

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

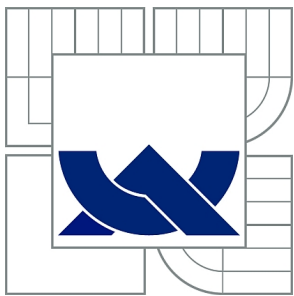
VYTVOŘENÍ PODPŮRNÝCH VÝUKOVÝCH PROSTŘEDKŮ PRO KURZ
MODERNÍ POČÍTAČOVÁ GRAFIKA

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

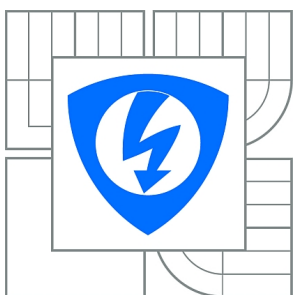
Bc. MAREK SILBER

BRNO 2015



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH
TECHNOLOGIÍ

ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

VYTVOŘENÍ PODPŮRNÝCH VÝUKOVÝCH PROSTŘEDKŮ PRO KURZ MODERNÍ POČÍTAČOVÁ GRAFIKA

CREATING 3D MODELS SUPPORTING MODERN COMPUTER GRAPHICS COURSE

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. MAREK SILBER

VEDOUCÍ PRÁCE

SUPERVISOR

Mgr. PAVEL RAJMIC, Ph.D.

BRNO 2015



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav telekomunikací

Diplomová práce

magisterský navazující studijní obor
Telekomunikační a informační technika

Student: Bc. Marek Silber

ID: 165844

Ročník: 2

Akademický rok: 2014/2015

NÁZEV TÉMATU:

Vytvoření podpůrných výukových prostředků pro kurz Moderní počítačová grafika

POKYNY PRO VYPRACOVÁNÍ:

Zorientujte se v problematice vyučované v předmětu MGMP (Moderní počítačová grafika). Vyhodnoťte, který 3D modelovací software by hrál nejlépe roli prostředníka mezi teorií 3D modelování a mezi programování grafiky v jazyku OpenGL. Po dohodě s garantem vytvořte podpůrné materiály zejména v podobě video-tutoriálů pro vybraný software. Půjde zejména o geometrické transformace, osvětlovací modely, projekční transformace a texturování. Praktické návody doprovodte příslušnou teorií.

DOPORUČENÁ LITERATURA:

[1] Jiří Žára, Bedřich Beneš, Jiří Sochor, Petr Felkel, Moderní počítačová grafika, druhé vydání, Computer Press, 2005, ISBN 80-251-0454-0

[2] Webová stránka software <http://www.blender.org/>

[3] Schneider, P.J., Eberly, D.H., Geometric tools for computer graphics. Elsevier, 2003.

[4] Hughes, J.F. a kolektiv, Computer graphics, principles and practice. Třetí vydání, Addison-Wesley, 2013.

Termín zadání: 9.2.2015

Termín odevzdání: 26.5.2015

Vedoucí práce: Mgr. Pavel Rajmic, Ph.D.

Konzultanti diplomové práce:

doc. Ing. Jiří Mišurec, CSc.

Předseda oborové rady

ABSTRAKT

Práce je rozdělena do dvou částí, kde první část se zabývá přehledem výuky a práce v 3D softwaru na českých univerzitách včetně srovnání jednotlivých programů. Popisem historie programu Blender, který byl vybrán pro podporu výuky předmětu Moderní počítačová grafika. V druhé části práce, která je více prakticky zaměřená je popsán základní ovládání programu od prvního spuštění a orientaci v jednotlivých sekcích programu a nastavení pracovního prostředí pro samostatnou práci. V práci jsou popsány základní transformační příkazy změny polohy, rotace a změnu měřítka objektů a editační příkazy jako je kopírování, mazání, skrývání a duplikace jednotlivých objektů včetně jejich teoretických popisů. Na transformační příkazy objektů navazuje práce se světlem včetně praktických ukázek jednotlivých druhů osvětlení. V posledních dvou kapitolách je popsána práce s materiály a texturami. Cílem této práce je popsat začátečníkům programu Blender orientaci v programu a elementární práci s objekty a přiřazení jednotlivých materiálů a textur jednotlivým objektům a jejich nasvícení ve scéně.

KLÍČOVÁ SLOVA

3D model, osvětlení, Blender, grafika, modelování, tutoriál, promítání, materiál, textura

ABSTRACT

This thesis is divided into two parts, where the first part is focused on an overview and work with 3D software at Czech universities, it includes a comparison of the different programs and description of history of Blender programme which was the one chosen as support for teaching the Modern computer graphics course. The second part of the thesis is more focused on practical matters and describes the basic workings of the programme from the first start up, orientation in the different sections and the settings of the work space for the work itself. The thesis details basic transformation commands of change of location, rotation and change of scale of objects and editing commands like copying, deleting, hiding and duplication of individual objects including their theoretical descriptions. Work with lightings continues after the transformation commands including practical examples of different types of lighting. Work with materials and textures is described in the last two chapters. The goal of this thesis is to describe how to work with the Blender programme and basic work with objects and assigning individual materials and textures for objects and the lighting in a scene for people unfamiliar with the programme.

KEYWORDS

3D model, lighting, Blender, graphics, modeling, tutorial, projection, material, texture

SILBER, Marek *Vytvoření podpůrných výukových prostředků pro kurz moderní počítačová grafika*: diplomová práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2015. 70 s. Vedoucí práce byl Mgr. Pavel Rajmic, PhD.

PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Vytvoření podpůrných výukových prostředků pro kurz moderní počítačová grafika“ jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno

.....

(podpis autora)

PODĚKOVÁNÍ

Děkuji vedoucímu diplomové práce Mgr. Pavlovi Rajmicovi, Ph.D. za velmi užitečnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování diplomové práce.

Brno

.....

(podpis autora)

OBSAH

Úvod	10
1 Přehled výuky 3D grafiky na Českých univerzitách	11
2 Základní specifikace programů	13
2.1 Zhodnocení jednotlivých programů	14
3 Vývoj programu Blender	15
4 Základní ovládání Blenderu	17
4.1 Rozhraní po prvním spuštění	17
4.2 Základní práce s okny	19
4.2.1 Nastavení oken	19
4.3 Základní práce s oknem 3D view	20
4.3.1 Rozdělení okna 3D view	20
5 Vytváření a editace objektů	23
5.1 Transformace ve 2D	23
5.2 Transformace a souřadnicové systémy	26
5.3 Homogenní souřadnice	27
5.4 3D transformace	28
5.5 Vkládání objektů do scény	29
5.6 Objektový a editační mód	30
5.6.1 Základní editační příkazy objektového módu	31
5.7 Příkazy transformací	32
5.7.1 Manipulátory	33
6 Promítání	35
6.1 Kamera	36
6.2 Rovnoběžné promítání	37
6.3 Středové promítání	37
7 Osvětlovací metody	39
7.1 Světlo	39
7.1.1 Radiometrické pojmy	39
7.1.2 Radiance	40
7.2 Dvousměrová odrazová distribuční funkce – BRDF	41
7.2.1 Vlastnosti BDRF	42

7.3	Phongův osvětlovací model	43
7.4	Světelné zdroje	45
7.4.1	Bodový světelný zdroj	45
7.4.2	Plošný zdroj	46
7.4.3	Rovnoběžný zdroj	47
7.4.4	Reflektor	48
8	Materiály	49
8.1	Materiálové nastavení	49
8.2	Typy stínování	50
8.3	Nastavení více materiálů jednomu objektu	51
9	Textury	53
9.1	Mapování textur	53
9.1.1	Inverzní mapování válcové a kulové plochy	53
9.2	Mapování prostředí (environment mapping)	55
9.3	Hrbolaté textury (bump texture)	57
9.4	UV mapování	58
10	Závěr	63
	Literatura	65
	Seznam symbolů, veličin a zkratk	68
	Seznam příloh	69
A	Obsah přiloženého DVD	70

SEZNAM OBRÁZKŮ

4.1	Zobrazení Blenderu po spuštění	17
4.2	Okno předvoleb	18
4.3	Ikona pro práci s okny	19
4.4	Nastavení předvoleb pro okno	20
4.5	Grafické 3D okno v programu Blender po spuštění	22
4.6	Okna vznikla rozdělením jednoho hlavního. Nastavení kamera (<i>Camera</i>), nárys (<i>Front</i>), pohled shora (<i>Top</i>), pohled zprava (<i>Right</i>) . . .	22
5.1	Transformace rotace	24
5.2	Transformace zvětšení	25
5.3	Transformace deformace	25
5.4	Transformace zkosení	26
5.5	Dva rozdílné souřadnicové systémy v \mathbf{R}^2 , vektor \mathbf{u} má v červeném systému souřadnice $[3, 2]$, kdežto souřadnice v modrém systému jsou přibližně $[-0, 2; 3, 6]$ [10].	27
5.6	Tlačítko pro přidání objektu do scény	29
5.7	Nabídka pro vložení nových objektů	30
5.8	Menu pro příkazy transformací	32
5.9	Možnosti vztažného bodu pro transformace	33
5.10	Manipulátory	34
6.1	Rozdíl mezi souřadnicovými systémy WCS a VCS. Na obrázku chybí průmětna [21].	35
6.2	Vlastnosti rovnoběžného promítání [11]	37
6.3	Vlastnosti středového promítání [20]	38
7.1	Dvousměrová odrazová distribuční funkce (BRDF)	42
7.2	Phongův osvětlovací model [14]	43
7.3	Vliv koeficientu h na velikost odlesku pro hodnoty 10, 20, 50, 100 a 1000	44
7.4	Panel <i>Preview</i> a <i>Material</i> pro práci s materiály	46
7.5	Nasvícení objektu pomocí bodového světelného zdroje s lineárním úbytkem intenzity	46
7.6	Nasvícení objektu pomocí plošného směrového zdroje	47
7.7	Nasvícení objektu pomocí světelného zdroje typu <i>Sun</i> a <i>Hemi</i>	47
7.8	Nasvícení objektu pomocí světelného zdroje typu <i>Spot</i>	48
8.1	Okno předvoleb pro nastavení materiálu	49
8.2	Jednotlivé typy difúzního stínování. Zleva <i>Fresnel</i> , <i>Minnaert</i> , <i>Toon</i> , <i>Lambert</i> , <i>Oren-Nayar</i> s intenzitou 0,5.	51

8.3	Jednotlivé typy spekulárního stínování. Zleva <i>Cook Torr</i> , <i>Phong</i> , <i>Blinn</i> , <i>Toon</i> , <i>WardIso</i> s intenzitou a tvrdostí nastavenou na hodnotu 0,5.	51
8.4	Přiřazení více materiálů jednomu objektu	52
9.1	Ikona pro textury v panelu předvoleb	54
9.2	Základní nastavení textur	55
9.3	Princip mapování prostředí. Převzato z [23]	56
9.4	Nastavení Blenderu pro mapování prostředí	57
9.5	Výsledek mapování prostředí	58
9.6	Nastavení 3D efektu pro bump mapping	59
9.7	Původní textura cihel vlevo a vpravo textura bump mapping	59
9.8	Vzhled 3D okna pro UV mapování	60
9.9	Menu pro UV mapování	60
9.10	Krychle a její UV mapa	61
9.11	Nakreslená textura krychle vpravo a obrázek finálního vykreslení	62

ÚVOD

Počítačová grafika je v posledních letech velmi rozšířeným oborem, který prošel velkým vývojem. Jedná se o relativně mladý obor, který se rozvinul především s rozšířením počítačů. S moderní počítačovou grafikou se můžeme setkat v mnoha oblastech jako je například reklamní a umělecká tvorba, konstrukční návrhy (CAD), tvorby webů, úpravy fotografií a další. Pro filmovou tvorbu nebo počítačové hry se v dnešní době stále více využívá 3D počítačové grafiky. Tato oblast umožňuje vytvořit a nasimulovat velmi reálné objekty, které by bylo finančně náročné vytvořit. Stejně jako v jiných oblastech při práci na počítači se používají specializované programy, je tomu tak i v 3D počítačové grafice. Jedním programem na 3D počítačovou grafiku se budeme zabývat i my v diplomové práci.

Diplomová práce Vytvoření podpůrných výukových prostředků pro kurz Moderní počítačová grafika je rozdělena na dvě části. V první části je srovnání výuky o grafice na českých univerzitách s praktickými cvičeními v některém z komerčně dostupných 3D softwarů. Programy jsou následně srovnány a popsány jejich výhody a vhodnost praktického využití.

V druhé části práce je popsáno základní ovládání programu Blender, který byl vybrán pro podporu praktických cvičení v předmětu MGMP – Moderní počítačová grafika. Teoretické základy jednotlivých kapitol jsou doplněny o praktické ukázky v programu Blender. Vybrané kapitoly obsahují i videotutoriály, které jsou určeny začátečníkům, kteří nemají žádné zkušenosti s modelováním 3D objektů. Program byl vybrán z důvodu, že licence je freeware, je založen na knihovnách OpenGL, což je hlavní náplní předmětu Moderní počítačová grafika. Program je relativně výpočetně nenáročný ve srovnání s ostatními 3D programy, podporuje platformy Windows, Mac OS X, Linux a další.

Klávesové zkratky jsou značeny tučným textem (např. **Tab**), navigace po jednotlivých funkcích programu jsou značeny kurzívou (např. *File – User Preferences*).

1 PŘEHLED VÝUKY 3D GRAFIKY NA ČESKÝCH UNIVERZITÁCH

Masarykova univerzita, Fakulta informatiky – Bakalářský program Počítačová grafika a zpracování obrazu Cílem předmětu Základy počítačové grafiky [7] je poskytnout přehled o základních algoritmech a metodách používaných při tvorbě modelů a jejich vykreslování. Studenti získají znalosti, které jim umožní orientaci v počítačové grafice včetně nezbytného matematického základu. Modelování prostorových těles probíhá pomocí softwaru Cinema 4D. V programu se vyučuje práce s křivkami (NURBS, Bézier), rovnoběžné a perspektivní promítání, osvětlovací metody jednotlivých těles a celých scén až stínovací techniky a odrazy světla.

Univerzita Tomáše Bati ve Zlíně, Fakulta aplikované informatiky – Bakalářský program Inženýrská informatika V rámci předmětu Počítačová grafika [6] si studenti osvojí teoretické principy základních grafických algoritmů v oblasti 2D a 3D grafiky. Výuka probíhá ve freeware programech jako je Gimp pro 2D a Blender pro 3D modelování. V programu Blender jsou demonstrovány různé druhy modelování a reprezentace 3D objektů a dále texturování a osvětlení celých scén.

Univerzita Tomáše Bati ve Zlíně, Fakulta aplikované informatiky – Bakalářský program Inženýrská informatika Student předmětu Počítačová grafika II [6] získá teoretické znalosti i praktické dovednosti z oblasti 3D počítačové grafiky. Během cvičení se studenti seznámí s 3D grafickým programem Blender, ve kterém se studenti získané znalosti naučí prakticky používat. Student má obecné znalosti o matematických principech 3D grafických systémů. Dále dokáže používat program Blender, ve kterém je schopen vytvářet komplexnější 3D grafické scény a animace. V předmětu se probírá pokročilá tvorba animací včetně použití částicových systémů a efektů. A v neposlední řadě je vysvětlené využití Python skriptů.

VŠE v Praze, Fakulta informatiky a statistiky – Bakalářský program Multimédia v ekonomické praxi Předmět Principy a aplikace 3D grafiky [22] seznamuje studenty s teorií a praxí 3D počítačové grafiky, kde je hlavní náplní předmětu vysvětlit obecné principy a postupy tvorby 3D grafických prvků s jejich následnou praktickou možností využití. Jsou zde podrobně vysvětleny způsoby a techniky modelace včetně aplikace textur a osvětlení jednotlivých modelů a celých scén. V předmětu je probírána teorie a praxe od kompozice scén, nasvícení modelů, modelování architektury a lidských postav až po animace. Pro praktickou část je použit software 3Ds Max od společnosti Autodesk.

ČVUT v Praze, Fakulta elektrotechnická – Volitelný předmět v bakalářských oborech Kurz Multimediálních aplikací [4] je pojat jako kurz aplikace Maya od společnosti Autodesk, která se používá v oblasti produkce, post-produkce, architektury a dalších oblastech zabývajících se 3D grafikou. V kurzu se studenti naučí ovládání programu Maya a hlavní techniky modelování jako je polygonální modelování, operace nad vrcholy, hranami a tělesy, modelování postupných dělení, modelování pomocí NURBS. Prostor je věnován pokročilejší práci s UV a renderingu jako např. MentalRay a víceprůchodový render. U animace scén je zahrnuta práce s dynamikou, particles a inverzní kinematikou.

ČVUT v Praze, Fakulta stavební – Volitelný předmět v bakalářských oborech Předmět Cinema 4D [8] probírá základní postupy tvorby v prostředí grafického programu Cinema 4D. Studenti se seznámí s různými druhy modelování jako je box modeling, mesh editing, NURBS nebo třeba Booleovské operace. V metodách texturování jsou probírány základy Body Paintu.

VUT v Brně, Fakulta strojního inženýrství – Povinný předmět v bakalářských oborech Cílem předmětu Počítačová geometrie a grafika je seznámit studenty se základy projektivní geometrie a počítačové grafiky, která se používá v CAD systémech. Nabyté teoretické znalosti jsou propojeny s praktickým použitím. Praxe probíhá v 3D softwaru Rhinoceros. Narozdíl od ostatních vyučovaných předmětů na ostatních univerzitách se zde probírá spíše geometrická interpretace křivek, ploch a těles. V rámci cvičení jsou v softwaru vysvětleny základní principy práce se základními objekty v rastrovém zobrazení. Na toto navazuje přesné modelování, jako je tvorba technických součástí a jejich následné osvětlení a rendering, případně animace [5].

2 ZÁKLADNÍ SPECIFIKACE PROGRAMŮ

Autodesk 3Ds Max Je celosvětově nejrozšířenějším vizualizačním a animačním softwarem. Je určen pro tvorbu vizuálních efektů, animaci postav a tvorbu počítačových her. Nabízí interaktivní prostředí a rychlý rendering. Plně přizpůsobitelná a otevřená architektura poskytuje absolutní tvůrčí svobodu. Pro 3Ds Max je k dispozici rozsáhlá knihovna doplňkových plug-in aplikací dalších vývojářů. Mezi uživatele patří například DreamWorks, Blizzard, UbiSoft a jiní. Nevýhodou tohoto software je jeho cena, která se pohybuje okolo 77 350,- Kč za standardní verzi, respektive 32 500,- Kč v rámci ročního předplatného. (Ceny za aktuální verzi 3Ds Max 2015)[1]. Program není lokalizován do českého jazyka, ale lze nalézt velké množství tutoriálů v českém jazyce, neboť má relativně velkou rozšířenost v české profesionální sféře.

Autodesk Maya Program Maya je další program pro 3D grafiku od společnosti Autodesk (momentálně je nejnovější verze 2015). Velkou výhodou tohoto modelovacího programu je dostupnost RenderMan [18], což je komerční nástroj pro renderování, vytvořený společností Pixar. Dosahuje velmi dobré výstupní kvality. Základní cena Autodesk Maya je 110 000,- Kč. Maya se hodí spíše pro modelování postav a simulování realistických jevů [2].

Blender Jedná se o software založený na grafické knihovně OpenGL. Mezi jeho hlavní výhody patří jeho dostupnost pro velké množství operačních systémů, jako je Microsoft Windows, Linux nebo Mac OS X. Další nespornou výhodou je, že je šířen pod licencí GNU GPL, tudíž se jedná o freeware software. Mezi nevýhody tohoto programu patří složitější uživatelské rozhraní, kde začátečník, který nemá s podobným softwarem zkušenosti se bude trochu problematicky orientovat.

Maxon Cinema 4D Cinema 4D (aktuálně ve verzi R16) je jeden z mála programů, který lze lokalizovat kompletně do českého jazyka. Mezi výhody programu patří možnost importovat 3D modely z konstrukčních CAD aplikací a jednoduchá ovladatelnost programu. Cinema 4D je v České republice nejvíce používaný komerční 3D software. Jeho základní cena se pohybuje okolo 100 000,- Kč [3].

Rhinoceros 3D software, který se využívá zejména v oblastech jako je strojírenství, šperkařství, konstrukce automobilů, lodí a jiné. Modelování zde probíhá hlavně za pomoci tzv. volných tvarů, kde lze vymodelovat jakýkoliv možný tvar, který je následně možné velmi přesně okótovat, a tím vytvořit výkres pro další zpracování. Za pomoci zásuvných modelů je možné dále výkresy zpracovávat pro 3D tisk, případně generovat vizualizace. Program plně podporuje formáty z AutoCADu, Adobe

Illustrator a mnoha dalšími. Další výhodou je česká lokalizace. Základní cena licence se pohybuje okolo 32 000,-Kč [19].¹

2.1 Zhodnocení jednotlivých programů

Pro potřeby předmětu moderní počítačová grafika se nejvíce hodí 3D software Blender. Samozřejmě jeho výhodou je, že je zcela zdarma, podporuje Windows, Linux a Mac OS X a svými možnostmi plně pokrývá rozsah výuky. Náplní předmětu moderní počítačová grafika není výuka v tomto programu, ale spíše podpora pro studenty na doplnění probírané látky. Pokud by se vyučovala 3D grafika komplexněji, bylo by vhodné použít jeden z komerčně dostupných programů. Jako nejvhodnější komerční alternativa se jeví Cinema 4D od společnosti Maxon. Má v ČR, jak již bylo řečeno, velkou uživatelskou profesionální základnu a mezi další výhody patří česká lokalizace. Výhodou programu Blender je založení na grafické knihovně OpenGL, což je náplní předmětu. Proto se budeme zabývat dále v této práci pouze programem Blender a všechny tutoriály v něm budou demonstrovány.

¹Ceny všech programů byli získány v Amerických dolarech a převedeny dle platného kurzu v době psaní práce

3 VÝVOJ PROGRAMU BLENDER

Blender je jeden z nejmladších programů pro 3D grafiku. První vývoj započala holandská firma NatGeo roku 1995 v čele s Tonem Rososendaalem. V roce 1998 byla založena firma "Not a number"(NaN) jako odnož NatGeo jejíž hlavním zaměřením byl rozvoj Blenderu. Ideou bylo vytvářet multiplatformní program na 3D grafiku, který by byl zcela zdarma. V té době šlo o revoluční myšlenku neboť všechny dostupné programy byli otázkou několika tisíc dolarů. V létě roku 2000 byl vydán Blender 2.0, který přidal integrované 3D rozhraní pro vývoj her – game engine. V roce 2002 vzniká nekomerční organizace Blender Foundation, jejímž základním cílem je rozvoj programu jako Open Source projektu. To se podařilo 13. října 2002, kdy je vydán pod licencí GNU General Public Licence [1].

Verze 1.60 Uvolněna verze pro Windows. Do této chvíle byl Blender pouze pro operační systém Linux a FreeBSD s výstupem pouze do OpenGL [13].

Verze 2.30 Verze Blenderu 2.30 obsahuje zcela nové uživatelské rozhraní. Celé rozložení ovládacích prvků bylo přepracováno a změněno. Jedním příkladem může být rozdělení funkcí po jednotlivých blocích, kdy funkce nejsou platné pro editaci celého objektu, ale pro jednotlivé sekce s dostupnější editací na hlubší úrovni. Další novinkou je možnost numerického zadávání souřadnic nebo transformací objektů. Zlepšena byla také funkce tlačítka *zpět*, která stále funguje pouze v editačním módu, ale pamatuje si více operací [15].

Verze 2.40 Hlavními novinkami je vylepšení a předělání animačních nástrojů – nové zobrazovací metody, zrychlení synchronizace pohybů. Verze přináší možnost animovat fyzikální jevy, jako je pohyb tekutin nebo simulace částic včetně statických částic pro tvorbu vlasů a chlupů [12].

Verze 2.5x Verze v řadě 2.5x obsahují podporu 64-bitových procesorů napříč všemi platformami (Windows, Mac OS X a Linux). Grafické rozhraní se dočkalo další aktualizace s novými ikonami a čistším vzhledem, který se může doplnit o různá rozšíření pomocí Python scriptů. Další novinkou je možnost zvolit si klávesové zkratky pro jednotlivé akce a funkce. Vylepšena byla také práce se stětcí. Je možné si uložit parametry jako je velikost, tvrdost, textury a jiné a těm přiřadit klávesovou zkratku.

Verze 2.6x V Blenderu řady 2.6x je mnoho významných novinek, které usnadňují práci. Hlavní novinkou je přidání *Cycles* renderu (od verze 2.63), který využívá GPU

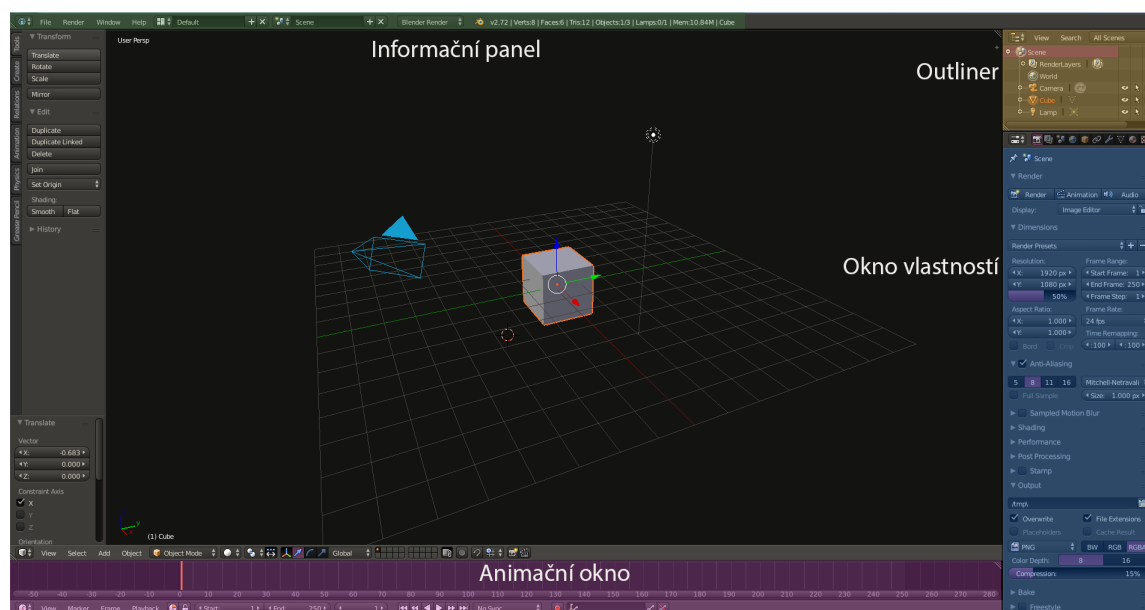
jednotky namísto CPU, a tím docílíme rychlejšího renderingu. Další novou funkcí je *Ocean Simulation*, který generuje mořskou hladinu včetně barvy, chování vln a další. Vylepšen byl také *Smoke Simulation*, do jehož voleb bylo přidáno generování ohně, případně ohně a kouře. Nově se kouř může kolorovat a nemusí být pouze šedý. Velmi užitečným nástrojem je *Bmesh* nástroj, což je mesh systém podporující mnohoúhelníky namísto standardních trojúhelníků a čtverců [16].

Verze 2.7x V posledních verzích jsou změny zaměřeny především na *Cycles* render nástroj, který je neustále s každou verzí vylepšován. Stále je optimalizována rychlost práce renderovacího nástroje. Optimalizací SSE instrukcí bylo dosaženo ve verzi 2.70 až 30% časové úspory v některých složitějších scénách oproti předchozí verzi. Drobnými úpravami prošlo také uživatelské rozhraní, kde byli nástroje zorganizovány do skupin.

4 ZÁKLADNÍ OVLÁDÁNÍ BLENDERU

4.1 Rozhraní po prvním spuštění

Po spuštění Blenderu vypadá výchozí nastavení jako na obrázku 4.1, kde je vidět několik barevných zón, které budou popsány dále včetně jejich funkcí.



Obr. 4.1: Zobrazení Blenderu po spuštění

Zelená zóna reprezentuje *informační panel*, kde kromě hlavní nabídky lze nalézt výběr renderovacího nástroje, verzi programu, kterou používáme a počty bodů (*Verts*), ploch (*Faces*), trojúhelníků (*Tris*) a světel (*Lamps*) ve scéně. Předposlední údaj ukazuje velikost aktuálně zabrané paměti RAM a poslední položka zobrazuje objekt, který je vybrán ve scéně [9].

Žlutá zóna se jmenuje *outliner*, který umožňuje komplexní přehled o scéně. Jsou v něm zobrazeny veškeré informace o každém objektu, který je umístěn ve scéně, včetně jejich vlastností jako je např. materiál nebo textura. Zde si lze nastavit pojmenování jednotlivých objektů (toto doporučuji pro lepší orientaci ve scéně) nebo předvolby, kdy je možné skrýt objekt ve scéně (oko), zakázat výběr daného objektu (šipka) nebo nezahrnovat objekt do renderované scény (ikona fotoaparátu) [9].

Modrá zóna reprezentuje *okno vlastností*, kde je velké množství předvoleb a nastavení. Tyto předvolby se vážou k jednotlivým funkcím zobrazeným na obrázku 4.2.

Pro příklad si zde popíšeme ve zkratce nastavení renderovacího okna. Lze vybírat z několika renderovacích předvoleb rozlišení jako je HDTV 720p, HDTV 1080p, TV NTSC a další. Toto lze dále specifikovat rozlišením, poměrem stran, počtem snímků, stínováním apod. Popsat každou položku není předmětem této práce. Důležitá je pro nás položka *Output*, kde si nastavujeme, kam mají být vyrenderované scény ukládány a formát v jakém chceme mít výstup. Podporovaných výstupů je celá řada např. JPEG, BMP, TIFF, PNG pro statické scény a pro videa jmenujme např. AVI, H.264, MPEG.

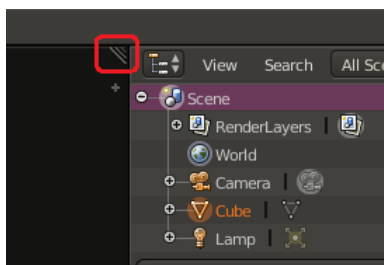


Obr. 4.2: Okno předvoleb

Fialová zóna je *animační okno*, kterému se budeme věnovat podrobněji později. V animačním okně vidíme časovou osu, která zobrazuje počet snímků, nikoliv počet vteřin jak by se na první pohled mohlo zdát. Oranžovou svislou čarou je označen počátek animace v bodě 0 a sekvence končí v bodě 250 (označeno tmavěšedou barvou), což při snímkovací frekvenci 25 fps je 10 s animace. Samozřejmě to neznamená, že lze vytvářet pouze takto krátké animace, ale podrobnějšímu nastavení se budeme věnovat později [9].

4.2 Základní práce s okny

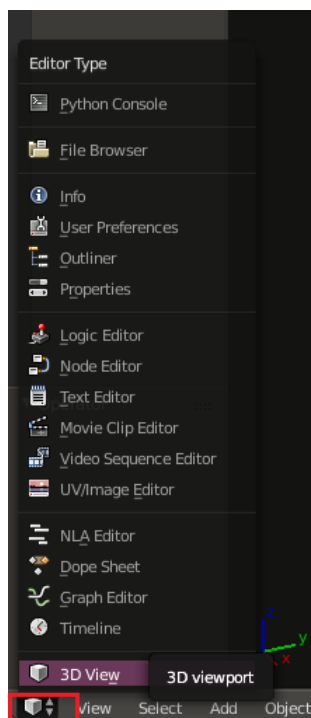
Práce s okny není v Blenderu zcela standardní jako u běžných programů. Zde lze okna modifikovat, přidávat a odebírat. V každém rohu jednotlivých oken je ikona jako na obrázku 4.3, která umožňuje, po stisknutí a držení levého tlačítka myši, okno rozdělit nebo sloučit. Pokud chceme vybrané okno rozdělit neboli přidat další okno, klikneme na ikonu v horním nebo spodním rohu a táhneme myší v horizontálním nebo vertikálním směru, nebo klikneme pravým tlačítkem myši a zvolíme možnost *Split Area*. V případě, že chceme okno odebrat, klikneme myší na již zmíněnou ikonu a při stisknutém levém tlačítku myši táhneme kurzor směrem, kde chceme okno sloučit. Uprostřed okna se utvoří šipka, která naznačuje, které okno budu sloučeno, případně nahrazeno. Druhým způsobem je kliknout na rozhraní dvou oken pravým tlačítkem myši a z nabídky vybrat *Join Areas*. Poté již jen stačí kliknout na okno, které má být sloučeno. Slučovat lze pouze ta okna, která mají stejnou výšku nebo šířku.



Obr. 4.3: Ikona pro práci s okny

4.2.1 Nastavení oken

Pro každé okno lze nastavit předvolby (jsou pro všechny okna stejné), které jsou znázorněny na obrázku 4.4. Tyto nástroje pro jednotlivá okna nám dovolují si zcela modifikovat výchozí rozložení a zobrazení oken podle toho, co každému vyhovuje. Pokud se z nějakého důvodu chceme vrátit k původnímu nastavení, máme dvě možnosti. V případě, že chceme vrátit původní nastavení jednoho konkrétního okna, klikneme pravým tlačítkem myši na ikonu nastavení a zvolíme možnost *Reset to Default Value*. Druhou možností je vrácení celého rozložení jednotlivých oken do původního nastavení, tuto možnost nalezneme v menu *File – Load Factory Settings*.



Obr. 4.4: Nastavení předvoleb pro okno

4.3 Základní práce s oknem 3D view

Po spuštění programu Blender je uprostřed zobrazení grafické okno, které se nazývá *3D view*. Standardně se v okně nachází čtyři objekty, které jsou zobrazeny na obrázku 4.5. Prvním je 3D kurzor, který se umísťuje libovolně po kliknutí levého tlačítka myši ve scéně. Určuje, kam se vkládají nové objekty nebo se k této poloze mohou provádět určité operace. Druhým objektem je kamera, kterou reprezentuje čtyřboký jehlan. Zobrazuje polohu uživatele a směr pohledu na scénu. Trojúhelník představuje orientaci kamery. Dalším objektem je krychle, která je umístěna ve středu souřadnicového systému, tedy v bodě $[x, y, z] = [0, 0, 0]$. Krychle nemá ve scéně žádný konkrétní význam. Posledním objektem je světelný zdroj, utvořen dvojicí kruhů. Standardně se jedná o bodový zdroj světla. Jeho vlastnosti a další typy osvětlení si posíšeme v dalších kapitolách.

4.3.1 Rozdělení okna 3D view

Nyní si 3D view rozdělíme na čtyři podokna a na každé si nastavíme jiný pohled. Měli bychom dosáhnout výsledku jako je na obrázku 4.6. Jednotlivých pohledů lze dosáhnout přes menu *View – výběr pohledu* nebo jednodušší cesta je pomocí klávesových zkratk. Základní pohledové zkratky jsou:

0 - pohled kamerou (Camera)

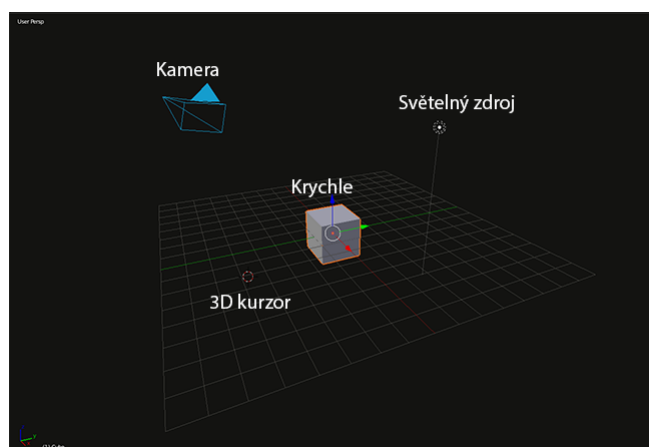
1 - čelní pohled (Front)

3 - boční pohled (Right)

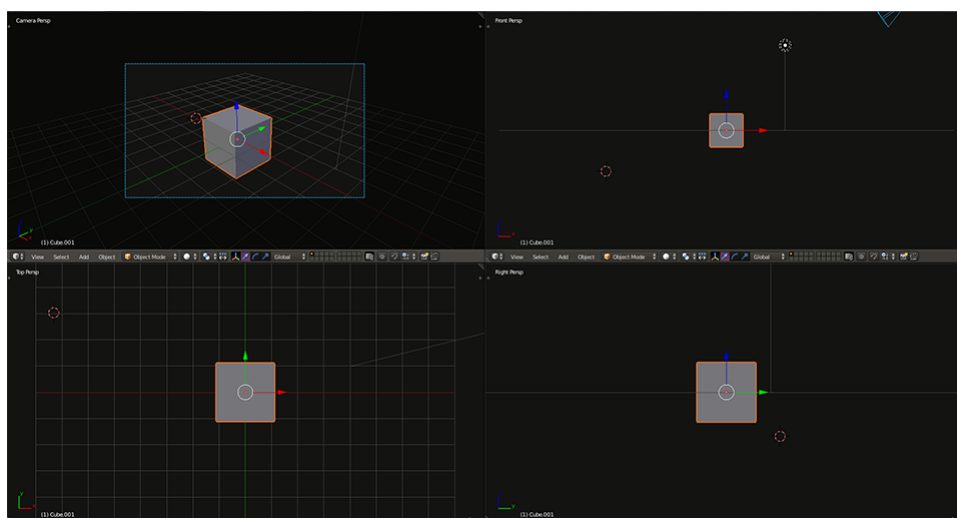
7 - pohled shora (Top)

V každém okně v levém spodním rohu vidíme orientaci jednotlivých os nebo pokud jsou zobrazeny jednotlivé pohledy, jsou zobrazeny osy, které jsou pro daný pohled relevantní.

Manuální rotaci lze provádět pomocí kláves **2, 4, 6, 8**, kdy krok je po 15 stupních. Přibližování a oddalování scény, případně jednotlivých pohledů, je možné pomocí kolečka na myši nebo klávesami "+" a "-" na numerické klávesnici nebo pomocí menu *View – Zoom Border* (klávesová zkratka **Shift+B**). Posun scény je možný pomocí klávesové kombinace Shift a stisknuté kolečko myši a pohybu. Pro vycentrování pohledu na vybraný objekt (objekt se vybírá pomocí pravého tlačítka myši) slouží desetinná čárka na numerické klávesnici.



Obr. 4.5: Grafické 3D okno v programu Blender po spuštění



Obr. 4.6: Okna vznikla rozdělením jednoho hlavního. Nastavení kamera (*Camera*), nárys (*Front*), pohled shora (*Top*), pohled zprava (*Right*)

5 VYTVÁŘENÍ A EDITACE OBJEKTŮ

5.1 Transformace ve 2D

S téměř každým objektem, který vložíme do scény potřebujeme udělat následující tři kroky: Posunout (*Move*) objekt ve scéně, zvětšit nebo zmenšit (*Scale*) objekt, aby byl v souladu s ostatními objekty a posledním krokem je rotace (*Rotate*) pro správnou orientaci. Tyto tři operace jsou součástí každého grafického systému. Rotace a zvětšování nebo zmenšování jsou **lineárními transformacemi** souřadnic k objektovým bodům. Vezměme transformaci:

$$T : \mathbf{R}^2 \rightarrow \mathbf{R}^2,$$

pro kterou platí $T(\mathbf{v} + \alpha\mathbf{w}) = T(\mathbf{v}) + \alpha T(\mathbf{w})$ pro libovolné vektory \mathbf{v} a \mathbf{w} v \mathbf{R}^2 a libovolné reálné číslo α . Definice linearity zaručuje, že při jakékoliv lineární transformaci T , máme $T(\mathbf{0}) = \mathbf{0}$. Pokud vybereme $\mathbf{v} = \mathbf{w}$ a $\alpha = 1$, definice nám říká, že:

$$T(\mathbf{0}) = T(\mathbf{0} + \mathbf{10}) = T(\mathbf{0}) + 1T(\mathbf{0}) = T(\mathbf{0}) + T(\mathbf{0})$$

Odečtením $T(\mathbf{0})$ od první a poslední části dostaneme výsledek $\mathbf{0} = T(\mathbf{0})$. To znamená, že posun (*Translation*) každého bodu v rovině o stejnou vzdálenost není lineární transformací kromě jediného případu, kdy posuneme objekt o nulovou vzdálenost (objekt zůstane zachován na původním místě). Každá matice \mathbf{M} , která je 2. řádu funkce $\mathbf{v} \mapsto \mathbf{M}\mathbf{v}$ je lineární transformací $\mathbf{R}^2 \rightarrow \mathbf{R}^2$. Toto se nazývá maticová transformace. V praxi si to můžeme ukázat na příkladech jako je rotace objektu, zvětšování a deformace. V příkladech budeme nazývat matice \mathbf{M}_1 až \mathbf{M}_3 a lineární transformace \mathbf{T}_1 až \mathbf{T}_3 .

Příklad 1: Rotace

$$\text{Nechť je } \mathbf{M}_1 = \begin{bmatrix} \sin 30^\circ & -\sin 30^\circ \\ \sin 30^\circ & \cos 30^\circ \end{bmatrix} \text{ a}$$

$$T_1 : \mathbf{R}^2 \rightarrow \mathbf{R}^2 : \begin{bmatrix} x \\ y \end{bmatrix} \mapsto \mathbf{M}_1 \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos 30^\circ & -\sin 30^\circ \\ \sin 30^\circ & \cos 30^\circ \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \quad (5.1)$$

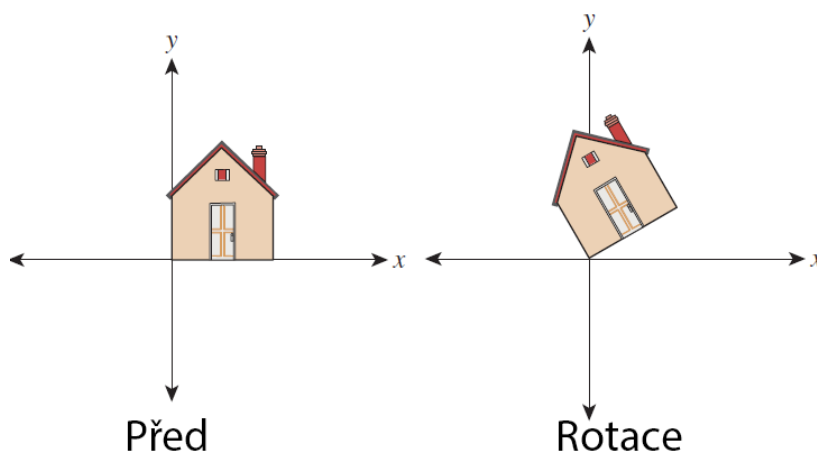
Pokud budeme mít vektory \mathbf{u}_1 a \mathbf{u}_2 , kdy

$$\mathbf{u}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \text{ a } \mathbf{u}_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

Tato transformace reprezentuje pro vektor

$$\mathbf{e}_1 \text{ hodnoty } \begin{bmatrix} \cos 30^\circ \\ \sin 30^\circ \end{bmatrix} \mathbf{e}_2 = \begin{bmatrix} -\sin 30^\circ \\ \cos 30^\circ \end{bmatrix}.$$

Pokud se podíváme na vztah (5.1) lze vidět, že se jedná o rotaci o 30° proti směru hodinových ručiček podle osy x a y . Když vyměníme úhel z příkladu jakýmkoliv jiným můžeme vytvořit transformaci, která rotuje proti směru hodinových ručiček o tento zvolený úhel [10]. Rotaci demonstruje obrázek 5.1.



Obr. 5.1: Transformace rotace

Příklad 2: Změna měřítka

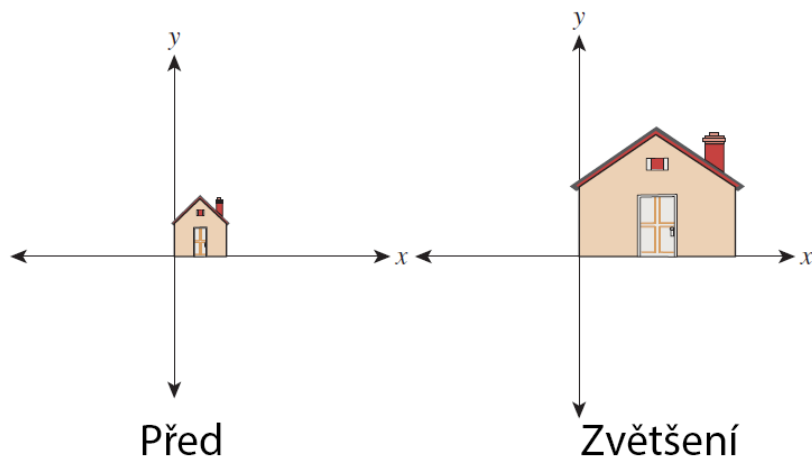
$$\text{Nechť je } \mathbf{M}_2 = \begin{bmatrix} 3 & 0 \\ 0 & 2 \end{bmatrix} \text{ a}$$

$$T_2 : \mathbf{R}^2 \rightarrow \mathbf{R}^2 : \begin{bmatrix} x \\ y \end{bmatrix} \mapsto \mathbf{M}_2 \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 3 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 3x \\ 2y \end{bmatrix} \quad (5.2)$$

Ze vztahu (5.2), můžeme říci, že tato transformace vše třikrát prodlužuje ve směru osy x a dvakrát ve směru osy y , jak lze vidět na obrázku 5.2. Pokud by byly oba faktory prodloužení rovno třem lze říci, že se jedná o uniformní transformaci. V případě, že budou na úhlopříčce matice je možné libovolně škálovat každou osu dle potřeby.

Příklad 3: Deformační transformace Nechť je

$$T_3 : \mathbf{R}^2 \rightarrow \mathbf{R}^2 : \begin{bmatrix} x \\ y \end{bmatrix} \mapsto \begin{bmatrix} 1 & -1 \\ 2 & -2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x - y \\ 2x - 2y \end{bmatrix} \quad (5.3)$$

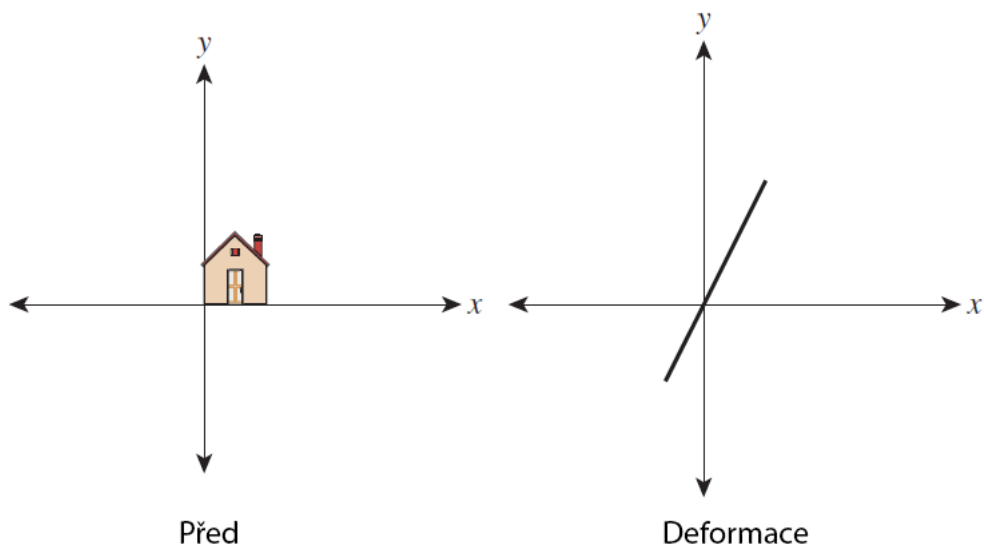


Obr. 5.2: Transformace zvětšení

Na obrázku 5.3 je vidět, proč se tato transformace nazývá deformační. Narozdíl od ostatních transformací deformuje celou 2D rovinu na jednodimenzionální podprostor, přímku. Tato transformace se také nazývá jako singulární, stejně jako matice, která jí definuje. Determinant matice

$$M_3 = \begin{bmatrix} 1 & -1 \\ 2 & -2 \end{bmatrix}$$

je nulový nebo říká, že sloupce jsou lineárně závislé.



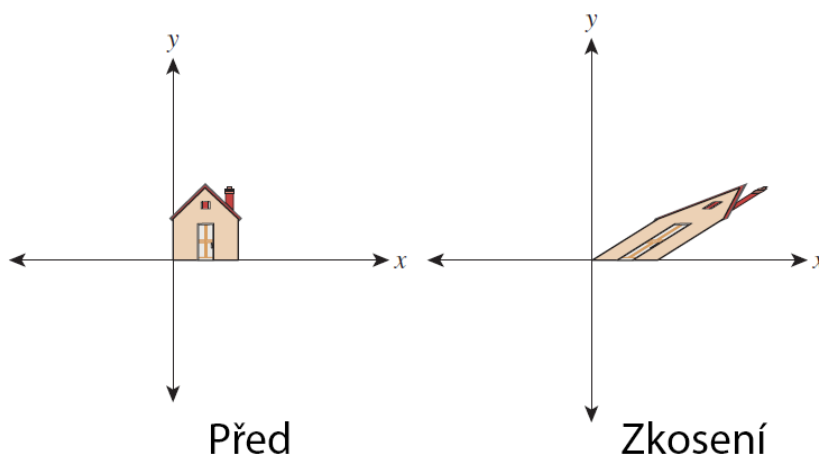
Obr. 5.3: Transformace deformace

Příklad 4: Zkosení Necht' je

$$\mathbf{M}_4 = \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix} a$$

$$\mathbf{T}_4 : \mathbf{R}^2 \rightarrow \mathbf{R}^2 : \begin{bmatrix} x \\ y \end{bmatrix} \mapsto \mathbf{M}_4 \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x + 2y \\ y \end{bmatrix}. \quad (5.4)$$

Na obrázku 5.4 je vidět zkosení podél osy x . Velikost původního objektu podél osy y je zachována.



Obr. 5.4: Transformace zkosení

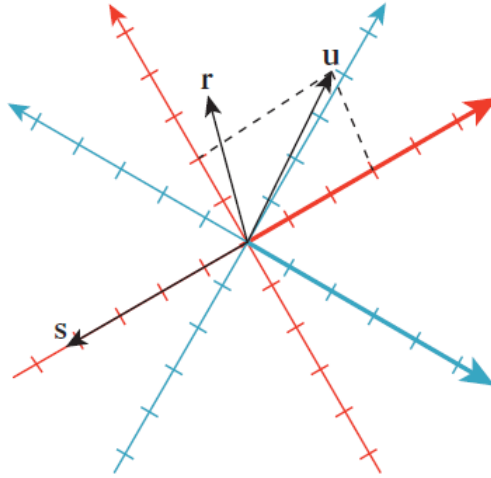
5.2 Transformace a souřadnicové systémy

Většinou přemýšlíme o lineárních transformacích jako o pohybujících se bodech, které ale mají fixní počátek a často je takto budeme používat. Stejně důležité je jejich použití v měnících se souřadnicových systémech. Pokud máme dva souřadnicové systémy v \mathbf{R}^2 se stejným počátkem jako je na obrázku potom každá šipka na obrázku má souřadnice jak v červeném tak v modrém souřadnicovém systému.

Červené i modré souřadnice mohou být napsány jako vektor. Například vektor \mathbf{u} má souřadnice $[3, 2]$ v červeném systému a přibližně $[-0, 2; 3, 6]$ v modrém systému.

Konverze z červeného do modrého souřadnicového systému se nazývá změna báze a můžeme jí vyjádřit pomocí maticových transformací [10].

$$\mathbf{M} = \frac{1}{2} \begin{bmatrix} 1 & -\sqrt{3} \\ \sqrt{3} & 1 \end{bmatrix}$$



Obr. 5.5: Dva rozdílné souřadnicové systémy v \mathbf{R}^2 , vektor \mathbf{u} má v červeném systému souřadnice $[3, 2]$, kdežto souřadnice v modrém systému jsou přibližně $[-0, 2; 3, 6]$ [10].

Násobením \mathbf{M} souřadnicemi vektoru \mathbf{u} z červeného systému dostaneme:

$$\mathbf{v} = \mathbf{M}\mathbf{u} \quad (5.5)$$

$$= \frac{1}{2} \begin{bmatrix} 1 & -\sqrt{3} \\ \sqrt{3} & 1 \end{bmatrix} \begin{bmatrix} 3 \\ 2 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 3 - 2\sqrt{3} \\ 3\sqrt{3} + 2 \end{bmatrix} \approx \begin{bmatrix} -0,2 \\ 3,6 \end{bmatrix}$$

Výsledek odpovídá souřadnicím v modrém souřadnicovém systému.

5.3 Homogenní souřadnice

Homogenní souřadnice se používají pro zjednodušení výpočtu transformací. Umožňuje nám to vyjádření lineárních transformací pomocí jedné matice, což v kartézských souřadnicích nelze. Popis transformací pomocí matic je velmi výhodný, protože každý grafický procesor má implementovanou knihovnu pro práci s maticemi. V kartézských souřadnicích je bod $\mathbf{P} = [x_{kart}, y_{kart}, z_{kart}]$. V homogenních souřadnicích je bod \mathbf{P} tvořen uspořádanou čtveřicí čísel $\mathbf{P} = [x_{hom}, y_{hom}, z_{hom}, w]$, kde w se nazývá váhou bodu. Vzájemný vztah je dán převodem:

$$x_{kart} = \frac{x_{hom}}{w}, y_{kart} = \frac{y_{hom}}{w}, z_{kart} = \frac{z_{hom}}{w}. \quad (5.6)$$

Váha nabývá intervalu $w \in (0, \infty)$, často se volí $w = 1$.

5.4 3D transformace

Transformace ve 3D prostoru jsou velmi podobné transformacím v rovinném prostoru. Využívá se obecné matice \mathbf{M} rozměru 4×4 reprezentující transformaci bodu \mathbf{B} na bod \mathbf{B}' .

$$\mathbf{B}' = \begin{bmatrix} x' \\ y' \\ z' \\ w' \end{bmatrix} = \mathbf{M} \cdot \mathbf{B} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix}. \quad (5.7)$$

Příklad 1: Posunutí Je definováno vektorem posunutí \mathbf{t} . Transformační a inverzní matice jsou:[23]

$$\mathbf{T}_1 = \begin{bmatrix} 1 & 0 & 0 & X_{t1} \\ 0 & 1 & 0 & Y_{t1} \\ 0 & 0 & 1 & Z_{t1} \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{T}_1^{-1} = \mathbf{T}_1(-X_{t1}, -Y_{t1}, -Z_{t1}) = \begin{bmatrix} 1 & 0 & 0 & -X_{t1} \\ 0 & 1 & 0 & -Y_{t1} \\ 0 & 0 & 1 & -Z_{t1} \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Příklad 2: Rotace Rotace v 3D prostoru je složitější oproti rovinné rotaci, protože se může jednat o rotaci kolem jednotlivých souřadnicových os. Rotace okolo osy x je dána maticemi:

$$\mathbf{T}_x(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta & 0 \\ 0 & \sin \theta & \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{T}_x^{-1}(\theta) = \mathbf{T}_x(-\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \theta & \sin \theta & 0 \\ 0 & -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Podobně jsou sestaveny matice pro osy y a z :

$$\mathbf{T}_y(\sigma) = \begin{bmatrix} \cos \sigma & 0 & \sin \sigma & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \sigma & 0 & \cos \sigma & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{T}_z(\omega) = \begin{bmatrix} \cos \omega & -\sin \omega & 0 & 0 \\ \sin \omega & \cos \omega & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Příklad 3: Změna měřítka Změnu měřítka v 3D prostoru popisuje matice:

$$\mathbf{T}_3(t_x, t_y, t_z) = \begin{bmatrix} t_x & 0 & 0 & 0 \\ 0 & t_y & 0 & 0 \\ 0 & 0 & t_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{T}_3^{-1} = \mathbf{T}_3(1/t_x, 1/t_y, 1/t_z),$$

kde koeficienty $t_x, t_y, t_z \neq 0$ popisují změnu ve směru příslušné osy. Pomocí jednotlivých koeficientů lze realizovat transformace souměrnosti [23].

5.5 Vkládání objektů do scény

Základní nastavení grafického zobrazení okna 3D view jsme si již popsali a nyní se budeme věnovat vkládání objektů do scény a jejich elementární práce s nimi. Zde si ještě popíšeme rozdíl mezi *Lokálním* a *Globálním* souřadnicovým systémem. Program Blender využívá zcela obecné jednotky, tudíž nikde není specifikováno jestli 1 jednotka označuje 1 metr nebo jinou hodnotu. Velmi vhodné je zachování poměru, který se dá nastavit tlačítkem *Spacing*, kdy jeden čtverec mřížky odpovídá např. jednomu metru.

Globální souřadnicový systém používá vztah k celé modelované scéně tzn. měřítko bude zachované ve stejném poměru jako je nastaveno v položce *Spacing* pro celou scénu.

Lokální souřadnicový systém se vztahuje k poloze středového bodu. Je možné si jednotlivé osy vůči globálním poupravit natočením např. o 30 stupňů, případně převrátit apod. Zde je potřeba si uvědomit rozdíl mezi lokálním a globálním souřadnicovým systémem [12].

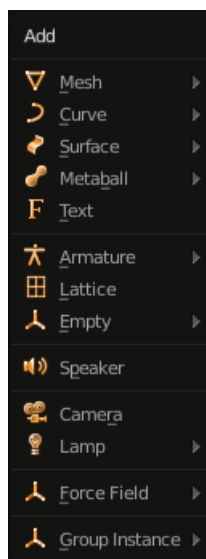
Vkládání objektů do scény je možné několika způsoby. První možností je kliknout na tlačítko *Add* (můžeme vidět na obrázku 5.6) a rozbalí se nabídka objektů, které lze přidat. Nový objekt se vždy vloží do místa, kde je umístěný 3D kurzor.



Obr. 5.6: Tlačítko pro přidání objektu do scény

Druhým možným způsobem, jak jednodušeji a rychleji vložit objekt do scény, je použití klávesové zkratky **Shift+A**. Typy objektů, které lze vložit do scény jsou vidět na obrázku 5.7.

- *Mesh* jedná se o objekty, se kterými se setkáme ve scéně nejčastěji. Jde o základní tvary jako jsou krychle, koule, čtverec, válec a další. Tyto objekty jsou složeny z bodů, hran a ploch, které lze libovolně upravovat
- *Curve* jsou křivky pomocí nichž je možné vytvářet objekty s oblými tvary
- *Surface* Plochy
- *Metaball* objekty pro tvorbu organických modelů a vazbu mezi nimi
- *Text* umožňuje vkládat libovolný prostý text do 3D prostředí
- *Armature* kosti, které se používají pro některé typy animací např. pro chůzi člověka
- *Lattice* je typ šablony pomocí které je možné tvarovat složitější tvary. Přiřazuje se objektům
- *Empty* jde o prázdný objekt s nulovou velikostí – není vidět. Používá se pro mapování textur například pro zrcadlové odrazy



Obr. 5.7: Nabídka pro vložení nových objektů

- *Speaker* umožňuje nastavit do scény zvuk
- *Camera* přidává do scény další kameru. Rendering probíhá z pohledu pouze jedné kamery, ale lze se mezi nimi přepínat
- *Lamp* vkládá do scény zdroj světla. Toto se využívá velmi často, protože většina scén používá více světelných zdrojů.
- *Force Field* nabízí možnost přidání dalšího pohybu pro dynamické systémy
- *Group Instance* vkládá skupiny objektů. Objekty lze dávat do skupin a pro jednotlivé skupiny nastavovat různé vlastnosti.[12, 9]

5.6 Objektový a editační mód

Většinu ze základních objektů po vložení potřebujeme upravit. Úpravy jsou možné ve dvou režimech – objektovém a editačním. Objekty typu *Camera*, *Speaker*, *Lamp*, *Group* a *Empty* lze upravovat pouze v objektovém režimu, protože editační režim pro ně nemá smysl. Pokud chceme přepínat mezi objektovým a editačním režimem, je nutné mít označený objekt, který chceme editovat. Označený objekt poznáme podle toho, zda má okolo sebe barevný obrys. Přepínání mezi objektovým a editačním módem je možné pomocí klávesy **Tab** nebo můžeme vybrat z rozbalovací nabídky volbu *Edit Mode*. Standardně je zapnutý *Object Mode*. V objektovém módu pracujeme s objekty jako celky a v editačním módu je možné pracovat pouze s jeho částmi jako jsou např. hrany, plochy a jiné. Každý objekt má ve svém těžišti bod, který se umístí po vložení na místo 3D kurzoru. Tento bod v objektovém módu v případě přesunu objektu se přesouvá společně, ale pokud se nacházíme v editačním

módu a přesuneme objekt, tak bod zůstane na původním místě. Na toto si musíme dávat pozor, protože některé operace mohou probíhat vůči tomuto bodu.

5.6.1 Základní editační příkazy objektového módu

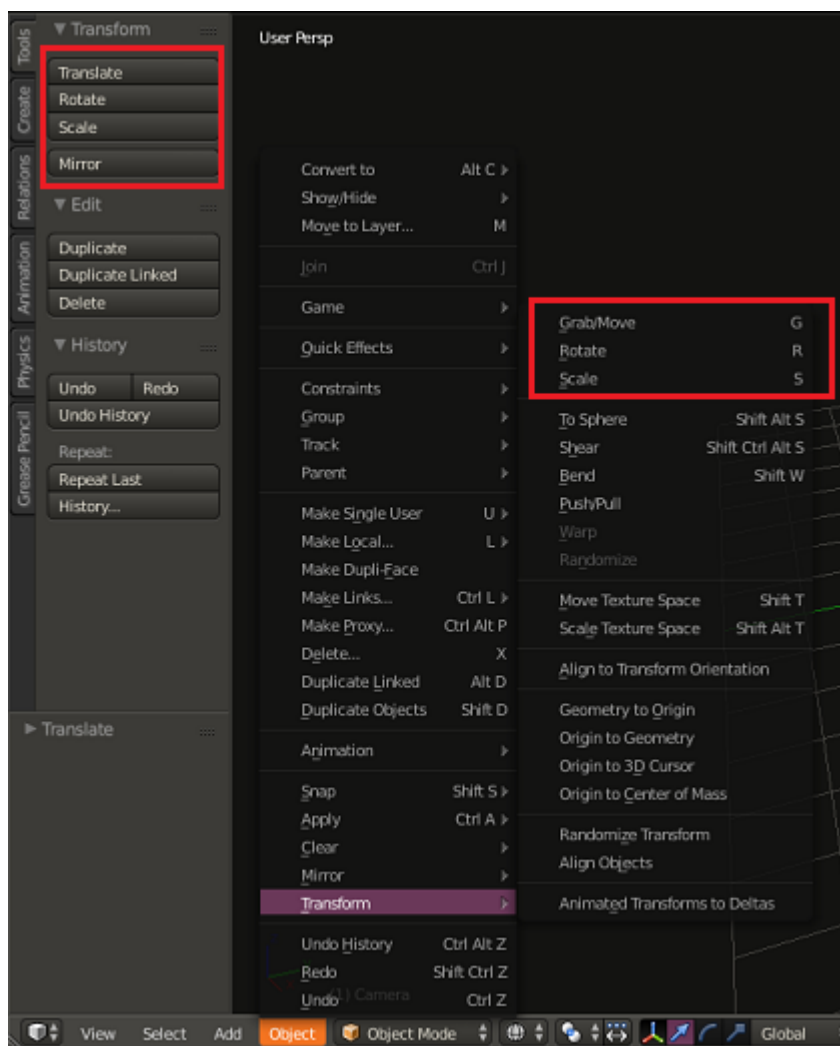
Výběry objektů Jak jsme již zmínili pro výběr objektu stačí kliknout pravým tlačítkem myši a objekt se barevně označí. Pokud chceme označit více objektů, je to možné pomocí klávesy **Shift**, kterou držíme stisknutou a vybíráme objekty. V případě, že máme ve scéně desítky objektů, můžeme kliknout na menu *Select* a vybrat jednu z následujících možností. Pokud chceme vybrat všechny objekty na scéně, stačí stisknout klávesu **A** a vyberou se všechny objekty a stejně tak je odznačíme. Po stisknutí klávesy **B** se objeví kříž a tažením myši utvoříme obdélník a vše co bude uvnitř se označí. S tímto výběrem lze použít kombinaci s funkcí *Inverse*, kdy všechno co nebylo označené se označí a opačně. Další metodou pro výběr je možnost *Select All by Type*, kdy se vyberou všechny objekty z dané skupiny, tzn. pokud vybereme objekty typu *Mesh*, označí se všechny objekty tohoto typu (krychle, koule, válec atd.).

Skrytí a mazání objektů Pokud potřebujeme nějaký objekt skrýt pro lepší přehlednost scény, je to možné pomocí příkazu *Hide*, kdy máme na výběr dvě varianty. První z nich je *Hide Selected* (klávesová zkratka **H**), kdy se skryjí aktuálně vybrané objekty a druhým je *Hide Deselected* (klávesová zkratka **Shift+H**), který skryje všechny nevybrané objekty. Mazání objektů lze klávesou **X** nebo **Delete**. Po zadání příkazu se objeví nabídka s potvrzením. V editačním módu se objekt nesmaže, ani pokud vybereme celý objekt. Zůstane nám středový bod, dle kterého to lze poznat.

Kopírování objektů Pro kopírování máme k dispozici dvě možnosti. Příkaz *Duplicate* (klávesová zkratka **Shift+D**) vytvoří nezávislou kopii vybraného objektu nebo vybrané skupiny, kdežto příkaz *Duplicate Linked* (klávesová zkratka (**Alt+D**)) vytvoří kopii, která se chová samostatně v objektovém módu, ale v editačním má dedičný s originálem, ze kterého byla kopírována. V praxi to znamená, že pokud provedeme nějakou operaci s kopií, projeví se to i na originále objektu. Tohoto se využívá např. pro modelaci symetrického předmětu, kde vymodelujeme jen jednu polovinu a druhou zkopírujeme. Předměty lze taktéž slučovat pomocí příkazu *Join Objects*, kdy můžeme několik objektů sloučit do jednoho. Zde musí platit jedna podmínka, že objekty musí být stejného typu např. *Mesh*.

5.7 Příkazy transformací

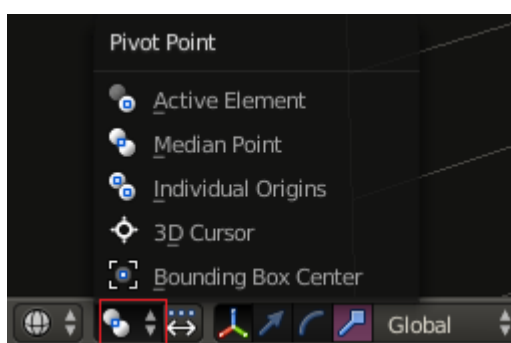
Transformace jsou jedny z nejvyužívanějších operací při práci s objekty. Mezi transformace patří posunutí, otočení, změna měřítka a zrcadlení. Tyto příkazy lze nalézt v okně *Object Tools*, které najdeme na levé straně 3D okna. Toto menu lze vyvolat a skrýt pomocí klávesy **T**, případně změny najdeme v menu *Object – Transform*. Obě menu jsou na obrázku 5.8. Všechny transformace mohou využívat globálního nebo lokálního souřadnicového systému.



Obr. 5.8: Menu pro příkazy transformací

Změna polohy je označena v okně *Object Tools* jako *Translate* nebo v menu *Object – Transform – Grab/Move* (klávesová zkratka je **G**). Volný pohyb lze pomocí myši, kdy stačí kliknout pravým tlačítkem myši a pohybovat se. Obdobně lze nalézt změnu měřítka *Scale* (klávesová zkratka **S**) a rotace *Rotate* (klávesová zkratka **R**). V případě, že chceme pohybovat s objektem ve skocích

po jednotkách nebo po pětistupňových intervalech pro rotaci, lze to pomocí stisknutí tlačítka **CTRL** a zvolený příkaz. Pokud chceme pohyb nebo rotaci podél vybrané osy stiskneme klávesu pro danou operaci a následně písmeno označující osu např. pohyb podél osy x bude vypadat **G+x**. Tento příkaz lze doplnit o hodnotu, jakou má být posunut případně natočen. Pak by měl příkaz tvar **G+x+5** a objekt se posune podle osy x o 5 jednotek. Takto zadané příkazy ukončíme klávesou **Enter**. Transformace lze vrátit do jejich původního stavu klávesovou kombinací **Alt+příslušná klávesa** G pro pohyb, S pro změnu měřítka a R pro rotaci. Všechny transformace jsou vztaženy k jejich bodům ležícím v těžisti objektu. Toto je možno změnit v menu *Pivot Point*, zobrazeno na obrázku 5.9.



Obr. 5.9: Možnosti vztažného bodu pro transformace

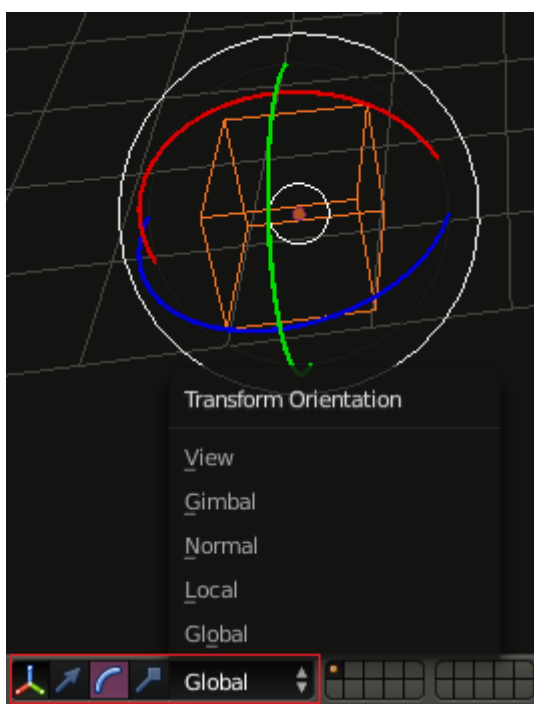
- *Active Element* operace se budou provádět vzhledem ke středovému bodu naposledy vybraného objektu
- *Median Point* transformace vzhledem ke středovému bodu objektu
- *Individual Origins* označuje, že se transformace budou vztahovat u každého objektu k vlastnímu středovému bodu
- *3D Cursor* transformace probíhá vzhledem k 3D kurzoru
- *Bounding Box Center* označuje střed pomyslného kváдру, který obepíná všechny vybrané objekty

Pokud chceme zadávat operace transformací přesně, je možné numerické zadávání v menu, které obsahuje informace o objektu. Menu vyvoláme klávesovou zkratkou **N**. V tomto okně lze editovat velmi přesně všechny transformace jako je pozice, rotace, měřítko a další.

5.7.1 Manipulátory

Manipulátory slouží ke zjednodušení operací s objektem. Zobrazují se kolem středového bodu vybraného objektu a všechny potřebné předvolby najdeme ve spodní části okna *3D view*. Manipulátory jsou zobrazené na obrázku 5.10. Po aktivaci

se objeví okolo objektu kružnice reprezentující jednotlivé osy nebo osový kříž pro manipulátory posunu a zvětšení. Ikona osového kříže povoluje použití manipulátorů



Obr. 5.10: Manipulátory

a následující tři ikony jsou pro jednotlivé typy transformací. Lze vybrat více manipulátorů najednou pomocí stisknuté klávesy **Shift**. Orientaci manipulátorů můžeme vybrat z roletového menu *Transform Orientation*.

6 PROMÍTÁNÍ

V počítačové grafice jsou všechny trojrozměrné objekty zobrazované na zařízeních využívající pouze dvojrozměrné zobrazení. Těto transformaci se říká promítání. Dochází ke ztrátě prostorové informace a díky tomu může pozorovatel vidět zkreslený tvar objektu. Abychom mohli z dvojrozměrného obrázku odvodit prostorové informace jako jsou úhel natočení nebo vzdálenost, je potřeba znát deskriptivní geometrii. Pro potřeby počítačové grafiky se používají prakticky orientované promítací metody. Uvedeme si zde dva příklady rovinného promítání, které je v počítačové grafice hojně využíváno. Důležitým pojmem je průmětna. Jde nejčastěji o rovinu, na kterou promítáme virtuální paprsky všech bodů ve scéně. Trojrozměrné úsečky se zobrazují jako dvojrozměrné. Definujeme dva souřadnicové systémy:

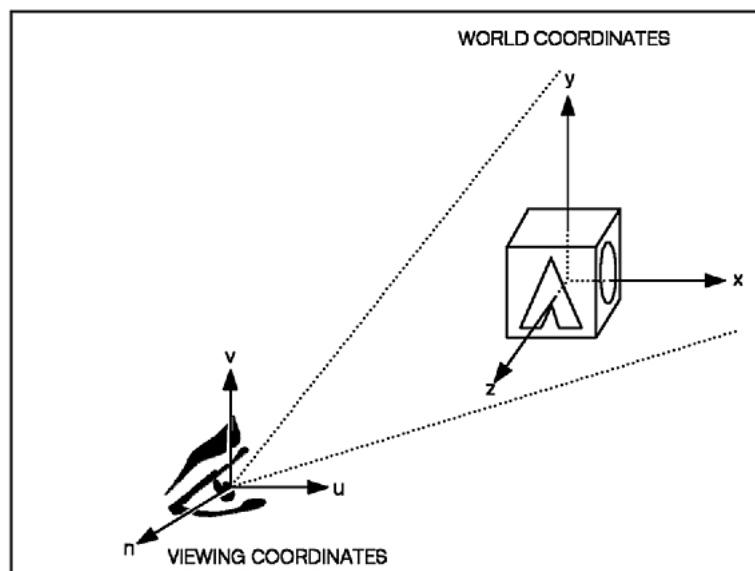
- WCS (*World Coordinate systém, světové souřadnice*) tyto souřadnice jsou neměnné.
- VCS (*Viewing Coordinate System, souřadnice průmětny*) jsou závislé na pohybu kamery.

Na obrázku 6.1 je zobrazen rozdíl mezi souřadnicovými systémy. Matice převádějí každý bod podle vzorce:

$$\mathbf{P}' = \mathbf{T}_{proj}\mathbf{P}, \text{ kde}$$

$$\mathbf{T}_{proj} = \mathbf{T}_{persp} \cdot \mathbf{T}_{cam} \cdot \mathbf{T}_{world} \quad (6.1)$$

Jako průmětna se často volí rovina xy nebo rovina s ní rovnoběžná.



Obr. 6.1: Rozdíl mezi souřadnicovými systémy WCS a VCS. Na obrázku chybí průmětna [21].

6.1 Kamera

Jako první musíme určit polohu pozorovatele, pozici a orientaci průmětny a stanovit cíl a směr pozorování. Kamery pořizují snímky na průmětnu, která je kolmá na hlavní optickou osu kamery [23]. Kamera bude umístěna v určitém bodě se světovými souřadnicemi WCS $[x_{cam}, y_{cam}, z_{cam}, 1]$. Ve scéně zvolíme bod $\mathbf{B} = [B_x, B_y, B_z, 1]$, na který kameru namíříme a dostaneme směr pozorování $\vec{L} = [L_x, L_y, L_z, 0]$, který je rovnoběžný s hlavní optickou osou kamery.

$$\begin{bmatrix} L_x \\ L_y \\ L_z \\ 0 \end{bmatrix} = \begin{bmatrix} B_x \\ B_y \\ B_z \\ 1 \end{bmatrix} - \begin{bmatrix} x_{cam} \\ y_{cam} \\ z_{cam} \\ 1 \end{bmatrix}.$$

Po normalizaci \vec{L} dostaneme jednotkový vektor \vec{l} . Požadujeme, aby transformace mezi WCS a VCS převedla jednotkový vektor \vec{l} na $[0, 0, -1, 0]$, který je kolmý na průmětnu a orientován ve směru záporné poloosy.

$$\begin{bmatrix} 0 \\ 0 \\ -1 \\ 0 \end{bmatrix} = \mathbf{T}_{cam} \begin{bmatrix} l_x \\ l_y \\ l_z \\ 0 \end{bmatrix}.$$

Orientaci kamery určuje vektor \vec{u} , kde předpokládáme, že je kolmý na vektor \vec{l} a jednotkový. Vektorovým součinem určíme třetí vektor $\vec{p} = \vec{l} \times \vec{h}$. Vektory jsou vzájemně kolmé. Potřebujeme, aby pohledová transformace mezi WCS a VCS převedla vektor \vec{h} na $[1, 0, 0, 0]$ a vektor \vec{p} na $[0, 1, 0, 0]$.

$$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \mathbf{T}_{cam} \cdot \begin{bmatrix} p_x \\ p_y \\ p_z \\ 0 \end{bmatrix}, \quad \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} = \mathbf{T}_{cam} \cdot \begin{bmatrix} h_x \\ h_y \\ h_z \\ 0 \end{bmatrix}.$$

Po sestavení dostaneme na levé straně rovnice jednotkovou matici a hledaná matice \mathbf{T}_{cam} je inverzní maticí sestavené z vektorů $\vec{l}, \vec{h}, \vec{p}$.

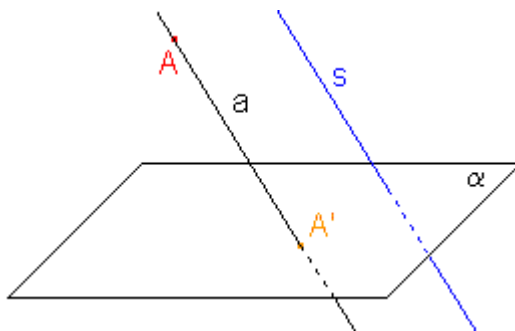
$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \mathbf{T}_{cam} \cdot \begin{bmatrix} p_x & h_x & -l_x & 0 \\ p_y & h_y & -l_y & 0 \\ p_z & h_z & -l_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Pokud matici transponujeme dostáváme výslednou matici pro \mathbf{T}_{cam} .

$$\mathbf{T}_{cam} = \begin{bmatrix} p_x & h_x & -l_x & 0 \\ p_y & h_y & -l_y & 0 \\ p_z & h_z & -l_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} p_x & p_y & p_z & 0 \\ h_x & h_y & h_z & 0 \\ -l_x & -l_y & -l_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (6.2)$$

6.2 Rovnoběžné promítání

Máme rovinu α a přímku p , která bude definovat směr promítání a zároveň přímka p není rovnoběžná s rovinou α , jedná se o rovnoběžné promítání prostoru na rovinu α , kdy směrem s je zobrazení. Body se promítají do roviny α vzájemně rovnoběžnými přímkami ve směru s [11]. Na obrázku 6.2 je vidět příklad rovnoběžného zobrazení, kdy obrazem bodu je bod.



Obr. 6.2: Vlastnosti rovnoběžného promítání [11]

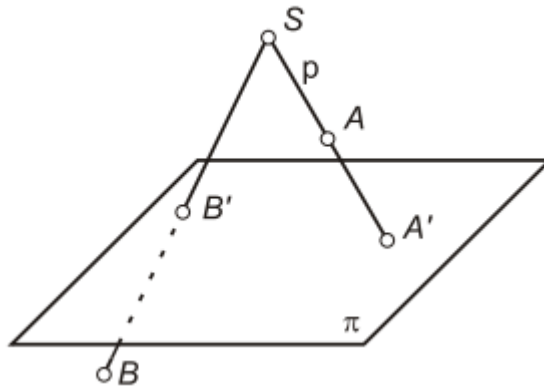
Nejjednodušším typem je promítání do některé z rovin $x = x_0$, $y = y_0$, $z = z_0$, ve směru příslušné osy kolmé na zvolenou průmětnu. V případě, že je průmětnou některá z hlavních rovin bude x_0 , y_0 nebo $z_0 = 0$. V příkladu si uvedeme popis promítání v homogenních souřadnicích včetně rovnoběžně snímající kamery [23]. Jelikož jde o případ, kdy promítací paprsky jsou rovnoběžné, je poslední řádek v matici roven $[0, 0, 0, 1]$.

$$\mathbf{T}_{xy} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{T}_{xz} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & y_0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{T}_{yz} = \begin{bmatrix} 0 & 0 & 0 & x_0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (6.3)$$

6.3 Středové promítání

Jedná se o optický model, který vyjadřuje lidské vidění reálného světa. Se zvětšující vzdáleností se pozorované objekty proporciálně zmenšují. Objekty, které se nacházejí

za průmětnou, jsou při zobrazení zmenšeny, objekty v průmětně zachovávají svoji velikost a objekty, které se nacházejí před průmětnou, jsou zvětšeny. Středové promítání je určeno středem S , tedy bodem, ze kterého všechny paprsky vycházejí. Zvolme rovinu π , která bude průmětnou, střed promítání leží mimo rovinu π . Libovolný bod A zobrazíme tak, že bod S a A proložíme přímkou p a dostaneme obraz bodu, který nazveme A' . Jde o průsečík přímky p a roviny π . Obdobně můžeme sestrojít bod B' jako středový průmět bodu B . Celou situaci můžeme vidět na obrázku 6.3 [20].



Obr. 6.3: Vlastnosti středového promítání [20]

Vezměme bod $\mathbf{P} = [x, y, z, 1]$, který bude transformován na bod $\mathbf{P}_{persp} = [x_p, y_p, z_p, w_p]$ pomocí matice

$$\begin{bmatrix} x_p \\ y_p \\ z_p \\ w_p \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{1}{z_0} & 0 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \mathbf{T}_{persp} \cdot \mathbf{P}. \quad (6.4)$$

Perspektivní promítání zobrazíme v homogenních souřadnicích pomocí lineární transformace.[23]

7 OSVĚTLOVACÍ METODY

7.1 Světlo

Světlo je pro nás médiem pro vizuální vnímání světa všude okolo nás. V počítačové grafice hraje světlo stejně důležitou roli, jako je tomu v reálném životě. O odraz světla od povrchu, modely stínování povrchů a simulace světelných zdrojů se stará zobrazovací rovnice, jeden z nejdůležitějších pojmů v počítačové grafice. Pracuje se výhradně s geometrickou optikou, kdy modelujeme světlo jako nezávislé paprsky a jejich trajektorie můžeme popsat geometrickými pravidly. Budeme vycházet z následujících předpokladů:

- Světlo se šíří přímočaře.
- Rychlost šíření světla je nekonečná. Světlo po opuštění zdroje je přítomno na všech místech a je tak v ustáleném stavu. Odezvy jsou okamžité.
- Světlo není ovlivněno gravitací a neuplatňují se relativistické jevy.

7.1.1 Radiometrické pojmy

Na mikroskopické úrovni je základní světelná částice foton. Je to nejmenší kvantum záření, které může být vyzářeno. Energie fotonu na vlnové délce je:

$$e_\lambda = hf = \frac{h \cdot c}{\lambda} [J], \quad (7.1)$$

kde $h \approx 6,63 \cdot 10^{-34} J \cdot s$ je Planckova konstanta, $c \approx 2,996 \cdot 10^8 m \cdot s^{-1}$ je rychlost světla ve vakuu.

Energie fotonů určité vlnové délky v určité oblasti se nazývá radiantní energie Q . Jde o nesměrovou veličinu a jedná se o součet energií v určitém místě. Radiantní energie Q se určí jako integrál

$$Q = \int_0^\infty n_\lambda e_\lambda d\lambda [J], \quad (7.2)$$

kde n_λ je počet fotonů vlnové délky a e_λ je energie fotonů na vlnové délce. Veškeré veličiny se mohou uvádět pro konkrétní vlnovou délku nebo mohou být nezávislé na vlnové délce. Pokud jsou nezávislé na vlnové délce, získá se jako integrál všech příspěvků závislých na vlnové délce skrze celé viditelné spektrum. Dopadající světlo na povrch objektu se nazývá zářivý tok. Jde o množství energie vyzářené nebo přijaté za jednotku času.

$$\Phi = \frac{dQ}{dt} [W], \quad (7.3)$$

Světelný tok dopadající na jednotku plochy se nazývá irradiance a označuje se E .

$$E = \frac{d\Phi_i}{dA} [Wm^{-2}], \quad (7.4)$$

kde A je ozářená plocha. Světelný tok vyzářený plochou (vlastní emise nebo odražený tok) se nazývá radiozita [23].

$$M = \frac{d\Phi_r}{dA} [Wm^{-2}]. \quad (7.5)$$

Zářivost označovaná též jako hustota výkonu v prostorovém úhlu je dána světelným tokem Φ proudící prostorovým úhlem $d\vec{\omega}$.

$$I = \frac{d\Phi}{d\vec{\omega}} [Wsr^{-1}m^{-2}]. \quad (7.6)$$

7.1.2 Radiance

Radiance je velmi důležitou veličinou pro globální osvětlování. Udává přijímaný nebo vyzářený výkon na jednotkovém prostorovém úhlu na jednotku kolmo promítnuté plochy. Radianci je možné si představit jako počet fotonů, které jsou přijaté nebo vyzářené v určitém směru za jednotku času a procházející průmětem diferenciální plošky. Radiance závisí na poloze bodu x a směru $\vec{\omega}$.

$$L(x, \vec{\omega}) = \frac{d^2\Phi}{\cos\Theta dAd\vec{\omega}} [Wsr^{-1}m^{-2}].$$

Jelikož velikost promítnuté plochy závisí na kosinu úhlu, který je svírá, kolmice k promítnuté ploše a kolmice k ploše promítnuté je ve jmenovateli zlomku $\cos\Theta$ [23]. Veškeré radiometrické veličiny, které jsme si uvedli lze vyjádřit z radiance. Radiantní energii (7.1) lze vypočítat jako integrál radiance přes čas T , polokouli Ω nad bodem x a přes všechny plochy $x \in A$.

$$Q = \int_T \int_{\Omega} \int_{x \in A} L(x, \vec{\omega}) \cos\Theta dAd\vec{\omega} dt.$$

Zářivý tok (7.3) lze vyjádřit jako integrál přes polokouli Ω a všechny body plochy A .

$$\Phi = \int_{\Omega} \int_{x \in A} L(x, \vec{\omega}) \cos\Theta dAd\vec{\omega}.$$

Přijímaná energie (7.4) (irradiance) je popsána integrálem vyzářené radiance přes polokouli.

$$E = \int_{\Omega} \int_{x \in A} L(x, \vec{\omega}) \cos\Theta d\vec{\omega}.$$

A nakonec intenzita (7.6) se získá integrací přes plochu A :

$$I = \int_{\Omega} \int_{x \in A} L(x, \vec{\omega}) \cos\Theta dA.$$

Ve všech algoritmech pro globální osvětlování se vychází z faktu, že ve vakuu je radianční na své dráze konstantní [17]. Výjimku tvoří částicové systémy jako jsou kouř, mlha, vodní pára atd. Fotoaparáty, lidské oko jsou citlivé na radianci, kde jejich odezva je přímo úměrná přijaté radianci a proto se barva nebo kontrast objektu nemění se vzdáleností pozorování. Radiance je interpretována obrazovkou počítače jako barva[23].

7.2 Dvousměrová odrazová distribuční funkce – BRDF

Většina světla, které vnímáme lidským okem je odražené světlo od různých povrchů objektů. Světlo, které dopadá přímo do lidského oka, je naprosté minimum. Barva objektů je dána spektrální charakteristikou světla a také materiálem povrchu. Každý povrch odráží jinou vlnovou délku. Také je důležité v jakém směru daný povrch světlo odráží. Pro popis jednotlivých schopností materiálu odrážet nebo absorbovat světlo potřebujeme definovat následující předpoklady:

1. Světlo, které se odrazí od povrchu se odrazí okamžitě.
2. Foton na vlnové délce λ bude vyzařen na stejné vlnové délce.
3. Přicházející světlo na povrch materiálu ze směru $\vec{\omega}_i$ v bodě x se odráží ve směru $\vec{\omega}_r$.

Pro výpočet odraženého paprsku použijeme vztah založený na skalárním součinu:

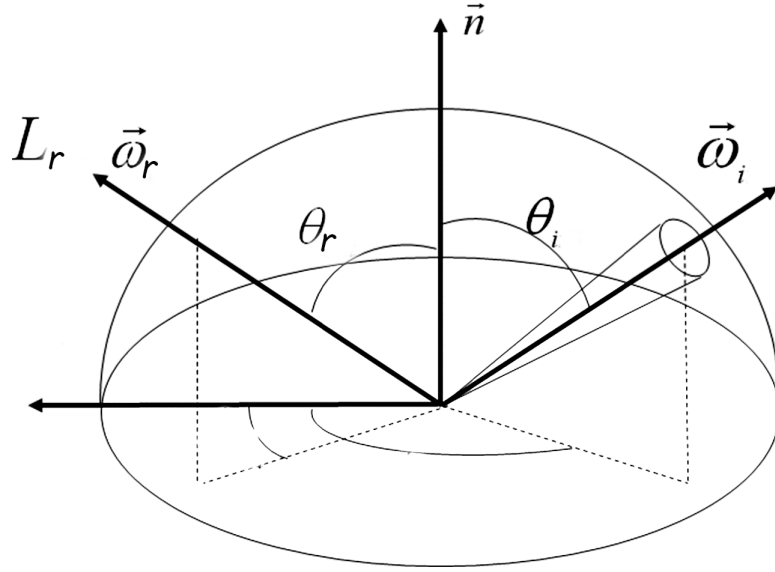
$$\vec{\omega}_r = 2(\vec{\omega}_i \cdot \vec{n}) \cdot \vec{n} - \vec{\omega}_i. \quad (7.7)$$

Dvousměrová odrazová distribuční funkce (Bidirectional Reflectance Distribution Function) popisuje odrazové schopnosti povrchu materiálu v určitém bodě x . Funkce je popsána na obrázku 7.1. Dopadající světlo označme $\vec{\omega}_i$ a odražené $\vec{\omega}_r$. BRDF se označuje $f_r(x, \vec{\omega}_r, \vec{\omega}_i)$ a definuje poměr odražené radianční $dL_r(x, \vec{\omega}_r)$ v bodě x ke vstupní diferenciální radianci $dL_i(x, \vec{\omega}_i)$ promítnuté na kolmou plochu. Největší světelný výkon je v případě, že radiance dopadá kolmo na plochu [23].

$$f_r(x, \vec{\omega}_r) = \frac{dL_r(x, \vec{\omega}_r)}{dL_i(x, \vec{\omega}_i \cdot \vec{n})d\vec{\omega}_i}. \quad (7.8)$$

Rovnice odrazu popisuje velikost celkové odražené radianční L_r z daného bodu do směru ω_r v závislosti na množství přicházejícího světla L_i a materiálu povrchu f_r . Rovnici lze odvodit z definice BRDF:

$$dL_r(\omega_0) = f_r(\omega_0 \rightarrow \omega_i) \cdot \cos \Theta_i \cdot d\omega_i. \quad (7.9)$$



Obr. 7.1: Dvousměrová odrazová distribuční funkce (BRDF)

Celková odražená radiance je definována integrálem příchozí radiance přes celou hemisféru H (přes všechny příchozí směry).

$$L_r(x, \omega_0) = \int_{H(x)} L_i(x, \omega_i) \cdot f_r(x, \omega_i \rightarrow \omega_0) \cdot \cos \Theta_i \cdot d\omega_i \quad (7.10)$$

7.2.1 Vlastnosti BDRF

- Pokud zaměníme směr odrazu a dopadu světelného paprsku, zůstane hodnota BDRF stejná dle Hemholtzova principu reciprocity.

$$f_r(x, \vec{\omega}_r, \vec{\omega}_i) = f_r(x, \vec{\omega}_i, \vec{\omega}_r)$$

Dopadající foton ze směru $\vec{\omega}_i$ je na povrchu materiálu několikanásobně odražen, kde vystupuje ve směru posledního odrazu $\vec{\omega}_r$. Pokud je směr opačný musí foton projít přesně stejnou cestou.

- Funkce BRDF není nikdy záporná $f_r(x, \vec{\omega}_r, \vec{\omega}_i) \geq 0$.
- Je anizotropní. To znamená, že v případě odrazu světla ve vstupním nebo výstupním směru v bodě x náleží také natočení povrchu kolem normálového vektoru. Funkce by potom měla mít tvar $f_r(x, \vec{\omega}_r, \phi, \vec{\omega}_i)$, kde ϕ je úhel natočení bodu. Avšak v počítačové grafice se anizotropie většinou neuvažuje.
- Zákon zachování energie říká, že energie se nemůže vytvořit ani zaniknout, může se pouze měnit na jiný druh [23].

$$\int_{\Omega} f_r(x, \vec{\omega}_r, \vec{\omega}_i) \cos \Theta_i d\vec{\omega}_i < 1, \forall \vec{\omega}_i.$$

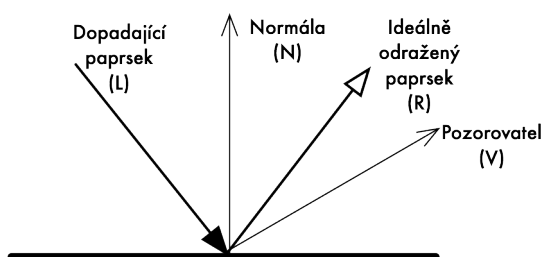
- Linearita popisuje, že vyzářený paprsek ve směru $\vec{\omega}_i$ je vyzářen ve výstupním směru bez ohledu na to, co přichází ze směrů jiných.
- Odrazivost definuje poměr odraženého světelného toku $\Phi_r(x)$ k dopadajícímu světelnému toku $\Phi_i(x)$ vztaženému k bodu x .

$$\rho(x) = \frac{d\Phi_r(x)}{d\Phi_i(x)}$$

Nabývá hodnot v intervalu $\langle 0, 1 \rangle$ a pokud $\rho(x) = 1$ je odraženo vškeré dopadající světlo.

7.3 Phongův osvětlovací model

Jedná se o empirický osvětlovací model pro výpočet odraženého světla z povrchu objektu. Tento model se používá již dlouhou dobu pro vykreslování v počítačové grafice. Vykazuje velmi dobré výsledky, ale nedokáže nasimulovat všechny optické jevy jako např. kaustiku. Odraz je určen dopadajícím paprskem \mathbf{L} směrem k pozorovateli, normálou \mathbf{N} v místě dopadu a zrcadlově odraženým paprskem \mathbf{R} , jak je naznačeno na obrázku 7.2. Výsledný odraz je celkově složen ze tří složek. Jedná se



Obr. 7.2: Phongův osvětlovací model [14]

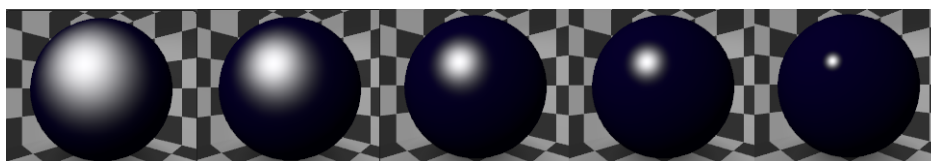
o spekulární (zrcadlový), difúzní a ambientní odraz.

Zrcadlová složka je vyjádřena vztahem

$$\mathbf{I}_s = \mathbf{I}_L \mathbf{r}_s (\vec{v} \cdot \vec{r})^h, \quad (7.11)$$

kde \mathbf{I}_L označuje barevné složení dopadajícího paprsku, \vec{v} je jednotkový vektor z pohledu pozorovatele z obrázku 7.2, \vec{r} vyjadřuje směr ideálního zrcadlového odrazu a je symetrický k dopadajícímu paprsku podle normály. Lze jej vypočítat podle vztahu $\vec{r} = 2(\vec{l} \cdot \vec{n})\vec{n} - \vec{l}$, kde skalární součin jednotkových vektorů je roven kosinu úhlu, který svírají. Koeficient zrcadlového odrazu je trojsložkovým vektorem a leží v intervalu $0 \leq \mathbf{r}_s \leq 1$ a určuje celkovou velikost odražené zrcadlové složky v odrazu.

Koeficient h je skalární veličinou a vyjadřuje ostrost zrcadlového odrazu a pohybuje se v intervalu $\langle 1, \infty \rangle$, a pokud je $\vec{v} \cdot \vec{r} < 0$ je pozorovatel v prostoru, kde je zdroj světla a odraz nemůže vidět. Zrcadlově odražené světlo nabývá maxima pokud je směr odrazu blízký směru pohledu. S roustoucím h se odrazy stávají menší a ostřejší. Dokonalé zrcadlo má $h = \infty$. Na obrázku 7.3 je vidět vliv koeficientu h na velikost odrazu.



Obr. 7.3: Vliv koeficientu h na velikost odlesku pro hodnoty 10, 20, 50, 100 a 1000

Difúzní složka se vypočítá podle vztahu

$$\mathbf{I}_d = \mathbf{I}_L \mathbf{r}_d (\vec{l} \cdot \vec{n}), \quad (7.12)$$

kde \mathbf{r}_d je koeficient difúzního odrazu, který udává velikost difúzní složky v celkově odraženém světle. Jde o trojsložkový vektor. Částečně také vyjadřuje naše barevné vnímání objektu. Množství světla \mathbf{I}_d je tím větší, čím je směr dopadu bližší normále. Lambertův zákon difúzního odrazu $\mathbf{I}_d = I \cdot \cos \alpha$ popisuje, že vztah má smysl pouze pro hodnoty $\vec{l} \cdot \vec{n} > 0$, protože v opačném případě je povrch odvrácen od světla a difúzní složka je nulová [23].

Ambientní složka je odrazem světla, které přichází ze všech směrů. Vzniká mnohonásobnými odrazy a ve většině případů je bílé. Je vyjádřeno vztahem

$$\mathbf{I}_a = \mathbf{I}_A \mathbf{r}_a, \quad (7.13)$$

kde \mathbf{I}_A vyjadřuje množství okolního světla, která je většinou konstantní pro celou scénu. Barevný koeficient \mathbf{r}_a je většinou totožný s vektorem \mathbf{r}_d , prakticky určuje, zda je objekt světlý nebo tmavý. Ambientní složka zabraňuje, aby povrchy objektů odvrácených od světelných zdrojů byly zcela černé. Sečtením všech výše uvedených složek získáme vztah pro světlo, které vnímá pozorovatel.

$$\mathbf{I}_V = \mathbf{I}_s + \mathbf{I}_d + \mathbf{I}_a. \quad (7.14)$$

Dnešní grafické procesory obsahují hardwarovou implementaci algoritmu osvětlení. Pro svou jednoduchost je Phongův osvětlovací model používaný v aplikacích v reálném čase (tzv. real-time).

7.4 Světelné zdroje

Správné nastavení světel ve scéně je velmi důležité, protože žádná sebelepší scéna nebude bez odpovídajícího nastavení světel působit dobře. Jako světelný zdroj můžeme považovat obecně jakýkoli objekt, protože může záření emitovat nebo odrážet. Dále si popíšeme a ukážeme praktické nastavení jednotlivých světelných zdrojů používaných v počítačové grafice. V Blenderu můžeme nové světlo vložit pomocí příkazu *Add – Lamp* a dále výběrem z nabídky. Světlo se vloží na pozici 3D kurzoru. S každým světelným zdrojem lze provádět běžné příkazy jako posun (klávesa **G**), otočení (klávesa **R**) nebo mázání (klávesa **Delete**). Otočení je pro některé světelné zdroje důležité, protože tím nastavujeme směr šíření paprsků. Se světelnými zdroji se pracuje pouze v objektovém módu. K náhledu světelných zdrojů slouží panel *Preview*, který se otevře, pokud vybereme světelný zdroj ve scéně, případně vybereme obrázek světla v okně předvoleb. V případě, že je ve scéně více světel je možné přepínat pomocí roletové nabídky. Předvolby pro světlo jsou vidět na obrázku 7.4. Barva se nastavuje po kliknutí do okna s barvou (výchozí hodnota je $RGB = [1, 1, 1]$). Intenzita světla se nastavuje pomocí posuvného tlačítka *Energy*. V případě zaškrtnutí tlačítka *Negative* se použije invertované osvětlení. To v praxi znamená, že je světlo odečítáno od ostatních zdrojů ve scéně. *This Layer Only* určuje, že osvětlení ovlivňuje pouze objekty ve scéně, které jsou ve stejné vrstvě jako tento světelný zdroj. Volba *Specular* zakazuje vliv na spekulární shader objekty a *Diffuse* určuje, zda světelný zdroj má vliv na difuzní materiál.

7.4.1 Bodový světelný zdroj

Bodový zdroj vyzařuje světlo rovnoměrně a do všech směrů se stejnou intenzitou. Jeho intenzita se snižuje v závislosti na vzdálenosti od tohoto zdroje. Nastavení na kartě *Lamp*, které je specifické pro tento typ osvětlení je numerické tlačítko *Distance*, které určuje vzdálenost od zdroje světla, kdy bude intenzita poloviční.

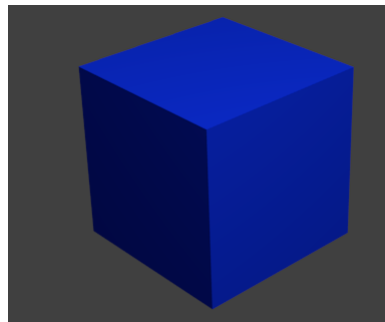
V případě zaškrtnutí položky *Sphere* se rovná poloměru koule definované vzdáleností *Distance*. Na obrázku 7.5 je vidět nasvícení objektu bodovým světlem s nastavenou vzdáleností 30 jednotek. Roletovou nabídkou s názvem *Falloff* se nastavuje úbytek intenzity vzhledem ke vzdálenosti od zdroje. Zde máme k dispozici několik předvoleb

- *Constant* – konstantní úbytek intenzity
- *Inverse Linear* – Intenzita ubývá inverzně a lineárně
- *Inverse Square* – Intenzita ubývá inverzně s druhou mocninou
- *Custom Curve* – Libovolná křivka, kdy v nastavení přibude nabídka *Falloff Curve*, kde si můžeme libovolně křivku poklesu intenzity tvarovat
- *Lin/Quad Weighted* – Lineární a kvadratický váhový úbytek intenzity, který lze



Obr. 7.4: Panel *Preview* a *Material* pro práci s *materiály*

nastavit numerickými tlačítky *Line* a *Quad*. Hodnoty se pohybují v intervalu $< 0; 1 >$

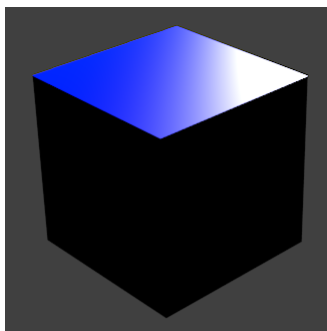


Obr. 7.5: Nasvícení objektu pomocí bodového světelného zdroje s lineárním úbytkem intenzity

7.4.2 Plošný zdroj

Plošný zdroj je planární, má konečnou plochu a vyzařuje paprsky do předního poloprostoru všemi směry. Tento typ světelného zdroje způsobuje stíny s neostrou hranicí. Používá se pro simulaci plošných směrových zdrojů jako je např. televizní obrazovka, světelná tabule apod. V nastavení předvoleb pro tento typ světla přibyla

nabídka *Area Shape*, kde určujeme jestli má být tvar světelného zdroje obdélníkový nebo čtvercový. Ve scéně je tento typ světla reprezentován jako prostý bod, ze kterého vychází přerušovaná úsečka udávající směr světla. Směr lze měnit příkazem rotace (klávesová zkratka **R**). Na obrázku 7.6 je vidět nasvícení tímto typem světla.

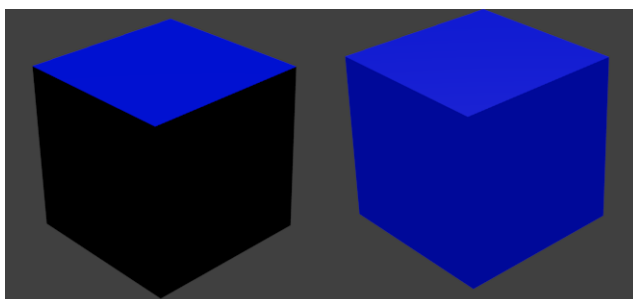


Obr. 7.6: Nasvícení objektu pomocí plošného směrového zdroje

7.4.3 Rovnoběžný zdroj

Rovnoběžný směrový zdroj v Blenderu, označován jako *Sun*, může být reprezentován jako bodový zdroj ležící v nekonečnu. Jeho paprsky dopadají rovnoběžně. Simulují se jím velmi vzdálená světla jako je např. sluneční svit. Paprsky světla mají v daném směru stejnou intenzitu, a proto nezáleží na jeho přesné poloze. V Blenderu se používá ještě typ světla, který je velmi podobný svými vlastnostmi typu *Sun* a používá se pro simulace, kdy světlo přichází ze zatažené oblohy. Tento světelný zdroj se nazývá *Hemi* a lze si jej představit jako polokouli, která obklopuje scénu. Nevýhodou je,

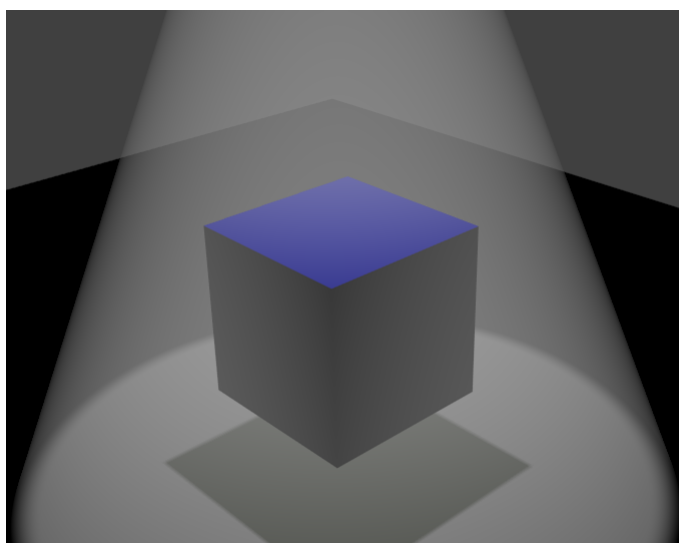
že nevrhá vůbec žádné stíny. Používá se spíše v kombinaci s dalšími světelnými zdroji. Oba světelné zdroje jsou na obrázku 7.7.



Obr. 7.7: Nasvícení objektu pomocí světelného zdroje typu *Sun* a *Hemi*

7.4.4 Reflektor

Posledním typem světelného zdroje je reflektor (*Spot*). Jde o směrově závislý zdroj světla, který je určen polohou a směrem, kterým vyzařuje. Geometricky je možné popsat reflektor jako kužel. V počítačové grafice je často používán, protože jako jediný ze všech zdrojů dokáže vrhat stín. V Blenderu je reflektor reprezentován bodem, ze kterého vychází kužel světla. Reflektor má možnost nastavení zapnutí funkce *Halo*, která způsobí viditelnost světelných paprsků jako je tomu na obrázku 7.8. Viditelnosti paprsků se používá u volumetrického osvětlení, kdy je žádoucí, aby paprsky byli viditelné např. u simulace slunečního světla skrze okno. Tato možnost se dá nastavit pouze u světelného zdroje typu *Spot*.



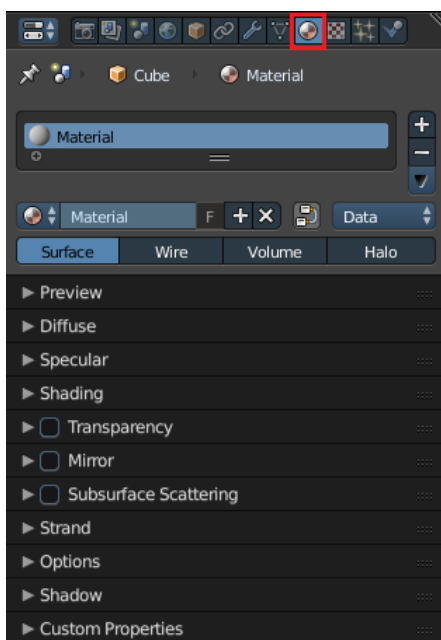
Obr. 7.8: Nasvícení objektu pomocí světelného zdroje typu *Spot*

8 MATERIÁLY

Jako materiál si můžeme představit barvu, která má určité vlastnosti. Vezměme si například modrou barvu, která může mít celou škálu odstínů od světlých až po velmi tmavé. Barva také může být zářivá nebo naopak matná nebo může přecházet z jednoho odstínu do jiného. Možnosti jsou téměř nekonečné. Při vykreslování celé scény je potřeba určit barvu každého pixelu. Nejlépe se toto provádí pomocí zpětného sledování paprsku. Pokud chceme určit barvu bodu, zpětně sledujeme paprsek světla a hledáme průsečík s libovolným objektem ve scéně. Následně se v tomtě bodě určuje, který světelný zdroj na něj působí. Na výslednou barvu mají vliv dopadající světelné paprsky ze zdroje a vlastnosti povrchu (hrubost, průhlednost, apod.).

8.1 Materiálové nastavení

Nastavení materiálů se nachází v okně předvoleb pod ikonou zobrazenou na obrázku 8.1. Výchozí nastavení materiálu v Blenderu je matná šedivá barva a každý objekt přidaný do scény bude mít stejné nastavení materiálu. Jednotlivé materiály jsou vidět v materiálovém okně, kde je možné, je pomocí tlačítek plus a minus, přidávat nebo odebrat jednotlivé pozice. Náhled vzhledu povrchu lze vidět v okně *Preview*, kde si lze měnit jednotlivé objekty, na kterých je náhled ukázán (plocha, koule, krychle, hlava opice Suzanne, statické částice a koule s pozadím). Každý povrch



Obr. 8.1: Okno předvoleb pro nastavení materiálu

může definován následujícími vlastnostmi:

- *Diffuse* na tomto povrchu dochází k vícenásobnému odrazu a lomu
- *Specular* okamžitě se zrcadlově odráží

Intenzita odraženého světla je dána součtem spekulární a difúzní složky, které se nazývají také jako *shadery*. Charakteristikou pro spekulární složku je její směrovost, která narůstá s hladkostí povrchů. Difúzní složka nezávisí na směru pohledu. Velikost této složky závisí na úhlu dopadu světelného paprsku a přináší informace o barvě povrchu [12]. S těmito složkami se dá v Blenderu dále manipulovat konkrétně v záložkách *Diffuse*, kde nastavujeme intenzitu pomocí posuvného tlačítka *Intensity* a typ použitého shaderu. U vlastností typu *Specular* nastavujeme intenzitu jako v předchozím případě, dále tvrdost a typ stínování.

8.2 Typy stínování

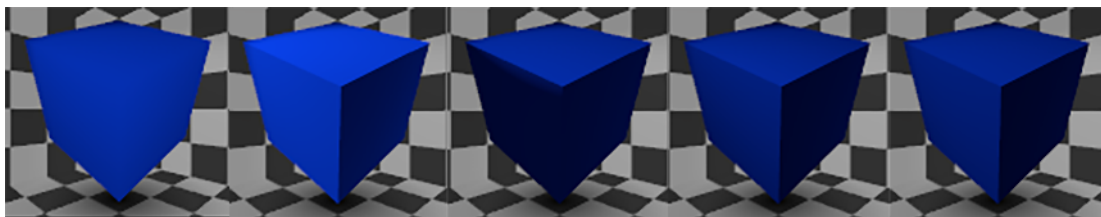
Pro difúzní stínování máme k dispozici tyto typy:

- *Fresnel* se používá pro materiály jako je sklo nebo voda. Má dva parametry. Parametrem *Fresnel* se nastavuje množství odraženého světla a parametrem *Factor* se určuje poměr míchání intenzity a množství odraženého světla.
- *Minnaert* najde použití u materiálů se zvýrazněným okrajem objektů např. ošacení. Tento typ stínování má jeden parametr *Darkness*, kterým se nastavuje světlost respektive tmavost.
- *Toon* je nerealistický shader, pro který jsou charakteristické jasné hranice světla a stínů. Parametrem *Size* lze nastavovat velikost osvětlených ploch a parametrem *Smooth* vyhlazenost přechodu mezi světlými a tmavými místy.
- *Lambert* je implicitně nastaven ke každému materiálu. Je základním typem pro stínování.
- *Oren-Nayar* je fyzikálně přesnějším modelem oproti *Lambertu*. V nastavení je parametr *Roughness*, kterým je možné nastavit mikroskopické nerovnosti povrchu.

Na obrázku 8.2 je vidět srovnání jednotlivých typů stínování pro difúzní vlastnosti materiálu.

Spekulární stínování nalezneme v záložce *Specular* a dále je to shodné jako v případě záložky *Diffuse* s rozdílem, že spekulární stínování má své typy.

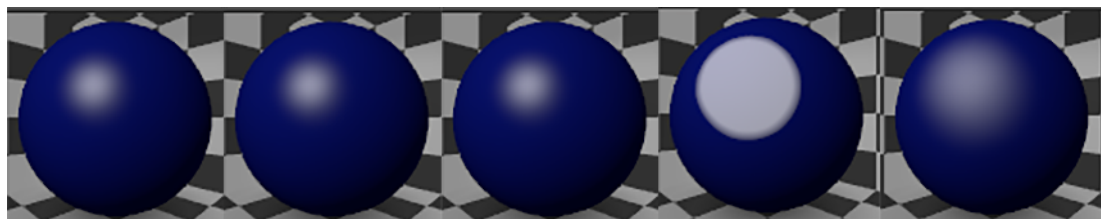
- *Cook Torr* je implicitně nastavený a od verze Blenderu 2.27 se stal téměř standardem. V nastavení jsou parametry *Intensity* a *Hardness*, kterým nastavujeme hrubost povrchu.
- *Phong* využívá jiný typ algoritmu oproti typu *Cook Torr*, ale výsledky jsou velmi podobné. Parametry jsou shodné jako v předchozím případě.



Obr. 8.2: Jednotlivé typy difúzního stínování. Zleva *Fresnel*, *Minnaert*, *Toon*, *Lambert*, *Oren-Nayar* s intenzitou 0,5.

- *Blinn* je zde podobnost s *Oren-Nayar*, kdy je fyzikálně přesnější a oproti předcházejícím typům má navíc parametr *IOR*, který reprezentuje nastavitelnou hodnotu indexu lomu.
- *Toon* obdoba stejného typu pro difúzní stínování, ale s rozdílem, že zde chybí parametr *Hard*.
- *WardIso* se používá pro materiály s plastickým vzhledem.

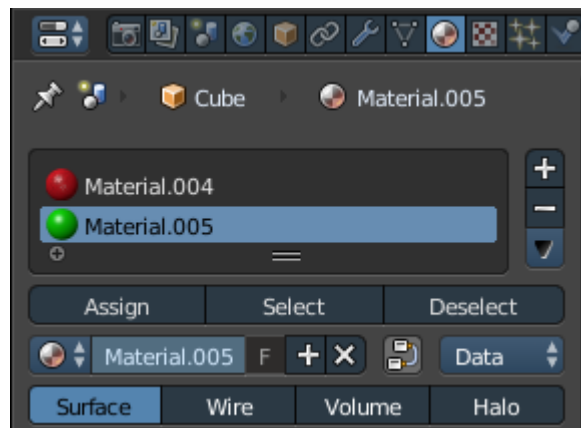
Na obrázku 8.3 je opět vidět srovnání jednotlivých typů stínování spekulární složku.



Obr. 8.3: Jednotlivé typy spekulárního stínování. Zleva *Cook-Torr*, *Phong*, *Blinn*, *Toon*, *WardIso* s intenzitou a tvrdostí nastavenou na hodnotu 0,5.

8.3 Nastavení více materiálů jednomu objektu

V Blenderu máme možnost přiřadit každé ploše jiný materiál. Vezměme si krychli a přiřadíme ji tři různé barvy. Nejprve se musíme přepnout do editačního módu (klávesová zkratka **Tab**) a v předvolbách pro materiály máme k dispozici tlačítko *Assign* jako na obrázku 8.4. Pokud si vybereme jednotlivé vertexy, které tvoří jednu stranu krychle a vytvoříme nebo vybereme materiál. Následně klikneme na tlačítko *Assign*. Tímto způsobem můžeme pro každou stranu krychle vybrat jiný materiál.



Obr. 8.4: Přiřazení více materiálů jednomu objektu

9 TEXTURY

Povrchy skutečných objektů jsou jiné, než jak jsme si popsali v kapitole o materiálech. Ve skutečnosti jsou nedokonalé, mají různé barevné vzory, nerovnosti, vytváří složité zrcadlové odrazy atd. Pomocí textur jsme schopni tyto vlastnosti dobře nasimulovat. Nejprve potřebujeme texturu vytvořit a nastavit jí jednotlivé materiálové parametry a nakonec jí namapovat na objekt. Za pomoci textur jsme schopni vytvořit velmi realisticky vypadající povrchy.

9.1 Mapování textur

Představme si mapování textury na příkladu. Máme k dispozici válcový objekt, na který chceme umístit etiketu (namapovat texturu). V praxi nalepit obrázek na láhev není problematické, obdobně jsme schopni nalepit obrázek na hranatý objekt. Čím bude, ale objekt složitějšího tvaru, tím složitější je mapovací proces. Mapování textury je složeno ze tří kroků. Definicí textury, prostorem (rovinná nebo trojrozměrná) a zda-li jde o texturu procedurální nebo tabulkovou. Dále tvarem objektu, na který texturu nanášíme a nakonec mapovanou veličinou. Nejjednodušším případem je mapování rovinné textury. Rovinná textura je definována jako funkce $T(u, v)$ přiřazující bodům $[u, v]$ v rovině hodnoty namapované veličiny

$$T : D_T \rightarrow H_T, D_T \subset \mathbf{R}^2.$$

Pro aplikaci textury musíme definovat funkci $M(x, y, z)$, která přiřazuje každému bodu na povrchu objektu bod z definičního oboru textury T (Označeno D_T). Funkci M říkáme inverzní mapování (inverse mapping) a je využívána pro nanášení textury:

$$M : D_M \rightarrow D_T,$$

kde $D_M \subset \mathbf{R}^3$ jsou body na povrchu tělesa [23]. Při vykreslování známe souřadnice $[x, y, z]$ a hledáme k nim informace o zobrazované textuře. Zobrazení je složením funkcí $M \circ T$. Funkci M je nutno volit tak, aby odpovídala tvaru tělesa, na které je nanášena. Pokud povrch tělesa tvoří jediná analytická plocha, je možné funkci M volit jako inverzní. Pro plochy, které lze rozvinout do roviny je možné najít funkci M , u které nedojde ke zkreslení plochy. U zborcených ploch dojde ke zkreslení textury a je problematické dosáhnout správného navázání textury.

9.1.1 Inverzní mapování válcové a kulové plochy

Pro určení inverzní mapovací funkce válcové plochy umístíme válec rovnoběžně s osou z a střed podstavy v počátku soustavy souřadnic. Mapovat budeme tak,

aby celý plášť válce byl v oboru $\langle 0, 1 \rangle \times \langle 0, 1 \rangle$. Do výpočtu zahrneme výšku h a poloměr válce r . Vyjdeme z popisu válcové plochy v cylindrických souřadnicích, $[x, y] = [r \cos \alpha, r \sin \alpha]$, ze kterého odvodíme inverzní funkci $M(x, y, z)$:

$$u = \begin{cases} \frac{1}{2\pi} \arccos \frac{x}{r} & \text{pro } y \leq 0 \\ 1 - \frac{1}{2\pi} \arccos \frac{x}{r} & \text{pro } y > 0 \end{cases} \quad (9.1)$$

$$v = \frac{z}{h}. \quad (9.2)$$

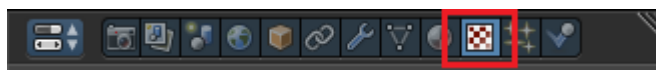
kde funkce \arccos nabývá hodnot z intervalu $\langle 0, \pi \rangle$. V příkladě je čtvercová textura mapována na plášť válce, kdy po rozvinutí do roviny má tvar obdélníku. Tím dochází k neuniformní změně měřítka textury ve vodorovné a svislé ose. Pro kulovou plochu postupujeme zcela analogicky. Koule o poloměru r je umístěna tak, aby střed ležel v počátku souřadnicového systému a body vyjádříme ve sférických souřadnicích $[x, y, z] = [r \cos \alpha \cos \beta, r \sin \alpha \cos \beta, r \sin \beta]$, kde $\alpha \in \langle 0, 2\pi \rangle$ a $\beta \in \langle -\frac{\pi}{2}, \frac{\pi}{2} \rangle$. Inverzní funkce má potom tvar:

$$u = \begin{cases} \frac{1}{2\pi} \arccos \frac{x}{\sqrt{x^2+y^2}} & \text{pro } y \leq 0 \\ 1 - \frac{1}{2\pi} \arccos \frac{x}{\sqrt{x^2+z^2}} & \text{pro } y > 0. \end{cases} \quad (9.3)$$

$$v = 0,5 + \frac{1}{\pi} \arcsin \frac{z}{r}, \quad (9.4)$$

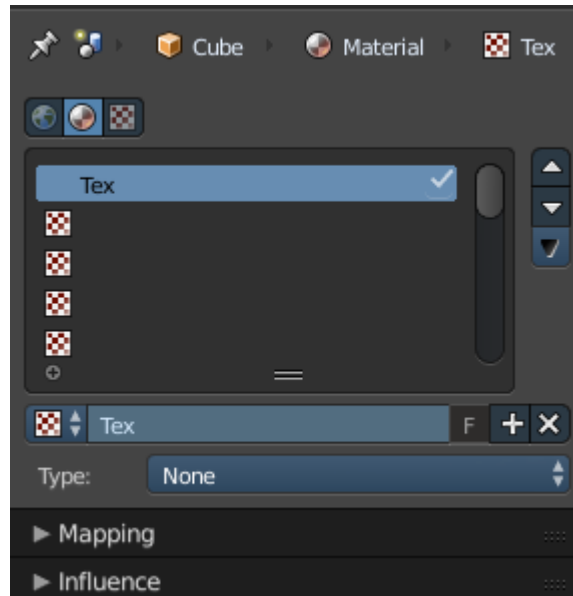
kde funkce \arccos nabývá hodnot z intervalu $\langle 0, \pi \rangle$ a \arcsin nabývá hodnot z intervalu $\langle -\frac{\pi}{2}, \frac{\pi}{2} \rangle$. Textura v pólech degeneruje na jediný bod, kvůli případu dělení nulou v odmocnině ze vzorce. Pokud je textura menší než objekt, na který se mapuje, nastane případ, že se ukončí a nebo se opakuje.

V Blenderu najdeme textury v panelu předvoleb pod ikonou šachovnice jako je na obrázku 9.1. V základu uvidíme panely jako je na obrázku 9.2, kdy je stan-



Obr. 9.1: Ikona pro textury v panelu předvoleb

dardně stisknuté tlačítko pro materiály, protože nejčastěji přiřazujeme textury materiálům. Dalšími volbami je možnost přiřadit texturu okolí scény (World) nebo stěpci (Brush). V hlavním okně vidíme název textur a zde také vytváříme nové textury. Vždy může být aktivní pouze jedna textura. Položkou *Type* si zvolíme typ textury. Na výběr máme širokou paletu možností. Většina je generována Blenderem na základě implementovaných algoritmů. Jedná se o tzv. procedurální textury. Na výběr máme možnosti jako jsou např. dřevo (*Wood*), šum (*Noise*), mraky (*Clouds*), mramor (*Marble*). Často používaným typem pro textury je obrázek (*Image*), kdy je možné



Obr. 9.2: Základní nastavení textur

načíst texturu z bitmapového souboru. Texturu může tvořit také animace. Pro příklad vybereme texturu typu obrázek (*Image*). Jako první musíme načíst texturu ze souboru. To provedeme v panelu *Image* tlačítkem *Open*. Otevře se nám vestavný prohlížeč souborů, kde zadáme cestu k obrázku. Po načtení textury ji bychom měli vidět

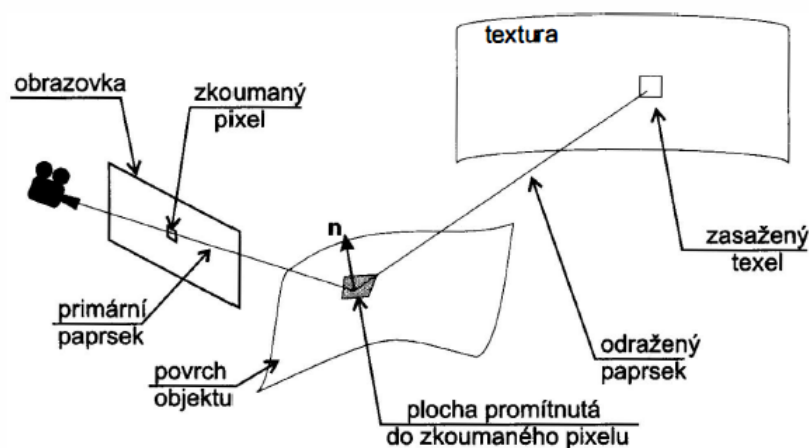
v panelu *Preview*. Důležité nastavení je v panelu *Image Mapping*, kde specifikujeme jak se textura bude na objektu chovat. Z roletové nabídky s názvem *Extension* vybíráme, jak se má textura chovat, pokud nepokryje celou plochu objektu. Na výběr máme

z možností *Extend*, kdy jsou konce textury prodlouženy, *Clip* nastaví nepokryté části objektu průhlednost, *Clip Cube* je totéž, akorát jsou započítány souřadnice osy z , *Repeat* zajistí opakování obrázku tak, že je obrázek pokládán dlaždicově vedle sebe, dokud nepokryje celý objekt a *Checker* vytváří šachovnicový vzor, kdy obrázek vytváří jeden typ pole. Načtený obrázek můžeme oříznout pomocí posuvných tlačítek *Crop Minimum* a *Crop Maximum*, kdy máme k dispozici osy x a y . A jako poslední možností pro mapování obrázku na objekt je jeho opakování *Repeat*, kdy opět nastavujeme opakování a zrcadlení v ose x nebo y .

9.2 Mapování prostředí (environment mapping)

Modeluje zrcadlení textury na povrchu texturovaného objektu, čímž se aproximuje odraz okolního prostředí objektu na jeho povrchu [23]. Často je technika nazývána

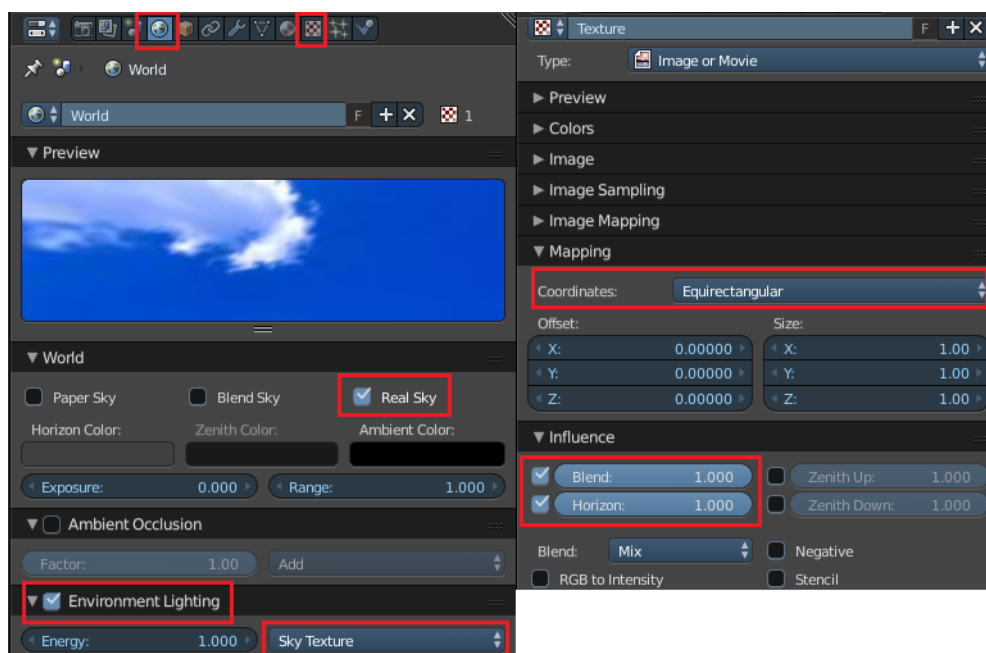
chromové mapování, protože je typická pro zrcadlové materiály. Odraz od povrchu objektu záleží na poloze pozorovatele, tělesa a okolním prostředí. Tato textura je pohybově závislá. V praxi to znamená, že pokud bude objekt v pohybu, dojde i ke změně textury na povrchu objektu. Využívá faktu, že lidské oko si nevšimne aproximace prostředí, a že reálný odraz by vypadal jinak. S touto technikou se často setkáváme v prostředích, kde je kladen důraz na rychlost, jako je například ve filmu. Principem této metody je uzavření texturovaného objektu do jiného objektu a aplikace textury na jeho vnitřek. Z polohy pozorovatele a bodu na obrazovce se určí primární paprsek. Pokud primární paprsek protne texturovaný objekt v nějakém bodě, určí se z průsečíku, primárního paprsku a normálového vektoru odražený paprsek. Princip je zobrazen na obrázku 9.3.



Obr. 9.3: Princip mapování prostředí. Převzato z [23]

Možnosti jak provést enviroment mapping v Blenderu máme dvě. První z nich je využití *Cycles renderu*, který má implementovány funkce pro mapování prostředí. A druhá možnost s využitím standardního vykreslování, což si ukážeme na příkladu. Potřebujeme do scény přidat plochu (*Plane*), na kterou namapujeme texturu šachovnice a dále použijeme místo krychle koule. Spodní část koule bude odrážet texturu z plochy a zeshora bude odrážet texturu nebe. Pro namapování textury nebe půjdeme do panelu předvoleb pro svět *World*, kde je potřeba zaškrtnout předvolby *Real Sky* a *Environment Lighting* s možností *Sky Texture*, jako je uvedeno na obrázku 9.4.

Poté se přepneme do předvoleb pro textury a vybereme texturu typu *Image* a v předvolbě *Mapping* je důležité přepnout souřadnice na *Equirectangular* (ekvidistantní válcovou projekci) a dále v předvolbě *Influence* zaškrtnout *Blend* a *Horizon* (aktivuje horizont a získáme výsledky podobné těm v reálném světě) s hodnotami rovné 1. Jako poslední krok musíme nastavit zrcadlový materiál koule. V nastavení *Material* zaškrtneme předvolbu *Mirror* a nastavíme *Reflectivity* na hodnotu 1. Tím



Obr. 9.4: Nastavení Blenderu pro mapování prostředí

docílíme totálního zrcadlení okolních materiálů. Výsledek je vidět na obrázku 9.5.

9.3 Hrbolaté textury (bump texture)

Metoda způsobuje optický dojem hrbolatého povrchu, bez změny geometrie tělesa. Je to způsobeno normálovým vektorem, který je při mapování tělesa pozměněn tak, aby změnil směr odrazu světla způsobem, který odpovídá lokálnímu zvrásnění hrbolatého tělesa. Vezměme bod, který je určen vektorovou funkcí $\vec{p}(u, v)$ a normálový vektor $\vec{n}(u, v)$. Budeme předpokládat, že velikost normálového vektoru je rovna jedné. Funkci zvrásnění označme $F(u, v)$. Udává vzdálenost, která určuje texturu od hladkého povrchu. Pokud posuneme původní bod ve směru normálového vektoru o vzdálenost F dostaneme bod \vec{p}' na novém povrchu

$$\vec{p}' = \vec{p} + F\vec{n}.$$

Dále určíme nový normálový vektor \vec{n}' . Parciální derivací v novém tečném směru získáme tečné vektory \vec{p}'_u a \vec{p}'_v a pokud je změna malá lze tyto vektory vyjádřit jako:

$$\vec{p}'_u = \vec{p}_u + F_u\vec{n} \quad (9.5)$$

$$\vec{p}'_v = \vec{p}_v + F_v\vec{n}, \quad (9.6)$$

kde \vec{p}_u je tečný vektor k původnímu povrchu a F_u je derivace mapy posunutí. Nový normálový vektor je dán součinem tečných vektorů k novému povrchu:

$$\vec{n}' = \vec{p}'_u \times \vec{p}'_v = \vec{n} + \vec{d},$$



Obr. 9.5: Výsledek mapování prostředí

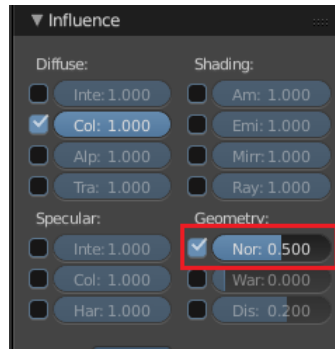
$$\vec{d} = F_u[\vec{n} \times \vec{p}_u - F_v[\vec{n} \times \vec{p}_v]]. \quad (9.7)$$

Hrbolatou texturu získáme, pokud nalezneme tečné vektory k funkci určující posunutí a k původnímu povrchu objektu a výpočtu normálového vektoru, potom aplikovaná textura může nabývat libovolné podoby. Typickým příkladem hrbolaté textury je textura zdiva.

Pokud chceme v Blenderu namapovat hrbolatou texturu, je nutné nejdříve v okně předvoleb vybrat záložku *Material*, kde v nastavení *Specular* nastavíme hodnotu odlesků na nulovou hodnotu. Dále se přepneme na okno předvoleb pro textury, kde vybereme texturu typu *Image*. Pokud si necháme vykreslit výsledek, chybí textuře určitá plasticita. Podstatou je vytvořit normálovou mapu textury, což je textura převedená do odstínů stupnice šedi. V posledních verzích Blenderu je toto možné dělat interně bez použití grafických editorů. Půjdeme do předvoleb *Texture* a v nastavení *Influence* aktivujeme položku *Normal* v předvolbě *Shading*, jako je na obrázku 9.6. Výsledek je zřejmý z obrázku 9.7.

9.4 UV mapování

UV mapování je proces nanášení bitmapových textur na trojrozměrné modely. Jeho hlavní použití je pro složitější objekty jako jsou zvířata nebo lidi. Bitmapy používané pro UV mapování jsou vytvořeny v grafických programech a lze dosáhnout



Obr. 9.6: Nastavení 3D efektu pro bump mapping



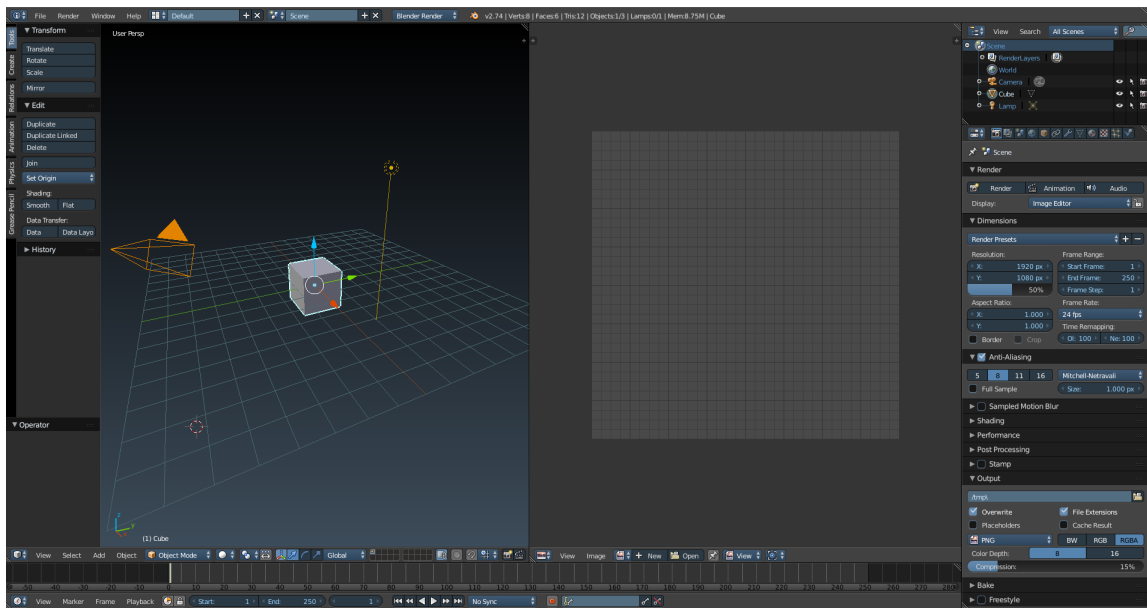
Obr. 9.7: Původní textura cihel vlevo a vpravo textura bump mapping

velmi dobrých výsledků. Pokud chceme použít UV mapování na objekt je nejprve třeba provést proces *UV unwrap*, což v praxi znamená objekt rozložit na plochy. Výsledkem je síť určující, která část se na jakou plochu nanáší. Tuto texturu je možné vyexportovat do bitmapy a tu následně editovat v grafickém editoru.

Po úpravě texturu opět namapujeme na objekt jako UV. Celý proces se zdá složitý, ale jeho hlavní výhodou je přesnost. V Blenderu se UV mapování provádí pomocí okna *UV/Image editor*. Ten umožňuje mapovat jednotlivé plochy typu *mesh*. Každá plocha má na textuře přiděleny své souřadnice. Pro příklad si předvedeme UV mapování na krychli. Je vhodné si hlavní 3D okno rozdělit na dvě části jako, je uvedeno na obrázku 9.8. Vybereme objekt, který chceme mapovat a přepneme se do editačního režimu (klávesa **Tab**). V panelu nástrojů (obrázek 9.9) se přepneme na záložku *Shading/UVs*.

V rozbalovacím menu *Unwrap* najdeme možnosti (popíšeme pouze nejdůležitější možnosti):

- *Unwrap* – používá se pro vytvoření UV sítě za pomoci švů. Tímto způsobem je možné rozbalit většinu objektů
- *Cube Projection* – kubické mapování
- *Cylinder Projection* – cylindrické mapování
- *Sphere Projection* – sférické mapování



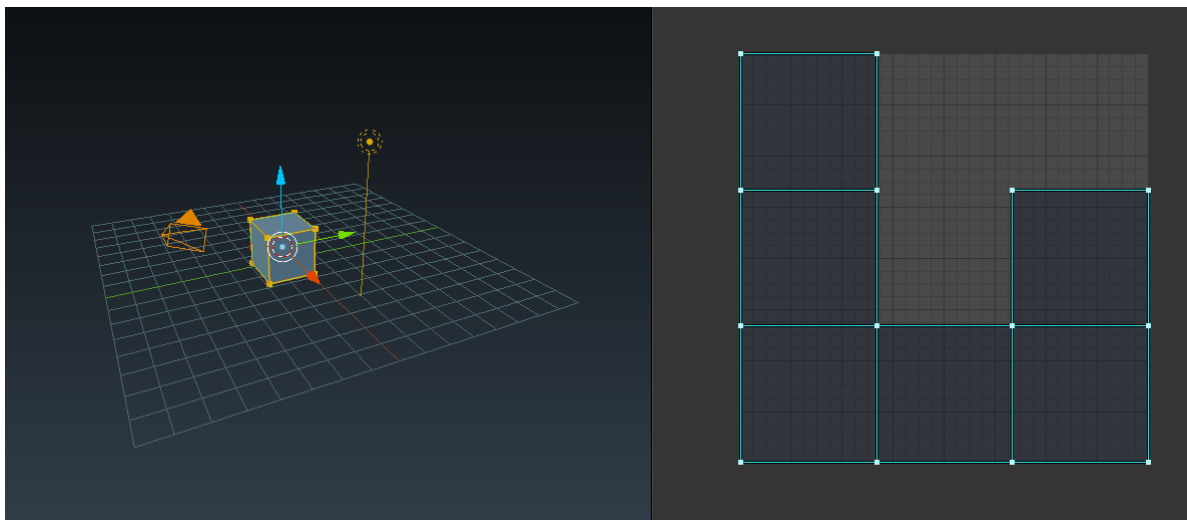
Obr. 9.8: Vzhled 3D okna pro UV mapování



Obr. 9.9: Menu pro UV mapování

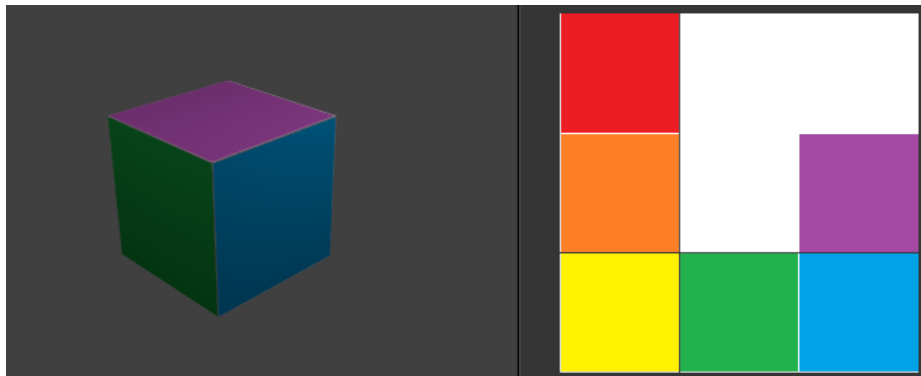
- *Project From View* – mapování odpovídající pohledu 3D scény
- *Reset* – vymazání namapované sítě

Všechny výše uvedené druhy rozložení objektu způsobí v okně *UV/Image editor* zobrazení výsledné sítě. S roloženou sítí je možné provádět běžné transformace jako je posun, rotace, změna měřítka a zarovnání souřadnic x a y . Pokud chceme tedy krychli rozbalit zvolíme volbu *Mark Seam*, která označí všechny švy na objektu a zvolíme *Unwrap* (klávesa **U**) a získáme mapu podobnou jako je na obrázku 9.10.



Obr. 9.10: Krychle a její UV mapa

Pro export mapy zvolíme v UV editoru položku *UVs* a *Export UV Layout*. Dále mapu lze zpracovat v grafickém editoru. Posledním krokem, který musíme provést je správné namapování zpět. Vytvoříme nový materiál a texturu a zvolíme typ *Image*. V záložce *Mapping* v nabídce textur změníme *Coordinates* na *UV* a v UV editoru zvolíme nabídku *Image* a *Open Image*. Pokud si ve scéně zvolíme způsob zobrazení *Texture*, bude textura zobrazena přímo na objektu ve scéně. Výsledek je vidět na obrázku 9.11. Obdobně bychom postupovali u složitějších objektů.



Obr. 9.11: Nakreslená textura krychle vpravo a obrázek finálního vykreslení

10 ZÁVĚR

V první části diplomové práce jsme se zaměřili na přehled výuky grafiky na českých univerzitách a následně jsme provedli srovnání jednotlivých 3D programů, ve kterých se praktická cvičení vyučují. 3D grafika se vyučuje celkem na 5 univerzitách v České republice. Nejčastěji je použit program Cinema 4D od firmy Maxon. Tento program má českou lokalizaci, což je samozřejmě výhodou, ale hlavní výhodou je velká uživatelská základna v České republice a s tím související dostupné velké množství materiálů pro podporu výuky. Druhým programem je Blender, jehož velkou výhodou je jeho dostupnost. Je šířen pod licencí GNU GPL, tudíž se jedná o volně dostupný software. Toto vnímáme jako velkou výhodu, protože studenti, kteří nikdy nepracovali s žádným 3D softwarem, si mohou vyzkoušet jednotlivé postupy a v případě zájmu se přeorientovat na jiný komerční produkt. Všechny univerzity, které jsou zmíněny v diplomové práci byly osloveny, zda-li by poskytly bližší informace o výuce, ale bohužel se nevyjádřily a byli jsme odkazováni pouze na anotace předmětů.

V další části jsme se stručně zaměřili na vývoj programu Blender. Tento program prošel od svého začátku velkými změnami a veškeré významné změny jsme se pokusili popsat v jeho jednotlivých verzích. Dnes tento program může konkurovat ostatním komerčně dostupným programům.

V teoretické části jsme popsali základní ovládání programu, protože Blender se může na první pohled zdát chaoticky uspořádaný a základní práce s ním je mírně odlišná od ostatních programů, které uživatelé znají. Tato problematika je probrána v prvním videotutoriálu, kde jsme se zaměřili na orientaci v programu po prvním spuštění a stručný popis jednotlivých oken včetně jejich základních nastavení. Videotutoriály jsou koncipovány tak, aby uživatel věděl, kde se nastavuje vykreslení výsledné scény, kde najde informace o objektu a jak objekt může transformovat a editovat, a nakonec si výslednou scénu uložit v požadovaném rozlišení do konkrétního formátu.

Druhý videotutoriál je věnován nastavení jednotlivých pohledů na scénu, a s tím je spojená práce s jednotlivými okny, protože toto patří mezi základní úkony, které musí uživatel znát, aby dále byl schopný modelovat jednotlivé scény. Po úvodním představení práce s programem Blender jsme teoreticky popsali jednotlivé transformace v rovinném i trojrozměrném prostoru, které lze s objektem provádět. Dále jsme si popsali používání homogenních souřadnic. Praktická část je zaměřena na rozdíl mezi objektovým a editačním módem a možnosti vkládání jednotlivých objektů do scény. Součástí videotutoriálu jsou transformační příkazy a používání manipulatorů. V kapitole o osvětlovacích metodách jsou popsány základní radiometrické pojmy, které jsou důležité pro globální osvětlování. Velká část kapitoly je věnována

popisu chování odraženého světla, protože většinu světla, kterou vnímáme lidským okem je světlo odražené. Odrazové schopnosti povrchu materiálu v určitém bodě popisuje dvousměrová odrazová distribuční funkce (BDRF). Pomocí BDRF jsme schopni vyjádřit velikost odražené radiance dopadající na plochu. Této funkci se využívá v počítačové grafice ve spojení s chováním materiálů. V praktické části jsme se zaměřili na popis jednotlivých světelných zdrojů a jejich použití.

Poslední dvě kapitoly jsou věnované materiálům a texturám. V počítačové grafice si lze představit materiál jako barvu, která má určité vlastnosti. Materiál jako celek se skládá ze dvou složek a pro každou z nich je nutné definovat, jak se budou jednotlivé složky chovat. To je předmětem praktické části. Abychom docílili realistického vzhledu je nutné přidat materiálům texturu. Pomocí textur jsme schopni namapovat na objekt textury z fotografií, upravených obrázků, apod. Díky texturám je možné vytvořit a přesně namapovat složité objekty, jakým je například lidský obličej. Tomuto typu texturování se říká UV mapování, na které jsme se zaměřili v posledním videotutoriálu.

Celkově jsme vytvořili osm podpůrných video materiálů, které jsou součástí této diplomové práce. Videá jsou určena začátečníkům, kteří si chtějí vyzkoušet práci s 3D programem Blender. Všechny popisy jsou demonstrovány na elementárních příkladech a koncipovány tak, aby člověk, který uvidí program poprvé, byl schopen orientovat se v programu a vytvářet jednotlivé úkony s objekty.

LITERATURA

- [1] *3D modelační a animační software Autodesk 3Ds Max* [online]. [cit. 26. 10. 2014]. Dostupné z URL: <<http://www.autodesk.com/products/3ds-max/features/all/gallery-view>>.
- [2] *3D modelační a animační software Autodesk Maya* [online]. [cit. 26. 10. 2014]. Dostupné z URL: <<http://www.autodesk.com/products/maya/features/all/gallery-view>>.
- [3] *3D software Cinema 4D* [online]. [cit. 18. 10. 2014]. Dostupné z URL: <<http://www.cinema4d.cz/produkty/maxon/cinema-4d-prime/>>.
- [4] *Anotace předmětu Kurz multimediálních aplikací na ČVUT v Praze* [online]. [cit. 18. 10. 2014]. Dostupné z URL: <<http://leyfi.felk.cvut.cz/courses/kma/>>>.
- [5] *Anotace předmětu Počítačová geometrie a grafika na VUT v Brně* [online]. [cit. 18. 10. 2014]. Dostupné z URL: <<http://www.old.fme.vutbr.cz/predmety/predmet.html?pid=136728&lang=0>>.
- [6] *Anotace předmětu Počítačová grafika II na Univerzitě Tomáše Bati ve Zlíně* [online]. poslední aktualizace 1. 12. 2014 [cit. 2. 12. 2014]. Dostupné z URL: <<http://ects.utb.cz/predmet/AUPKS/A5PGR?lang=cs&rocnik=3&statut=A>>>.
- [7] *Anotace předmětu Základy počítačové grafiky na Masarykově univerzitě v Brně* [online]. [cit. 18. 10. 2014]. Dostupné z URL: <<https://is.muni.cz/predmet/fi/jaro2015/PB009>>.
- [8] *Anotace předmětu Cinema 4D – obecné modelování* [online]. [cit. 18. 10. 2014]. Dostupné z URL: <<https://stavnet.fsv.cvut.cz/ anotace.php?katedra=375>>>.
- [9] BLAIN, John M. *The complete guide to Blender graphics: computer modeling and animation*. 1st ed. London: CRC Press, c2012, xvii, 389 p. ISBN 978-146-6517-035.
- [10] HUGHES, John F. *Computer graphics: principles and practice*. 3rd ed. Upper Saddle River: Addison-Wesley, 2013, xlvii, 1209 s. ISBN 978-0-321-39952-6.
- [11] JURCZYKOVÁ, Kristýna. *Stereometrie*. Praha, 2014. Dostupné z: http://www.karlin.mff.cuni.cz/katedry/kdm/diplomky/kristyna_jurczykova/index.php. Diplomová práce. Matematicko-fyzikální fakulta Univerzity Karlovy v Praze. Vedoucí práce RNDr. Jarmila Robová, CsC.

- [12] POKORNÝ, Pavel. *Blender: naučte se 3D grafiku*. 2., aktualiz. a rozš. vyd. Praha: BEN – technická literatura, 2009, 286 s. ISBN 978-80-7300-244-2.
- [13] *Historie programu Blender* [online]. [cit. 7. 11. 2014]. Dostupné z URL: <<http://wiki.blender.org/index.php/Doc:CZ/2.6/Manual/Introduction/History>>.
- [14] Osvětlení. In: *Mendelova univerzita v Brně* [online]. [cit. 2015-05-09]. Dostupné z: http://is.mendelu.cz/eknihovna/opory/zobraz_cast.pl?cast=19899
- [15] *Popis programu Blender verze 2.30* [online]. poslední aktualizace 3. 11. 2003 [cit. 8. 11. 2014]. Dostupné z URL: <<http://www.grafika.cz/rubriky/3d-grafika/blender-2-30-revolucni-verze-blenderu-ke-stazeni-131930cz>>.
- [16] *Popis programu Blender verze 2.60* [online]. [cit. 8. 11. 2014]. Dostupné z URL: <<http://archive.blender.org/development/release-logs/blender-260/index.html>>.
- [17] COHEN, Michael F a John R WALLACE. *Radiosity and realistic image synthesis*. Boston: Academic Press Professional, 1993, xv, 381 p. ISBN 01-217-8270-0.
- [18] *Renderovací nástroj RenderMan od společnosti Pixar* [online]. [cit. 29. 10. 2014]. Dostupné z URL: <<http://renderman.pixar.com/view/Production-Proven>>.
- [19] *3D software Rhinoceros* [online]. [cit. 29. 10. 2014]. Dostupné z URL: <http://www.rhino3d.com/sales/europe/Czech_Republic/all/?PageSpeed=noscript>.
- [20] TOMICZKOVÁ, Světlana. *Deskriptivní geometrie I: Pomocný učební text*. Plzeň, 2006. Dostupné z URL: <https://mat.fsv.cvut.cz/bakalari/kog/files/DEG_Svetlana.pdf>. Učební text. Západočeská univerzita v Plzni.
- [21] The graPHIGS Programming Interface: Understanding Concepts: View Orientation Information. IBM Knowledge Center [online]. [cit. 2015-05-23]. Dostupné z URL: <http://www-01.ibm.com/support/knowledgecenter/ssw_aix_53/com.ibm.aix.graPHIGS/doc/phigsund/ViewCap.htm%23hdrviewing>.
- [22] *Sylabus předmětu Principy a aplikace 3D grafiky* [online]. poslední aktualizace 11. 3. 2014 [cit. 18. 10. 2014]. Dostupné z URL: <<http://isis.vse.cz/katalog/syllabus.pl?predmet=100198>>>.

- [23] ŽÁRA, Jiří. *Moderní počítačová grafika*. Vyd 1. Brno: Computer Press, 2004, 609 s. ISBN 80-251-0454-0.

SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

AVI	formát multimediálního kontejneru pro synchronní přehrávání audia a videa – Audio Video Interleave
BMP	počítačový formát pro ukládání rastrové grafiky – BitMap
CAD	počítačem podporované projektování – Computer Aided Design
CPU	centrální procesorová jednotka – Central Processing Unit
FPS	snímková frekvence – Frames Per Second
GNU	projekt zaměřený na svobodný software – GNU's Not Unix
GPU	grafický procesor – Graphic Processing Unit
H.264	standart pro kompresi videa
HDTV	televize s vysokým rozlišením – High Definition Television
JPEG	metoda ztrátové komprese pro ukládání počítačových obrázků – Joint Photographic Experts Group
PNG	grafický formát pro bezztrátovou kompresi grafiky – Portable Graphics Network
RAM	zařízení pro čtení a zápis – Random Access Memory
SSE	instrukční sada typu SIMD (jedna instrukce, více dat) – Streaming SIMD Extensions
TIFF	souborový formát pro ukládání rastrové počítačové grafiky – Tagged Image File Format

SEZNAM PŘÍLOH

A Obsah přiloženého DVD

70

A OBSAH PŘILOŽENÉHO DVD

Obsahem přiloženého DVD je elektronická verze diplomové práce a osm videotutoriálů, které se zaměřují na:

- *Základní ovládání* – ukázka základního ovládání programu Blender
- *Práce s okny* – modifikace oken a způsoby jejich využití
- *Výběr objektů* – elementární práce s objekty a jejich výběrem
- *Transformace* – transformační příkazy a jejich aplikování na objekty ve scéně
- *Světla* – popis jednotlivých zdrojů osvětlení
- *Materiály a textury* – aplikace materiálů a textur na objekty
- *UV mapování* – ukázka UV mapování na elementárním příkladu
- *Kamery* – základní práce s kamerou

Vše je demonstrováno v programu Blender 2.74.