalezení hodnot rozměrů pohledového jehlanu, a to konkrétně hodnot l , r , t , b . Hodnoty n a f si můžeme libovolně zvolit s tím, že $n > 0$ a $f > n$. Předpokládejme, že známe body v prostoru, které vymezují rozměry monitoru. Označme si tyto body P_a , P_b , P_c , dále známe pozici pozorovatele nebo přesněji jeho oka, kterou označme P_e – viz. obrázek 2.5.



Obrázek 2.5: Vrcholy projekční plochy.

Prvním úkolem bude určit ortonormální bázi zobrazovací plochy – to znamená určit bázové vektory. Vektory si označme v_x , v_y , v_z . Výpočet je následující

$$\vec{v}_x = \frac{P_b - P_a}{|P_b - P_a|}, \vec{v}_y = \frac{P_c - P_a}{|P_c - P_a|}, \vec{v}_z = \frac{\vec{v}_x \times \vec{v}_y}{|\vec{v}_x \times \vec{v}_y|},$$

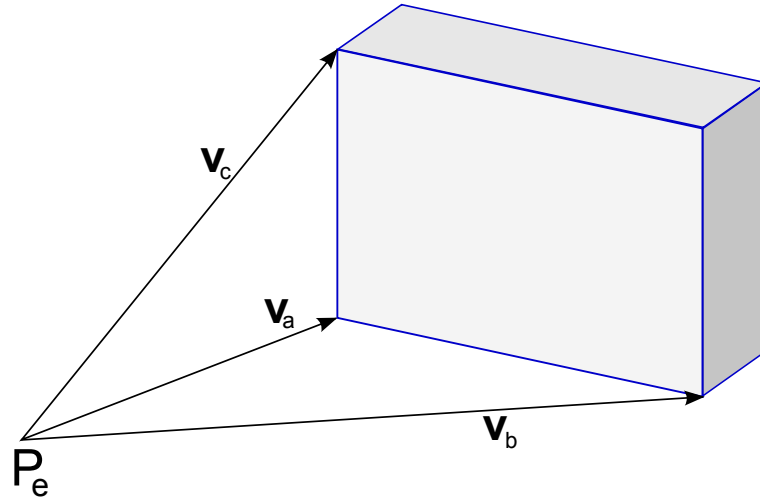
jelikož požadována báze má být ortonormální a souřadný systém je pravotočivý, hodnota vektoru \vec{v}_z se určí jako vektorový součin vektoru \vec{v}_x a \vec{v}_y . Všechny vektory je třeba normalizovat tak, že se podělí vlastním modulem. Je zřejmé, že báze promítací plochy, která je

rovnoběžná s rovinou XY bude mít vždy tvar:

$$\vec{v}_x = (1, 0, 0), \vec{v}_y = (0, 1, 0), \vec{v}_z = (0, 0, 1),$$

proto bázi pro takovouto promítací plochu počítat nemusíme.

Než začneme s výpočty rozměrů pohledového jehlanu, které nám scházejí k určení projekční matice, budeme potřebovat hodnoty vektorů směřujících z vrcholu tohoto jehlanu k bodům P_a, P_b, P_c , tyto vektory si označíme jako $\vec{v}_a, \vec{v}_b, \vec{v}_c$ viz. obrázek 2.6. Všechny hodnoty k určení těchto vektorů jsou známy.



Obrázek 2.6: Vektory $\vec{v}_a, \vec{v}_b, \vec{v}_c$

Dále budeme potřebovat hodnotu vzdálenosti pozorovatele od monitoru, tuto vzdálenost označíme d a vypočítáme jako skalární součin vektorů \vec{v}_z s některým z vektorů \vec{v}_a, \vec{v}_b nebo \vec{v}_c :

$$d = -(\vec{v}_z \cdot \vec{v}_a).$$

Součin $(\vec{v}_z \cdot \vec{v}_a)$ je skalární projekcí vektoru \vec{v}_z do směru vektoru \vec{v}_a (skalární projekce viz. [1]). Správně by rovnice měla vypadat takto $d = -(\vec{v}_z \cdot \vec{v}_a)/|\vec{v}_z|$, ale protože je vektor \vec{v}_z normalizovaný byl by dělitel vždy roven 1. Jelikož je směr vektorů opačný než potřebujeme, je třeba navíc výslednou hodnotu skalární projekce vynásobit -1 .

V tomto okamžiku máme všechny potřebné hodnoty pro výpočet rozměrů pohledového jehlanu (obrázek 2.4). Pro výpočet hodnot rozměrů použijeme opět skalární projekce. Tentokrát do směrů vektorů \vec{v}_x a \vec{v}_y .

$$l = (\vec{v}_x \cdot \vec{v}_a)n/d,$$

$$r = (\vec{v}_x \cdot \vec{v}_b)n/d,$$

$$t = (\vec{v}_y \cdot \vec{v}_a)n/d,$$

$$b = (\vec{v}_y \cdot \vec{v}_c)n/d.$$

Činitel n/d v každém vzorci je uplatněním metody podobností trojúhelníků. Kdybychom podíl n/d vynechali získali bychom sice rozměry l, r, t, b , ovšem nebyly by vztaheny k bližší zobrazovací rovině, ale k ploše monitoru.

Nyní už známe všechny potřebné proměnné, abychom mohli sestavit projekční matici pro konkrétní polohu pozorovatele a můžeme tyto proměnné dosadit do matice (2.1).

Rotace pohledového jehlanu

Jak již jsem se zmínil, výše uvedenou maticí můžeme popsat projekci pouze takového zařízení, jehož plocha je rovnoběžná s plochou os XY . Pokud bychom například chtěli využít více zobrazovacích ploch, které by obklopovaly pozorovatele a zároveň byly ve stejném souřadném systému, tak by nám taková projekční matice nestačila. Musíme převést námi upravený pohledový jehlan ze souřadného systému monitoru do obecného souřadného systému pro všechny monitory. Převod mezi dvěma rozdílnými ortonormálními bázemi můžeme provést tak, že vynásobíme už vypočítanou matici maticí přechodu

$$R = \begin{pmatrix} v_{x1} & v_{y1} & v_{z1} & 0 \\ v_{x2} & v_{y2} & v_{z2} & 0 \\ v_{x3} & v_{y3} & v_{z3} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (2.2)$$

kde v_x, v_y, v_z jsou bázové vektory monitoru. Se změnou báze se čtenář může blíže seznámit v kapitole 3 nebo doporučuji článek [3].

Posunutí pohledového jehlanu

Stále to ovšem nestačí k tomu, abychom dosáhli toho, co jsme si v podkapitole 2.2 definovali jako mimoosovou projekci. Změnou projekční matice jsme pouze změnili tvar pohledového jehlanu, nyní potřebujeme jeho vrchol přenést na novou pozici pozorovatele. Nejsnáze dosáhneme požadovaného posunutí tak, že projekční matici vynásobíme maticí translace. Tvar matice translace pak bude následující:

$$T = \begin{pmatrix} 1 & 0 & 0 & -P_{e1} \\ 0 & 1 & 0 & -P_{e2} \\ 0 & 0 & 1 & -P_{e3} \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (2.3)$$

kde P_{e1}, P_{e2}, P_{e3} jsou prvky bodu, který jsme si na začátku této podkapitoly definovali jako polohu oka pozorovatele.

Shrnutí

Nyní se pokusím shrnout celý výpočet. K tomu, abychom o projekci mohli hovořit jako o mimoosové, musíme pozměnit pohledový jehlan, který nemusí být jako u osové projekce symetrický. Pohledový jehlan je definován projekční maticí, je tedy nutné provést s každou změnou polohy pozorovatele změnu této matice. Tuto změnu provedeme tak, že nejdříve spočítáme základní projekční matici pro pohledový jehlan tak, jako kdyby plocha monitoru ležela v rovině určené osami X a Y – viz. (2.1). Poté, pokud požadujeme, aby monitor ležel v obecně různé poloze vůči rovině os XY , tuto matici vynásobíme maticí přechodu – viz. (2.2). Nakonec přesuneme vrchol pohledového jehlanu na novou pozici pozorovatele pomocí vynásobení stávající matice translační maticí – viz. (2.3).

Kapitola 3

Změna souřadného systému

Čas od času vznikne problém, kdy vyvíjený systém používá více než jen jeden souřadný systém. Pro snazší práci s takovým systémem je někdy výhodnější souřadné systémy sjednotit. Tato kapitola se zabývá eliminací jednoho souřadného systému převodem jeho souřadnic do souřadnic jiného souřadného systému v třidimenzionálním prostoru. Na konci kapitoly je uveden praktický příklad výpočtu změny souřadného systému v dvojrozměrném prostoru.

Teorie

Předpokládejme, že existuje třidimenzionální prostor R^3 . Tento prostor můžeme popsat dvěma souřadnými systémy S a S' s bázemi (soustavami bázových vektorů) B a B' . Počátek systému S neleží ve stejném bodě jako počátek systému S' . Osy souřadného systému S nejsou rovnoběžné s osami souřadného systému S' . Dále předpokládejme v prostoru R^3 bod P . Bod P popsany bází B si označme $[P]_B$ a bází B' si označme $[P]_{B'}$. Body i vektory budeme zapisovat ve sloupcovém tvaru.

Nyní si definujeme takovou matici T_B , aby platilo

$$[P]_{B'} = [T_B]_{B'} \cdot [P]_B. \quad (3.1)$$

Taková matice se nazývá matice přechodu od báze B k B' a má tvar

$$[T_B]_{B'} = \begin{pmatrix} \alpha_1 & \beta_1 & \gamma_1 \\ \alpha_2 & \beta_2 & \gamma_2 \\ \alpha_3 & \beta_3 & \gamma_3 \end{pmatrix}, \quad (3.2)$$

kde $\vec{\alpha}, \vec{\beta}, \vec{\gamma}$ jsou bázové vektory báze B měřené v systému s bází B' . Pokud budeme chtít provést zpětný přechod z báze B' k bází B , stačí invertovat matici přechodu, čímž získáme matici zpětného přechodu od báze B' k bází B

$$[P]_B = [T_{B'}]_B \cdot [P]_{B'} = [T_B]_{B'}^{-1} \cdot [P]_{B'}. \quad (3.3)$$

Důkaz rovnic 3.1, 3.2 a 3.3 je uveden v knize [5, kap. 3.9].

V této chvíli jsme schopní provést změnu báze dvou souřadných systémů. To ovšem nestačí k tomu, abychom mohli říci, že umíme převést souřadnice bodu z jednoho souřadného systému do druhého. To co jsem popsal výše je možné označit jako převod mezi souřadnými systémy pouze tehdy, mají-li tyto systémy společný počátek.

Změnou báze jsme provedli pouze rotaci os a změnili měřítko původního souřadného systému. Abychom mohli změnu báze využít i v případech, kdy počátky souřadných systému nebudou umístěny ve stejném bodě, je třeba před přechodem mezi bázemi provést translaci bodu $[P]_{B'}$ o vektor \vec{a} . Souřadnice vektoru odpovídají souřadnicím středu souřadného systému, do kterého převádíme. Souřadný systém vektoru \vec{a} bude vždy ten, z něhož převádíme. Tento vektor nazýváme translační vektor. Rovnice přechodu z S do S' bude mít tvar

$$[P]_{S'} = [T_B]_{B'} \cdot ([P]_S - [\vec{a}]_S). \quad (3.4)$$

Rovnice zpětného přechodu ze systému S' do S pak má následující tvar

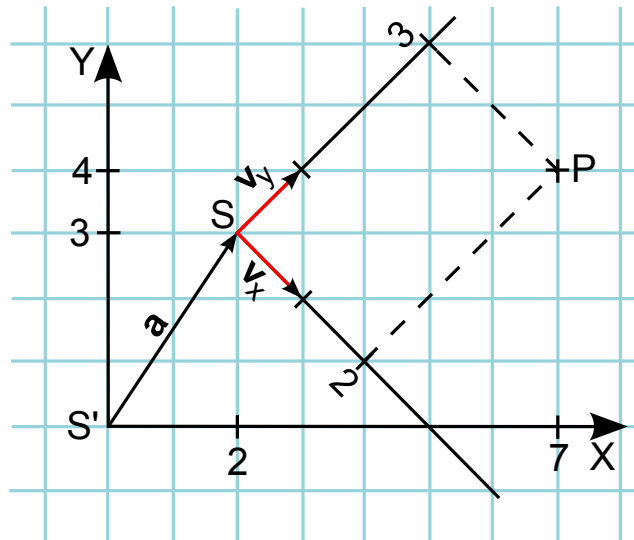
$$[P]_S = [T_B]_{B'}^{-1} \cdot ([P]_{S'} - [\vec{a}]_{S'}), \quad (3.5)$$

kde $[P]_S$ je poloha bodu P v souřadném systému S , $[P]_{S'}$ je poloha bodu P v souřadném systému S' , $[\vec{a}]_{S'}$ je translační vektor směřující z počátku systému S' do počátku systému S v souřadnicích systému S' a $[T_B]_{B'}$ je matice přechodu, jejíž sloupce tvoří bázové vektory báze B rovněž v souřadnicích S' .

Praktický příklad

Vzhledem k tomu, že matematická definice změny souřadného systému nemusí být úplně názorná, uvedu i praktický příklad. Pro jednoduchost demonstřuji princip výpočtu v prostoru R^2 . Mějme situaci popsanou obrázkem 3.1.

Máme dva souřadné systémy se středy S a S' . Dále máme bod P , jehož souřadnice známe v souřadném systému S' a chceme je převést do systému S . V systému S' jsme dále schopni určit bázi (\vec{v}_x a \vec{v}_y) a střed souřadného systému S .



Obrázek 3.1: Obrázek k praktickému příkladu.

Číselné hodnoty tedy jsou:

$$[P]_{S'} = \begin{bmatrix} 7 \\ 4 \end{bmatrix}, [\vec{v}_x]_{S'} = \begin{pmatrix} 1 \\ -1 \end{pmatrix}, [\vec{v}_y]_{S'} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, [\vec{a}]_{S'} = \begin{pmatrix} 2 \\ 3 \end{pmatrix}.$$

Máme tedy translační vektor a zbývá sestavit matici zpětného přechodu z báze systému S' do báze systému S . Matice bude vypadat takto

$$[T_{B'}]_B = [T_B]_{B'}^{-1} = \begin{pmatrix} [v_{x1}]_{S'} & [v_{y1}]_{S'} \\ [v_{x2}]_{S'} & [v_{y2}]_{S'} \end{pmatrix}^{-1} = \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix}^{-1} = \begin{pmatrix} \frac{1}{2} & -\frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{pmatrix}.$$

Všechny potřebné hodnoty známe a můžeme převést bod P ze souřadnic S' do souřadnic S pomocí rovnice 3.5.

$$\begin{aligned} \underline{[P]}_S &= [T_B]_{B'}^{-1} \cdot ([P]_{S'} - [\vec{a}]_{S'}) \\ &= \begin{pmatrix} \frac{1}{2} & -\frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{pmatrix} \cdot \left(\begin{pmatrix} 7 \\ 4 \end{pmatrix} - \begin{pmatrix} 2 \\ 3 \end{pmatrix} \right) \\ &= \begin{pmatrix} \frac{1}{2} & -\frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{pmatrix} \cdot \begin{pmatrix} 5 \\ 1 \end{pmatrix} \\ &= \begin{pmatrix} 5 \cdot \frac{1}{2} + 1 \cdot \left(-\frac{1}{2}\right) \\ 5 \cdot \frac{1}{2} + 1 \cdot \frac{1}{2} \end{pmatrix} \\ &= \begin{bmatrix} 2 \\ 3 \end{bmatrix}, \end{aligned}$$

dosáhli jsme tedy správného výsledku. Bod, jehož souřadnice v souřadném systému S' jsou $\begin{bmatrix} 7 \\ 4 \end{bmatrix}$, má v systému S souřadnice $\begin{bmatrix} 2 \\ 3 \end{bmatrix}$.

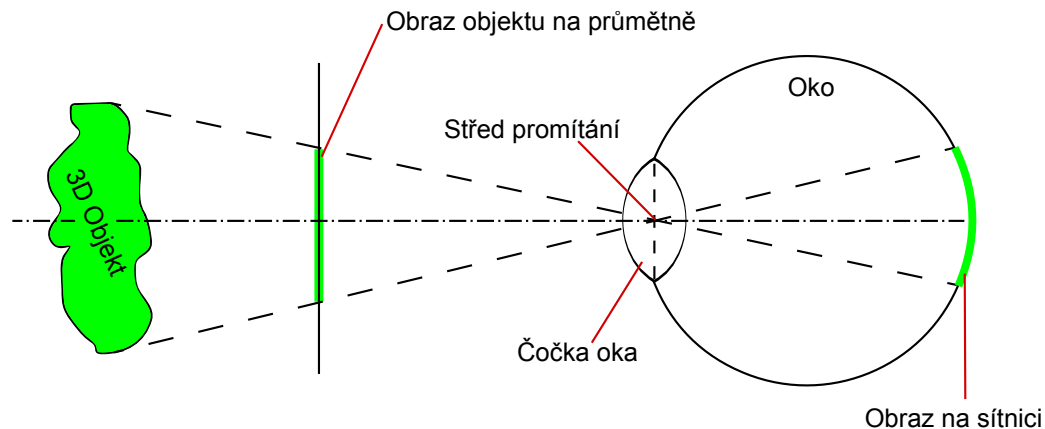
Kapitola 4

Návrh aplikace

Před implementací aplikace je dobré si ujasnit jaké procesy budou v programu této aplikace probíhat a co by mělo být na projekční ploše zobrazeno. Tento návrh bude obecný co se týče použitého prostředí pro vykreslování 3D grafické scény. Jako zdroj souřadnic polohy pozorovatele budu považovat zařízení Nest of Birds popsané v části 5.1 další kapitoly.

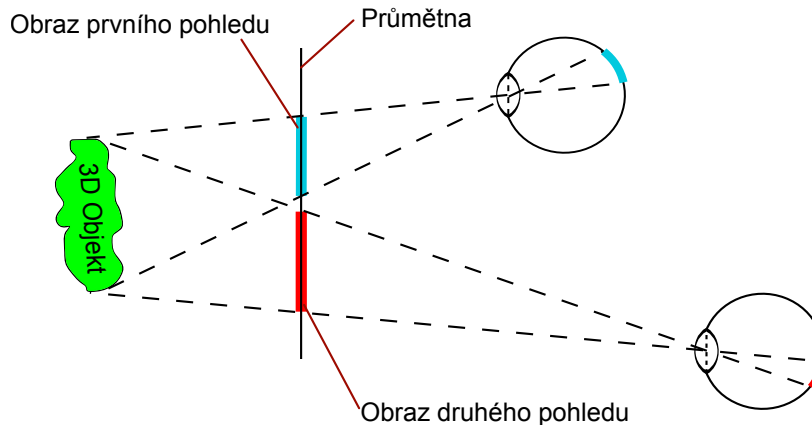
4.1 Využití mimoosové projekce

Perspektivní projekce slouží v počítačové grafice i v jiných oblastech jako simulace toho jakým způsobem vidí člověk. Pokud se k takovému vidění chceme přiblížit co nejvíce, pak budeme jako střed perspektivní projekce uvažovat optický střed čočky lidského oka – viz. obrázek 4.1.



Obrázek 4.1: Lidské vidění. Pohled z boku na lidské oko pozorující 3D objekt. Obraz objektu je projektován na průmětnu na průmětnu.

Uvažujme nyní, že obraz 3D objektu je středovým zobrazením (perspektivní projekce) promítnut na průmětnu tak, že středem promítání je oko pozorovatele. Pokud se člověk nebude pohybovat a bude se dívat pouze jedním okem, pak nepozná zda pozoruje 3D objekt nebo jeho obraz na průmětně. Pohne-li však hlavou rozdíl mezi obrazem a skutečným objektem rozezná. Pokud bychom chtěli, aby člověk nepoznal rozdíl ani tehdy, když se bude pohybovat, pak budeme muset obraz na průmětně s každou změnou polohy oka překreslit – viz. obrázek 4.2.



Obrázek 4.2: Obraz stejného objektu pozorovaného ze dvou odlišných pozic.

Čtenář již jistě tuší, že průmětnou, na které budeme zobrazovat obraz 3D objektů, bude buď monitor nebo jiné zobrazovací zařízení a k zobrazení využijeme mimoosové projekce popsané v kapitole 2.

Jak namapovat 3D souřadnice objektu na 2D souřadnice jeho obrazu již víme – použijeme projekční matici mimoosové projekce. Zbývá ještě vyřešit problém stereoprojekce, protože člověk k pozorování využívá oči dvě (obrázek 4.3). Z pohledu použití mimoosové projekce je to pouze triviální problém – stačí spočítat projekční matice pro dvě kamery, které v grafické knihovně reprezentují oči pozorovatele, a pak zobrazit výsledné dva obrazy zařízením, které umožňuje stereopromítání (například anaglyf nebo polarizací světla ze dvou projektorů).

4.2 Volba souřadného systému scény

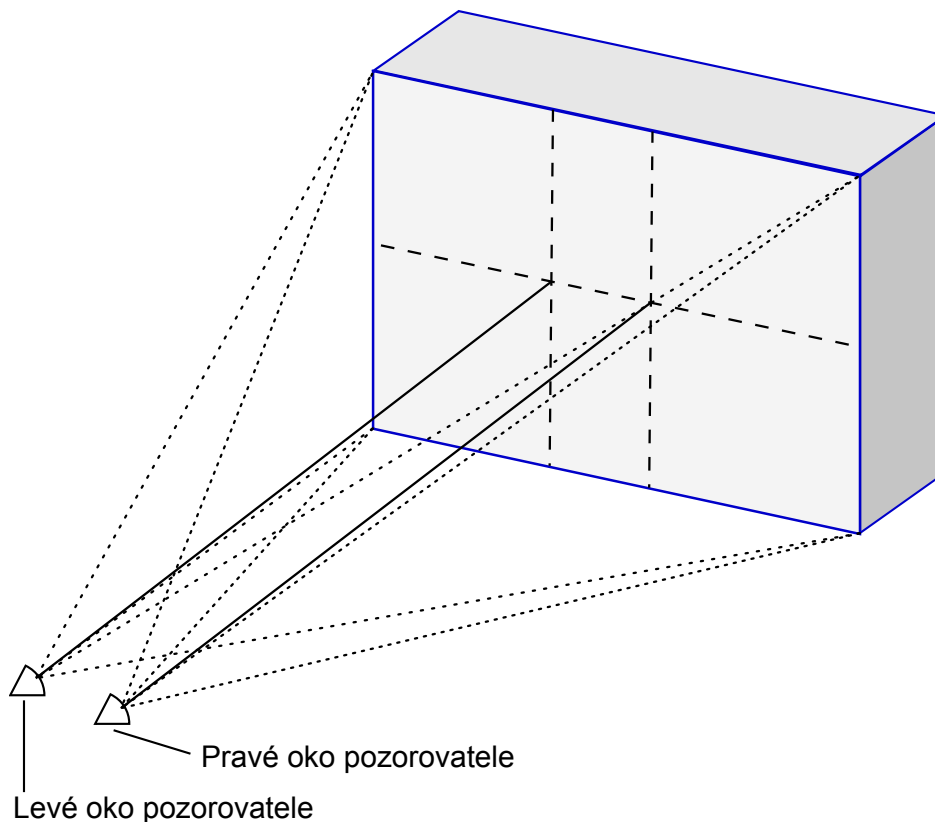
U volby souřadného systému scény máme na výběr ze dvou možností. Buď bude scéna ve stejném souřadném systému jako zařízení pro zjišťování polohy nebo scénu určíme vlastní souřadný systém a budeme hodnoty naměřené v systému zařízení převádět do systému scény.

Představme si, že máme jeden souřadný systém, a to systém zařízení Nest of Birds. Počátek tohoto systému je umístěn ve středu vysílače magnetického pole. V tomto souřadném systému vytvoříme scénu. Před vykreslením scény změříme pro správné zobrazení polohu projekční plochy a vykreslíme scénu. Poté dojde z neznámého důvodu, k posunutí vysílače magnetického pole. Změříme tedy znovu polohu projekční plochy a vykreslíme scénu. Projekce scény bude sice správná, ale samotná scéna bude posunuta.

Nyní si představme, že máme scénu s vlastním souřadným systémem, jehož bázi i střed můžeme určit zařízením Nest of Birds. Pokud můžeme určit bázi i střed souřadného systému v jiném souřadném systému, pak mezi těmito systémy můžeme převádět souřadnice – viz. kapitola 3. Vykreslíme tedy scénu a opět dojde k posunutí vysílače magnetických impulsů. Změříme tedy opět bázi a střed systému scény v souřadnicích Nest of Birds a znovu vykreslíme. Nyní se scéna zobrazí stejně jako před posunutím souřadného systému.

Z předchozího textu vyplývá, že pro 3D monitor bude výhodnější, když scéna nebude závislá na souřadném systému zařízení pro určování polohy. Nezávislost scény se vyplatí i při možném přechodu na jiné zařízení k lokalizaci uživatele, než je Nest of Birds.

Z kapitoly 3 víme, že k převodu souřadnic mezi dvěma souřadnými systémy je nutné určit



Obrázek 4.3: Projekční plocha pozorovaná očima pozorovatele.

bázi a střed souřadného systému, do kterého převádíme v souřadnicích systému, ze kterého převádíme. Tyto hodnoty můžeme změřit například tak, že zobrazíme bázi souřadného systému scény na projekční ploše pomocí úseček a ty poté změříme zařízením Nest of Birds. Pokud zvolíme souřadný systém scény tak, aby osy X a Y takového systému ležely v rovině projekční plochy, budou mít zobrazené úsečky reprezentující osy souřadného systému velikost i směr odpovídající bázovým vektorům. Bázový vektor osy Z můžeme určit například pomocí kartézského součinu zbývajících dvou s tím, že pořadí operandů této operace závisí na tom je-li souřadný systém scény levotočivý či pravotočivý. Ve společném bodě těchto úseček změříme střed souřadného systému scény. Tím získáme všechny potřebné hodnoty a po dosazení do matice přechodu a výpočtu translačního vektoru můžeme začít převádět mezi souřadnými systémy.

4.3 Vykreslovací cyklus

Scéna musí být překreslována co nejčastěji, aby vznikl dojem spojitého pohybu objektů ve scéně i spojitě změny projekce. Část programu starající se o vykreslování scény bude tvořena cyklem, v jehož těle bude muset program vykonat několik úloh. Jelikož bude k zobrazování scény využito stereozobrazování, budou muset být některé úlohy vykonávány dvakrát – zvláště pro každé oko.

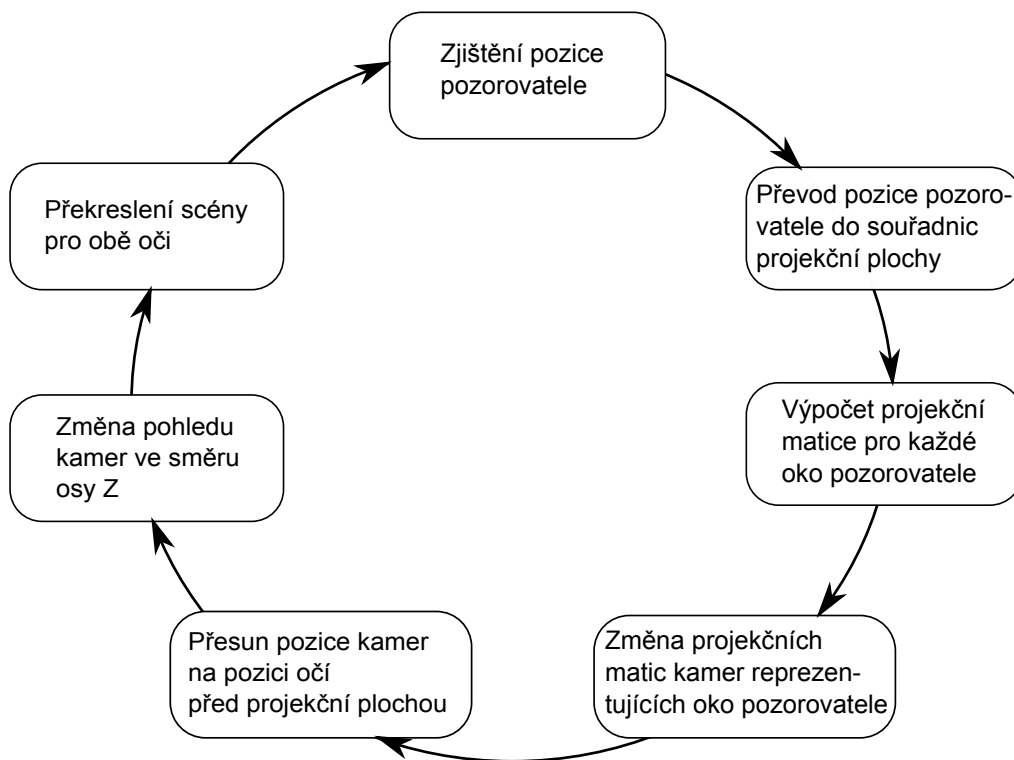
Abychom mohli určit pohled na scénu, potřebujeme znát polohu očí pozorovatele. První úlohou ve vykreslovacím cyklu tedy bude určení této polohy. Nest of Birds určí polohu senzoru přibližně sto krát za sekundu. Tato frekvence měření je dostačující k tomu, abychom

mohli v každém cyklu změřit polohu oka a přitom zachovali spojitý vjem zobrazení scény.

Dalším krokem po získání pozice očí pozorovatele bude převod souřadnic získaných z předešlého kroku do souřadného systému scény. K řešení této úlohy poslouží znalosti z kapitoly 3.

Pokud budeme mít pozice obou očí v souřadnicích scény, můžeme přikročit k výpočtu projekčních matic obou obrazů (jsou dva pro každé oko jeden) scény. K výpočtu využijeme znalosti z kapitoly 2. Kromě polohy oka musíme znát i souřadnice vrcholů obdélníka, který reprezentuje projekční plochu. Protože jsme eliminovali souřadnice měřícího zařízení, nemělo by smysl udávat souřadnice vrcholů projekční plochy v jiném souřadném systému než v systému scény. Vrcholy projekční plochy si můžeme určit například podle rozměrů okna, ve kterém budeme scénu zobrazovat.

Po vypočtení projekčních matic můžeme změnit projekční matice kamer reprezentujících očí pozorovatele. Tato úloha bude záviset na konkrétní knihovně, kterou použijeme pro vykreslování.



Obrázek 4.4: Vykreslovací smyčka scény.

Dalším úkolem bude přenesení pozice kamery na místo oka. Opět půjde o úlohu, kterou je třeba provést pro každé oko zvlášť.

Kromě pozice kamery je také potřeba změnit směr jejího pohledu. Vektor pohledu kamery musí být rovnoběžný s osou Z (viz. obrázek 2.4).

Pokud není třeba měnit chování objektů ve scéně, může se scéna po provedení všech výše uvedených úloh překreslit.

Schéma navrženého vykreslovacího cyklu můžete vidět na obrázku 4.4.

Kapitola 5

Implementace

Pro demonstraci a testování poznatků, které jsem načerpal a následně je popsal v předchozích kapitolách jsem vytvořil dvě aplikace. Obě aplikace jsem napsal v jazyce *C++* s využitím nástroje *Microsoft Visual Studio C++*. Program jsem implementoval pouze pro operační systém *Microsoft Windows*, a to z důvodu nedostupnosti ovladačů k Nest of Birds pro jiné operační systémy.

Nejprve v této kapitole popíšu zařízení Nest of Birds a soustavu projektorů, které jsem využil při realizaci navrženého systému.

Pro práci se zařízením Nest of Birds jsem vytvořil modul, který popíšu v další části kapitoly.

V závěru kapitoly popíšu implementaci zmíněných aplikací, které jsem vytvořil.

5.1 Získání pozice pozorovatele

Vzhledem k možnosti využít fakultního zařízení pro určování polohy bodu v prostoru pomocí magnetického pole (Nest of Birds) jsem se při implementaci zabýval tímto způsobem.

Kromě Nest of Birds by se k určování pozice pozorovatele daly využít například i optické metody. Získávání pozice pozorovatele z obrazové informace, získané soustavou kamer, by sebou mohlo nést výhody, které při použití Nest of Birds nejsou možné. Jednou z těchto výhod by mohla být nižší pořizovací cena kamer. Nevýhodou takového systému je však poměrně obtížná implementace v porovnání s Nest of Birds.

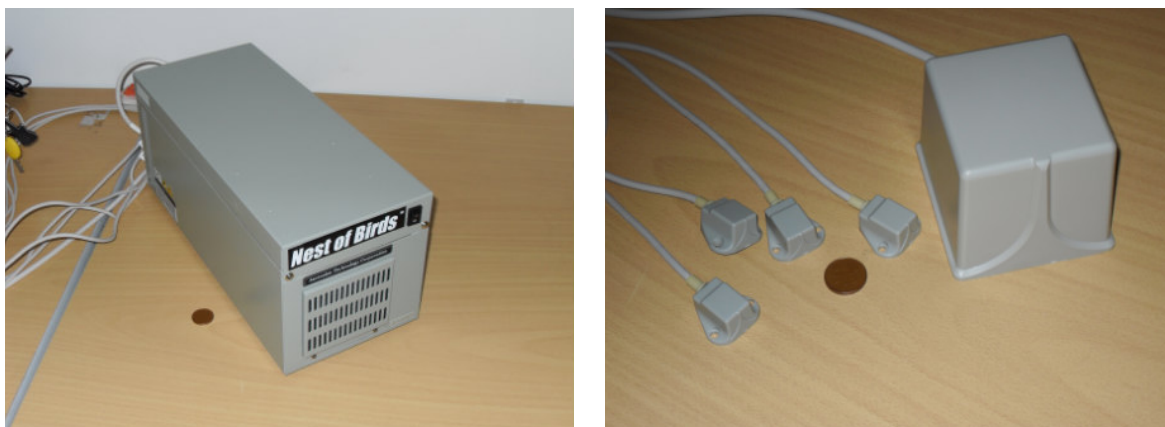
Nest of Birds

Nest of Birds (viz. obrázek 5.1) je produktem firmy Ascension Technology Corporation, která se zabývá výrobou řady zařízení pro určování polohy pomocí magnetických impulsů.

Přístroj je tvořen hlavní napájecí a výpočetní jednotkou, vysílačem magnetických impulsů a snímači intenzity magnetického pole. Vysílač i senzory jsou k hlavní jednotce připojeny pomocí kabelů.

Zařízení udává souřadnice sensorů v třírozměrném ortogonálním souřadném systému, jehož počátek je umístěn ve středu vysílače. Fakulta informačních technologií (dále jen FIT) vlastní model se standardním vysílačem impulsů. Standardní vysílač umožňuje vytvořit magnetické pole, ve kterém lze detekovat polohu senzoru do vzdálenosti cca 36 palců ($\approx 91,44$ cm) od vysílače, tato vzdálenost je pro volný pohyb, například před promítacím plátnem, poměrně krátká. Kromě standardního vysílače je možné dokoupit tzv. rozšířený vysílač, umožňující pokrýt prostor magnetickým polem do vzdálenosti cca 3 metrů (při

použití dvou rozšířených vysílačů až 6 metrů). FIT bohužel nemá rozšířené vysílače k dispozici.



Obrázek 5.1: Obrázek vlevo je hlavní část přístroje Nest of Birds. Vpravo je standardní vysílač a senzory přístroje Nest of Birds. (K porovnání je přiložena desetikorunová mince)

Fakultní zařízení je vybaveno čtyřmi snímači. Kromě polohy bodu v prostoru, lze snímat i orientaci snímačů vzhledem k vysílači. Pozice snímače je snímána s přesností $1,8\text{ mm}$ v případě, že je snímač ve vzdálenosti od 23 do 76 centimetrů od vysílače magnetických impulsů, kdy je pozice měřena nejpřesněji. Přesnost měření je ovlivňována nejen vzdáleností od vysílače impulsů, ale hlavně přítomností kovových předmětů nebo elektrických zařízení, která značně ovlivňují naměřené hodnoty. Například měřit pozici snímače ve vzdálenosti několika centimetrů (např. 2 cm) od zapnutého LCD monitoru je nemožné, jelikož naměřené hodnoty nejsou z hlediska pozice senzoru relevantní.

Přístroj na FIT komunikuje s počítačem pomocí rozhraní *RS-232* (firma Ascension nabízí i model s rozhraním USB). Je možné provést okolo 100 měření za sekundu, to je dostatečná frekvence pro navrhovanou aplikaci.

V době psaní této práce byly pro přístroj Nest of Birds ovladače pouze pro operační systém Windows. Přístup k funkcionalitě přístroje je umožněn pomocí *software development kit (SDK)* pro programovací jazyk *C*. Kromě tohoto *SDK* lze využít i přímé komunikace přes rozhraní *RS-232* (k dispozici je dokumentace k jednotlivým signálům). Implementace takového řešení by ovšem byla časově velmi náročná.

Důležité parametry přístroje Nest of Birds jsou uvedeny v tabulce 5.1.

Parametr	Hodnota
Rozsah měřeného prostoru	$\pm 91,44\text{ cm}$ – ve všech směrech
Počet senzorů	4
Komunikační rozhraní	RS-232
Počet měření za sekundu	až 105 měření za sekundu
Přesnost měření pozice	$1,8\text{ mm}$
Podporovaný programovací jazyk	jazyk <i>C</i>
Podporovaný operační systém	MS Windows

Tabulka 5.1: Tabulka parametrů zařízení Nest of Birds.

Z výše uvedeného popisu zařízení plyne, že je možné toto zařízení v omezené míře použít pro mou aplikaci, je ovšem nutné uvést jeho výhody a nevýhody. Výhodou tohoto zařízení je rychlost s jakou je možné provádět měření polohy bodu v prostoru a poměrně jednoduchá práce s tímto přístrojem. Nevýhodou je pak značně omezený prostor, ve kterém je možno měřit a také to, že ovladače jsou dostupné pouze pro operační systém Windows.

5.2 Stereoprojekce

Člověk používá k pozorování okolního světa dvě oči. Využívá stereoskopického vidění k rozeznávání prostorových vlastností objektů, jež pozoruje. Nestačí tedy zobrazit pouze jeden obraz pro obě oči, je nutné zobrazit pro každé oko jeden obraz – zobrazení dvou obrazů nazýváme stereoprojekcí.



Obrázek 5.2: Sestava dvou projektorů umožňující stereoprojekci obrazu scény.

Při implementaci a testování aplikací, které jsem vytvořil, jsem využil přípravku nainstalovaného na naší fakultě – viz. obrázek 5.2. Tento přípravek umožňující stereoprojekci je složen ze dvou projektorů, které promítají obraz na společnou projekční plochu. Světlo vycházející z těchto projektorů je polarizováno polarizačními filtry tak, že světlo jednoho projektoru je polarizováno horizontálně a druhého vertikálně. Jelikož je světlo promítáno na jednu plochu je výsledný obraz obou projektorů nutné oddělit pro každé oko. K tomu slouží polarizační brýle.

5.3 Modul pro práci s Nest of Birds

Obě vytvořené aplikace používají jako zdroj informací o poloze pozorovatele zařízení Nest of Birds. Vytvořil jsem pro práci s tímto zařízením modul s jednoduchým rozhraním.

Jednotlivé úkony, pro správnou práci se zařízením Nest of Birds se dají rozdělit na části – inicializace zařízení, měření hodnot a ukončení práce. Během práce s přístrojem může dojít i k neočekávanému chování a proto je občas třeba obnovit původní stav komunikace s přístrojem.

- Pro **inicializaci zařízení** jsem vytvořil funkci `initNest`. Tato funkce má jeden parametr, a to číslo sériového portu typu `int`, na kterém je Nest of Birds připojeno. Pokud se inicializace zdaří, funkce vrátí hodnotu 1, jinak `-1`.

Musím však upozornit na to, že inicializace neproběhne v pořádku pouze pomocí modulu, který jsem vytvořil. Pro správnou funkčnost modulu je třeba před spuštěním aplikace, ve které je modul použit, spustit program *winBIRD* dodávaný na instalačním CD k přístroji. V tomto programu je třeba navázat komunikaci pomocí rozhraní *RS-232* a poté přepnout režim vysílání magnetických impulsů na možnost *Cool-Down*. Pokoušel jsem se tuto nepříjemnou vlastnost mého modulu odstranit, ovšem nefunkční je dokonce i kód přebraný přímo od výrobce Nest of Birds. Řešením by mohlo být sledování komunikace na sériovém rozhraní při nastavování režimu vysílání impulsů v programu *winBIRD*.

Postup nastavení Nest of Birds po zapnutí je uveden v příloze B.

- **Přístrojem naměřené hodnoty** jsou předávány jediným parametrem funkce `getDataFromBirds`, kterým je dvourozměrné pole typu `float` o 4 řádcích a 3 sloupcích, ve kterém jsou vráceny polohy všech čtyř senzorů. Řádky reprezentují jednotlivé senzory a sloupce souřadnice polohy těchto senzorů. Jednotkou předávaných hodnot jsou palce a pohybují se v rozmezí od 0 do 32 palců, tedy hodnot jež je schopno Nest of Birds naměřit se standardním vysílačem magnetických impulsů popsaném v podkapitole 5.1.
- Řádné **ukončení práce** s přístrojem je v modulu implementováno pomocí funkce `shutdownNest` bez parametrů.
- Během měření může dojít k poruše správného chování přístroje. Implementoval jsem proto navíc funkci `resynchNest`, která neprovádí nic jiného než ukončení práce a opětovnou inicializaci zařízení, tím dojde k **resynchronizaci** aplikace s přístrojem. Parametrem funkce je stejně jako u funkce pro inicializaci číslo sériového portu přístroje typu `int`.

Modul je implementován ve formě hlavičkový soubor s rozhraním (`nest-of-birds.h`) a soubor se zdrojovým kódem (`nest-of-birds.cpp`).

5.4 Aplikace v OpenGL

Aplikace implementována v prostředí OpenGL byla první aplikací, kterou jsem vytvořil. Pomocí této aplikace jsem ladil vykreslování s využitím mimoosové projekce i práci s přístrojem Nest of Birds.

Manuál k této aplikaci je možné nalézt v příloze C.

Popis programu

Zdrojový kód jsem rozdělil na několik částí. Již jsem se zmínil, že jsem pro práci s přístrojem Nest of Birds vytvořil jeden modul. Další částí programu je modul scény (`scene.h`), který se stará o vykreslování jednotlivých snímků. Kód tohoto modulu využívá knihovny s matematickými funkcemi (`math-functions.h`). Tato knihovna obsahuje funkce pro práci s vektory a maticemi, bližší informace naleznete v hlavičkovém souboru.

Ve funkci `main`, po nezbytné inicializaci scény, grafické knihovny *OpenGL* a zařízení *Nest of Birds*, je řízení předáno hlavnímu vykreslovacímu cyklu. Správné vykreslování ovšem nezačne ihned po spuštění programu. Nejdřív je potřeba změřit bázové vektory souřadného systému scény. Postup tohoto úkonu je popsán v příloze C. Poté už program vykresluje obraz scény na monitoru v závislosti na poloze senzoru zařízení *Nest of Birds*. Vykreslovací cyklus kopíruje pořadí úkonů popsáných v podkapitole 4.3.

Pro vykreslení jednotlivých snímků je vytvořena funkce `drawView`. Tato funkce má pět parametrů, první čtyři parametry specifikují část okna, do kterého se bude vykreslovat – tedy parametry funkce `glViewport` v pixelech. Pátým parametrem funkce `drawView` je pozice oka uživatele 3D monitoru v souřadnicích scény.

Z těla `drawView` je volána funkce `projection`, která má jeden parametr a to opět pozici oka pozorovatele v souřadnicích scény. Funkce `projection` vypočte a nastaví projekční matici a posune scénu tak, aby vrchol pohledového jehlanu byl na pozici oka. Nastavování projekční matice je v *OpenGL* usnadněno tím, že v něm je implementována funkce `glFrustum`. Parametry této funkce odpovídají proměnným v projekční matici uvedené v kapitole 2 (popis `glFrustum` viz. [8, Reference Pages]).

Po nastavení správné projekční matice a pozice kamery je vykreslena scéna.

V *OpenGL* je implementována podpora pro stereoprojekci. Snímky pro levé a pravé oko lze vykreslit zvlášť, pomocí tzv. *color bufferů*. Pokud je k tomu uzpůsoben grafický adaptér, pak se snímky budou vykreslovat odděleně, každý na zvláštní video rozhraní. Toho jsem využil k stereoprojekci pomocí polarizovaného světla projektorů, kterou jsem popsal v podkapitole 5.2.

Popis scény

Scéna je tvořena z pěti stran otexturovaným kvádrem, jehož přední strana je tvořena zobrazovací plochou. Tato strana není uzavřena a uživatel 3D monitoru tak může nahlédnout dovnitř. Dále jsou ve scéně tři modely čajové konvice, které jsou k dispozici v knihovně *OpenGL*. První konvice se (v souřadnicích scény) nachází před zobrazovací plochou, druhá pak za ní – uvnitř kvádrů. Třetí otexturovaná konvice se periodicky pohybuje od zadní stěny kvádrů směrem k pozorovateli. Snímek scény je zachycen na obrázku 5.3.

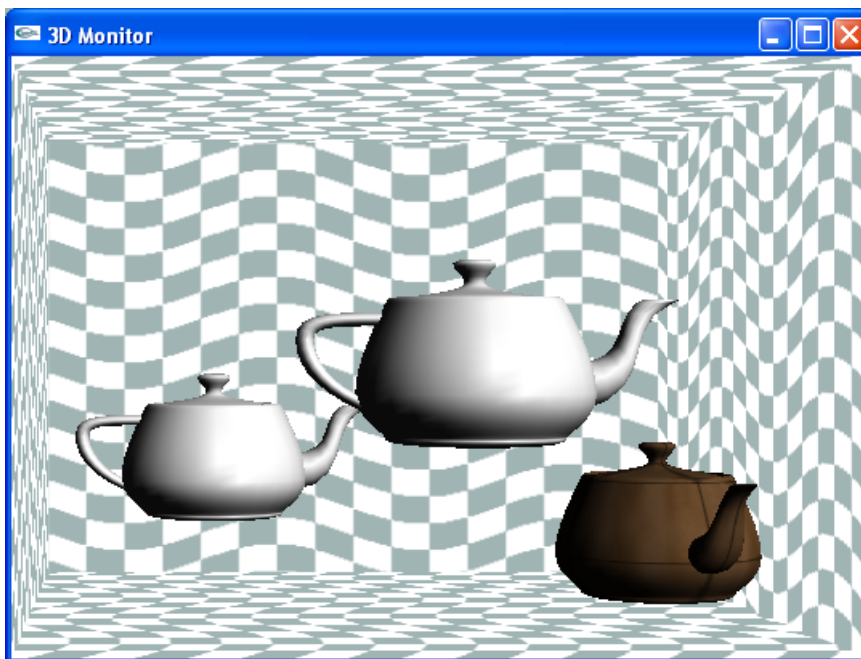
Scénu jsem se snažil navrhnout tak, aby uživateli co nejlépe navodila prostorový vjem.

5.5 FPS Aplikace

Druhou aplikaci kterou jsem vytvořil je tzv. *First Person Shooter (FPS)* hra. Kromě pozorování scény je v této hře umožněno uživateli i interagovat se zobrazenou scénou. Interakce je umožněna zbraní, ze které vystupuje ve scéně zobrazený proud částic. Zbraň je tvořena dvojicí senzorů *Nest of Birds* upevněných na krátké tyči. Úkolem uživatele je pak zasáhnout co nejvíce modelů trpaslíků, kteří se přibližují z pozadí scény směrem k zobrazovací ploše. Se vzrůstajícím počtem sestřelených trpaslíků roste obtížnost hry. Trpaslíci s větší obtížností se před proudem paprsku mohou několikrát krýt. Počet krytí závisí na úrovni obtížnosti.

Grafický engine Irrlicht

Program byl napsán s využitím grafického engine *Irrlicht*. Tento engine jsem zvolil pro jeho jednoduchost a také pro to, že je v něm již vytvořen jednoduchý detektor kolizí.



Obrázek 5.3: Zobrazovací okno 3D monitoru implementovaného pomocí OpenGL.

Irrlicht ovšem nepodporuje stereopromítání. Musel jsem tedy vytvořit dvě identické scény, které jsem zobrazil do dvou samostatných oken. Každá scéna je snímána z jiného pohledu – jedna z pohledu levého a druhá z pohledu pravého oka. Tyto scény jsem zobrazil přes sebe pomocí projektorů s polarizačními filtry. Obrazy pro každé oko zvlášť je možné oddělit speciálními polarizačními brýlemi. Zařízení umožňující stereozobrazování je popsáno výše v podkapitole 5.2.

Popis kódu

Kód jsem, podobně jako u aplikace napsané v OpenGL, strukturoval do několika modulů. Pro práci s Nest of Birds je použit stejný modul jako v první aplikaci.

Dalším modulem je třída `ChangeMatrix`, ve které se nachází implementace práce spojené se změnou souřadného systému. Konstruktor této třídy má 3 parametry typu `float[3]`. Tyto parametry jsou body získané měřením bázových vektorů scény v souřadnicích zařízení Nest of Birds. První bod je počátek souřadného systému scény. Druhý parametr je bod na ose X a třetí na ose Y systému scény. Body jsou na osách umístěny ve vzdálenosti rovnající se velikosti bázového vektoru jednotlivé osy. Po zavolání konstruktoru se vypočítá matice přechodu, pokud se výpočet zdaří (indikuje to vlastnost třídy `invExist`) je možné převádět mezi systémy pomocí metody `changeCoords` s dvěma parametry typu `float[3]`. Prvním (vstupním) parametrem je bod v souřadnicích Nest of Birds. Druhý (výstupní) parametr je převedený bod v souřadnicích scény.

Dále jsem vytvořil třídu `Dwarf` implementující chování modelu trpaslíka. Toto chování jsem rozdělil do několika stavů jako chůze, boj a podobně. V každém cyklu je objektu této třídy zaslána zpráva metodou `computeState`. Tato metoda slouží ke kontrole, zda-li má trpaslík setrvat ve svém současném stavu nebo zda má své chování změnit. Pokud dojde k sestřelení trpaslíka je objektu třídy `Dwarf` zaslána zpráva `shootDown`.

Popis scény

Scéna je tvořena modely terénu, oblohy, domu a dvou stromů. Kromě těchto statických modelů jsou ve scéně ještě pohybující se modely již zmíněných trpaslíků (všechny modely jsou dostupné k volnému použití – zdroji těchto modelů jsou Google 3D Warehouse, psionic3d.co.uk a tutoriál k enginu Irrlicht). Proud částic vylétávajících ze zbraně je simulován pomocí funkce *particle emitter*, která je implementována v použitém enginu Irrlicht.

Ve scéně lze zobrazit i souřadný systém pomocí trojice úseček – viz. podkapitola 4.2. Při použití projektorů je však dosah magnetického pole Nest of Birds příliš krátký na to, aby byly báze vektory zobrazeny na svém reálném místě. Jsou tedy zkráceny a zobrazeny v dolní části ve středu plátna, kde jsou změřeny. Přepočtem je pak změněna poloha bodů tak, jako kdyby byly změřeny na správném místě – uprostřed plátna.

Snímek scény můžete vidět na obrázku 5.4.



Obrázek 5.4: Snímek FPS aplikace.

Kapitola 6

Testování

V této kapitole popíšu jak jsem testoval aplikace popsané v kapitole 5. Provedené testy byly spíše subjektivního charakteru.

6.1 Testování aplikace v OpenGL

Výslednou aplikaci jsem testoval tak, že jsem se dotazoval uživatelů na subjektivní dojem užívání 3D monitoru. Mými respondenty byli můj vedoucí práce a několik jeho kolegů z Ústavu počítačové grafiky a multimédií.

Bohužel se u respondentů nepodařilo navodit realistický 3D vjem ze zobrazované scény. Tento poměrně závažný nedostatek přikládám několika faktorům.

Prvním z nich je pravděpodobně nedostatečná rychlost reakce na změnu polohy pozorovatele. Systém na tuto změnu nezareaguje dostatečně včas na to, aby uživatel nezaznamenal krátkou prodlevu. Prodleva je nejspíše způsobena zařízením Nest of Birds, které sice dává informaci o poloze senzoru ± 100 krát za sekundu, ale s krátkým zpožděním.

Druhým důvodem může být to, že při měření souřadného systému scény nebyla naměřená informace dostatečně přesná na to, aby člověk vnímal obraz na zobrazovací ploše jako třídimenzionální.

V rámci testování aplikace bylo natočeno video tak, že místo pozorovatele byla snímána poloha videokamery, která natáčela scénu 3D monitoru. Video je přiloženo na DVD k práci. Výsledný prostorový dojem z tohoto videa působí poměrně realisticky, i když jde pouze 3D vjem zprostředkovaný – vjem který není vyvolán přímo uživateli 3D monitoru.

Závěrem lze tedy říci, že ač zobrazení nevyvolává v uživateli přímo 3D vjem, ověřil jsem testováním, že mé poznatky, které jsem nashromáždil v předešlých kapitolách jsou správné. Správnost zobrazení potvrzuje natočené video.

6.2 Testování FPS aplikace

Aplikaci FPS jsem testoval podobně, jako výše uvedenou, aplikaci v OpenGL – zaměřil jsem se na subjektivní dojmy ze zobrazené scény. I výsledky tohoto testování se nijak neliší od výsledků testování aplikace v OpenGL.

Kromě subjektivních dojmů jsem provedl i test přesnosti míření zbraně. Svůj pohled jsem zaměřil na místo ve scéně a do směru tohoto pohledu jsem umístil zbraň. Mé oko, dva body zbraně a bod na který jsem mířil tedy tvořily jednu přímku. Posuzoval jsem přesnost dopadu částic vylétávajících ze zbraně na cílový bod. Test ukázal, že míření je poměrně

přesné a výpočty, kterými aplikace určuje směr proudění částic, jsou tedy správné. Drobné odchylky mohou být způsobeny nepřesností měření bázových vektorů scény.

Poměrně závažným problémem při testování aplikace se ukázal být značně krátký rozsah magnetického pole, ve kterém je možno měřit souřadnice senzorů. Rozsah standardního vysílače magnetických impulsů neumožňuje praktické použití systému, při zobrazování na tak velké ploše jakou je promítací plátno.

Snímek z praktického testu můžete vidět na obrázku 6.1. Video zaznamenávající testy je přiloženo na DVD k práci.



Obrázek 6.1: Test míření zbraně na modely trpaslíků. (Snížená kvalita snímku je způsobená nedostatečným osvětlením. K zobrazení scény pomocí projektorů je potřeba zatemnit místnost.)

Závěr

Mým cílem, který jsem si vytyčil při definici pojmu 3D monitor (viz. podkapitola 1.2), bylo vytvořit takový systém, který by uživateli pomocí dostupných zobrazovacích zařízení zprostředkoval prostorový vjem zobrazených objektů. Tento můj cíl se mi nepodařilo úplně splnit, protože člověk používající 3D monitor nemá pocit, že sleduje reálné 3D objekty.

Důvodů proč tomu tak je, může být několik. Nejzávažnějším z nich se zdá být nedostatečná rychlost reakce zařízení Nest of Birds na změnu pozice uživatele. Tento problém bohužel nelze vyřešit jinak, než navržením jiného způsobu získávání polohy pozorovatele. Řešením by mohlo být využití optických metod získávání polohy. Kamerový systém v součinnosti s počítačem by mohl mít požadovanou reakci rychlejší. Navíc by takovýto systém snížil cenu 3D monitoru, protože cena kamer je v porovnání s cenou zařízení Nest of Birds nižší. Optickými metodami lokalizace objektů se ve své diplomové práci [6] letos zabýval Marek Šolony.

Kromě rychlosti reakce na změnu polohy senzoru je další nepříjemnou vlastností Nest of Birds, krátký rozsah ve kterém je možno měřit. Tento problém by opět mohl být řešen optickými metodami nebo zakoupením rozšířeného vysílače magnetických impulsů.

Celkově spatřuji využití kamer pro lokalizaci jako výhodné řešení pro 3D monitor. Pokud by existoval takový systém, který by byl schopen dostatečně rychle lokalizovat polohu lidského oka, pak by 3D monitor jistě našel uplatnění například jako prezentační nebo herní zařízení.

Literatura

- [1] *Dot Products and Projections* [online]. <http://www.math.oregonstate.edu/home/programs/undergrad/CalculusQuestStudyGuides/vcalc/dotprod/dotprod.html>, 1996 [cit. 2009-05-12].
- [2] *OpenGL Projection Matrix* [online]. http://www.songho.ca/opengl/gl_projectionmatrix.html, [cit. 2009-04-01].
- [3] Horčík, R.: *Výpočet matice rotace* [online]. <http://www2.cs.cas.cz/~horcik/Teaching/applications/node3.html>, 2009-01-04 [cit. 2009-03-30].
- [4] Kooima, R.: *Generalized Perspective Projection* [online]. <http://aoeu.snth.net/?p=43>, 2008-08-28 [cit. 2009-04-06].
- [5] Lancaster, P.; Tismenetsky, M.: *The theory of matrices: with applications*. New York: Academic Press, druhé vydání, 1985.
- [6] Šolony, M.: *Lokalizace objektů v prostoru* (Diplomová práce). Brno : VUT, 2009.
- [7] Penner, E.; Parker, J. R.: *The virtual window simulator*. 2007, doi:<http://doi.acm.org/10.1145/1328202.1328213>.
- [8] *OpenGL Software Development Kit Documentation* [online]. <http://www.opengl.org/sdk/>, [cit. 2009-05-1].
- [9] *Viewing frustum* [online, Wikipedia]. http://en.wikipedia.org/wiki/Viewing_frustum, 2008-11-15 [cit. 2009-04-01].

Příloha A

Obsah DVD

- Složka *video* obsahující natočené ukázky práce s vytvořenou aplikací.
- Složka *projects* s projekty aplikací ve Visual C++.
- Složka *applications* s přeloženými výstupními aplikacemi.
- Tento text ve formátu PDF.
- Plakát ve formátu PDF.

Příloha B

Manuál k práci s Nest of Birds

Postup při práci se zařízením Nest of Birds je následující:

1. Po připojení a zapnutí hlavního vypínače zařízení, je potřeba spustit program winBIRD dodávaný na CD s přístrojem.
2. Poté je nutné navázat komunikaci s přístrojem prostřednictvím programu winBIRD. Provedeme to tak, že v záložce SetUp vybereme rozhraní připojení (RS-232 pro zařízení na FIT). Specifikujeme počet senzorů (4) a port, na kterém je zařízení připojeno.
3. Po správné inicializaci je potřeba vybrat v záložce *Options* možnost *System options*, kde nastavíme mód vysílače na *Cool-Down mode*.
4. Nyní je možno uzavřít program winBIRD a používat mnou napsaný modul – a tedy i aplikace jež jsem v rámci své bakalářské práce vytvořil.

Příloha C

Manuál k aplikaci v OpenGL

Postup při práci s aplikací napsanou v OpenGL:

1. Nejdříve je potřeba inicializovat Nest of Birds podle postupu v příloze B.
2. Následuje spuštění aplikace a volba portu zařízení.
3. Klávesou 'f' je možné okno aplikace roztáhnout na celou obrazovku.
4. Následně musí být změřeny souřadnice bázových vektorů. Měření báze:
 - (a) Klávesou 's' zobrazíme bázi.
 - (b) Přiložíme senzor č. 1 na horní konec vertikální úsečky a stiskneme klávesu 't'.
 - (c) Přiložíme senzor č. 1 na společný bod úseček a stiskneme klávesu 'b'.
 - (d) Přiložíme senzor č. 1 na pravý konec horizontální úsečky a stiskneme klávesu 'r'.
 - (e) Stiskneme klávesu 'u' a obraz začne reagovat na změnu polohy senzoru č. 1.
 - (f) Klávesou 's' skryjeme bázi.
5. Pro ukončení práce stiskneme klávesu 'q'.

Upozorňuji, že při měření báze v blízkosti zobrazovací plochy monitoru nejsou výsledky relevantní pro úspěšný chod aplikace. Východiskem je vypnutí monitoru při měření.

Příloha D

Manuál k FPS aplikaci

Po spuštění aplikace ještě nereaguje na pohyby se senzory Nest of Birds. Pro správný chod aplikace je potřeba dodržet tento postup:

1. Před spuštěním aplikace je třeba správně inicializovat zařízení Nest of Birds podle postupu v příloze B.
2. Poté následuje spuštění aplikace a zvolení portu na kterém je Nest of Birds připojeno.
3. Poté musí být okna přeneseny na odpovídající obrazovky a maximalizovány. Okno pro levé oko má nadpis *Device - L* a pro pravé *Device - R*.
4. Nyní je nutné změřit bázi souřadného systému scény. Provedeme to následovně:
 - (a) Stiskneme na klávesnici klávesu 's' pro zobrazení báze scény.
 - (b) Přiložíme senzor číslo 1 (první zleva při pohledu zezadu na přístroj) na horní konec vertikální úsečky a stiskneme klávesu 'šipka nahoru'.
 - (c) Tentýž senzor přiložíme na průsečík úseček a stiskneme klávesu 'šipka dolů'.
 - (d) Opět přesuneme senzor č. 1. Tentokrát na pravý konec horizontální úsečky a stiskneme klávesu 'šipka vpravo'.
 - (e) Stiskneme klávesu 'enter' a systém začne reagovat na změnu polohy senzoru č. 1 překreslováním pohledu na scénu. Proud částic začne reagovat na změnu polohy senzorů č. 3 a 4, které měří polohu zbraně.
 - (f) Klávesou 's' skryjeme bázi scény.
5. Klávesou 'mezerník' spustíme hru a toutéž klávesou můžeme hru kdykoli restartovat.
6. Pro ukončení aplikace stiskneme klávesu 'q'.

Aplikaci lze ovládat i myší bez použití Nest of Birds.