

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE  
FAKULTA ELEKTROTECHNICKÁ  
KATEDRA POČÍTAČOVÉ GRAFIKY A INTERAKCE



BAKALÁŘSKÁ PRÁCE

**Měření povrchu BRDF s využitím LIGHTEC Minidiff V2  
pro využití v programování her**

Lukáš Cezner

Vedoucí: prof. Ing. Vlastimil Havran, Ph.D.

Studijní program: Otevřená informatika

Květen 2023



## I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Cezner** Jméno: **Lukáš** Osobní číslo: **498875**  
Fakulta/ústav: **Fakulta elektrotechnická**  
Zadávací katedra/ústav: **Katedra počítačové grafiky a interakce**  
Studijní program: **Otevřená informatika**  
Specializace: **Počítačové hry a grafika**

## II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

**Měření povrchu BRDF s využitím LIGHTTEC Minidiff V2 pro využití v programování her**

Název bakalářské práce anglicky:

**BRDF surface reflectance using LIGHTTEC Minidiff V2 for Game Engines**

Pokyny pro vypracování:

Seznamte se s měřicím přístrojem Lighttec Mini Diff V2 pro měření isotropní odrazivosti povrchu a naučte se jej používat, data měřit a exportovat do stávajících formátů. Nastudujte si problematiku užití BRDF formou tzv. shaderů v programových prostředích pro psaní počítačových her a to nejméně v UNITY a UnrealEngine. Najděte způsob vyžití naměřených dat prostorové odrazivosti v uvedených programových prostředích pro nastavení vzhledu povrchu a to buď s využitím přímé cesty nebo konverze z dostupných datových formátů přístroje do vhodných datových formátů v programových prostředích her.

Změřte nejméně 200 různých povrchů, fyzické vzorky dodá vedoucí práce. Na vhodných demonstračních 3D tvarech demonstруйте funkčnost použití měřených dat, vytvořte krátká videa se zobrazením. Porovnejte rychlost zobrazení programových prostředích pro psaní počítačových her s měřeními daty a jednoduchými analytickými modely typu Phong. Znalost C++ a základů počítačové grafiky nezbytná. Podmínkou je absolvování alespoň souběžně s předmětem Počítačové hry.

Seznam doporučené literatury:

- 1) Lighttec Mini Diff V2  
<http://www.lighttec.fr/mini-diff-v2-release/>
- 2) Weinmann, Klein:  
Advances in geometry and reflectance acquisition (course notes),  
SA '15: SIGGRAPH Asia 2015 Courses, November 2015 Article No.: 1, Pages 1–71,  
<https://doi.org/10.1145/2818143.2818165>.
- 3) Lighttec Mini Diff V2 manual, 2021. K dispozici u vedoucího práce.

Jméno a pracoviště vedoucí(ho) bakalářské práce:

**prof. Ing. Vlastimil Havran, Ph.D. Katedra počítačové grafiky a interakce**

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Datum zadání bakalářské práce: **26.04.2023**

Termín odevzdání bakalářské práce: **26.05.2023**

Platnost zadání bakalářské práce: **16.02.2025**

prof. Ing. Vlastimil Havran, Ph.D.  
podpis vedoucí(ho) práce

podpis vedoucí(ho) ústavu/katedry

prof. Mgr. Petr Páta, Ph.D.  
podpis děkana(ky)

### III. PŘEVZETÍ ZADÁNÍ

Student bere na vědomí, že je povinen vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

\_\_\_\_\_  
Datum převzetí zadání

\_\_\_\_\_  
Podpis studenta

## **Poděkování**

Rád bych poděkoval svému vedoucímu práce prof. Ing. Vlastimilu Havranovi, Ph.D. za vedení a cenné rady. Také bych rád poděkoval své rodině a přátelům za jejich podporu při mém studiu.

## **Prohlášení**

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

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů, zejména skutečnost, že České vysoké učení technické v Praze má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona.

## Abstrakt

Tato práce se zabývá možností využití naměřených dat odrazivosti skutečných materiálů pro simulaci osvětlení povrchů v herních enginech Unity a Unreal Engine za účelem dosažení realistického vzhledu 3D objektů v počítačových hrách.

Nejprve jsou v úvodu popsány fyzikální aspekty BRDF a kategorizovány existující analytické modely BRDF. Dále jsou rozebrány některé formáty pro ukládání dat BRDF, popsán měřicí přístroj MiniDiff v2 od společnosti Synopsys (dříve Lightec) a proces měření vzorků.

Výsledkem práce jsou naměřená data 217 vzorků, vytvořená aplikace na zpracování a export naměřených dat BRDF, krátká videa zobrazení a shadery implementované v herních enginech Unity a Unreal Engine.

**Klíčová slova:** BRDF, Unity, Unreal Engine, OpenGL, odrazivost povrchu

## Abstract

This thesis investigates the possibility of using measured reflectance data of real materials to simulate the illumination of surfaces in Unity and Unreal Engine game engines in order to achieve realistic appearance of 3D objects in computer games.

First, the physical aspects of BRDF are described and existing analytical models of BRDF are categorized. Then, some BRDF data storage formats are discussed, the MiniDiff v2 measuring instrument from Synopsys (formerly Lightec) and the sample measurement process are described.

As a result of the work, 217 samples have been measured, an application has been developed to process and export the measured BRDF data, short videos of BRDF visualizations and shaders implemented in the Unity and Unreal Engine game engines.

**Keywords:** BRDF, Unity, Unreal Engine, OpenGL, surface reflectance

# Obsah

Obsah	VII
<b>1 Úvod</b>	<b>1</b>
1.1 Základní radiometrické veličiny . . . . .	1
1.1.1 Zářivý tok . . . . .	1
1.1.2 Zářivost . . . . .	1
1.1.3 Intenzita vyzařování . . . . .	2
1.1.4 Zář . . . . .	2
1.2 BRDF . . . . .	2
1.2.1 Směrově-hemisférická odrazivost (albedo) . . . . .	3
1.2.2 Dvou-hemisférická odrazivost (průměrné albedo) . . . . .	3
1.2.3 Symetrie . . . . .	3
1.2.4 Zachování energie . . . . .	3
1.2.5 Izotropie . . . . .	4
1.2.6 ARS . . . . .	4
1.2.7 BTDF . . . . .	4
1.3 Zobrazovací rovnice . . . . .	4
<b>2 Analytické modely BRDF</b>	<b>5</b>
2.1 Ideální modely . . . . .	5
2.1.1 Lambert . . . . .	5
2.1.2 Ideální odraz . . . . .	5
2.2 Empirické modely . . . . .	6
2.2.1 Phong . . . . .	6
2.2.2 Lafortune-Phong . . . . .	6
2.2.3 Ward . . . . .	6
2.3 Fyzikálně založené modely . . . . .	7
2.3.1 Fresnelovy rovnice a Schlickova aproximace . . . . .	7
2.3.2 Cook-Torrance . . . . .	7
2.3.3 Oren-Nayar . . . . .	8
2.3.4 Walter (GGX) . . . . .	8
2.3.5 Disney . . . . .	9
<b>3 Formáty pro ukládání BRDF</b>	<b>11</b>
3.1 Zemax . . . . .	11
3.2 MERL databáze . . . . .	12
3.3 BiRD JSON . . . . .	12
<b>4 MiniDiff v2</b>	<b>13</b>
4.1 Obslužný software . . . . .	13

4.1.1	Kalibrace . . . . .	14
4.1.2	Vizualizace . . . . .	14
4.1.3	Extrapolace a symetrie . . . . .	14
4.1.4	Export dat . . . . .	14
<b>5</b>	<b>Měření BRDF</b>	<b>15</b>
5.1	Časté problémy . . . . .	15
5.1.1	Nerovnosti povrchu . . . . .	15
5.1.2	Průsvitnost vzorku . . . . .	16
5.1.3	Měkkost vzorku . . . . .	16
5.2	Výběr vzorků . . . . .	16
5.2.1	Příklady nevhodných vzorků . . . . .	16
5.2.2	Použité vzorky . . . . .	17
<b>6</b>	<b>Implementace vizualizačního softwaru</b>	<b>19</b>
6.1	Knihovna brdf_parser . . . . .	19
6.1.1	Podporované formáty . . . . .	19
6.1.2	Slučování barevných složek . . . . .	20
6.1.3	Interpolace a extrapolace . . . . .	20
6.1.4	Datová struktura pro vyhledávání nejbližších sousedů . . . . .	22
6.2	Aplikace brdf_gui . . . . .	25
6.2.1	Zobrazení laloku . . . . .	26
6.2.2	Zobrazení modelu . . . . .	27
6.2.3	Vytvoření animace zobrazení . . . . .	28
6.2.4	Nastavení . . . . .	28
6.2.5	Slučování barevných složek . . . . .	31
6.2.6	Vytváření atlasu BRDF . . . . .	31
6.2.7	Volání z příkazové řádky . . . . .	32
<b>7</b>	<b>Implementace shaderů</b>	<b>33</b>
7.1	Přímé osvětlování . . . . .	33
7.2	Osvětlování mapou okolí . . . . .	35
<b>8</b>	<b>Vykreslovací řetězce herních enginů</b>	<b>37</b>
8.1	Unity . . . . .	37
8.1.1	Způsoby tvorby shaderů . . . . .	37
8.1.2	Možnosti implementace vlastní BRDF . . . . .	38
8.2	Unreal Engine 5 . . . . .	38
8.2.1	Způsoby tvorby shaderů . . . . .	38
8.2.2	Možnosti implementace vlastní BRDF . . . . .	39
<b>9</b>	<b>Použití měřených dat v herních enginech</b>	<b>41</b>
9.1	Unity . . . . .	41



9.2 Unreal Engine . . . . .	42
9.3 Porovnání rychlosti vykreslování . . . . .	44
<b>10 Závěr</b>	<b>45</b>
<b>Zdroje</b>	<b>47</b>
<b>Seznam obrázků</b>	<b>50</b>
<b>Seznam tabulek</b>	<b>51</b>
<b>Přílohy</b>	<b>53</b>
<b>A Seznam naměřených vzorků</b>	<b>53</b>
<b>B Náhledové obrázky vyříznuté z vytvořených videí</b>	<b>56</b>
<b>C Popis struktury přiložených souborů</b>	<b>58</b>



# 1 Úvod

Základním cílem počítačové grafiky je vytvoření snímku virtuálních modelů. K dosažení realistického výsledku je nutné splnit několik požadavků, mezi které patří i věrohodná simulace šíření světla a jeho interakcí s prostředím. Při dopadu světla na povrch objektu je nutné určit do jakých směrů se paprsek odrazí, tento jev popisuje matematická funkce nazývaná BRDF (dvousměrová distribuční funkce odrazu, anglicky Bidirectional Reflectance Distribution Function).

V následujících kapitolách jsou popsány základní používané veličiny, definováno BRDF a jeho vlastnosti, představeny některé existující analytické modely a formáty pro ukládání naměřených dat.

Vlastní práce pak spočívá v měření odrazivosti materiálů pomocí přístroje MiniDiff v2 od společnosti Synopsys, ve naprogramování softwaru na zpracování naměřených dat a vytvoření shaderů pro herní enginey Unity a Unreal Engine.

## 1.1 Základní radiometrické veličiny

Radiometrie je disciplína fyziky zabývající se měřením elektromagnetického záření. Poskytuje nástroje a definuje veličiny popisující šíření a odražení světla. Pro potřeby počítačové grafiky se uvažuje světlo jako částice pro tři vlnové délky odpovídající RGB.

Pro popis odrazivosti je nutné nejdříve popsat základní používané veličiny: zářivý tok, zářivost, intenzita vyzařování a zář.

### 1.1.1 Zářivý tok

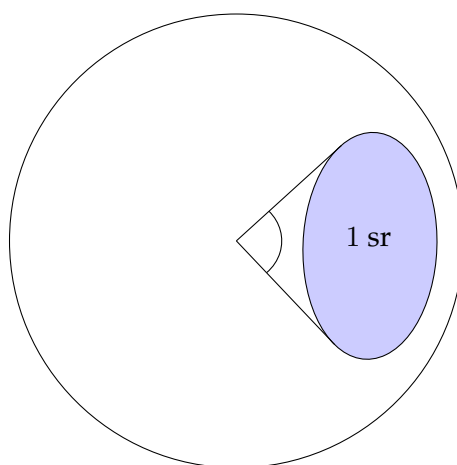
Zářivý tok ( $\Phi [W]$ ) určuje množství energie procházející danou plochou za jednotku času.

### 1.1.2 Zářivost

Zářivost ( $I [\frac{W}{sr}]$ ) popisuje zářivý tok vztažený k prostorovému úhlu  $\omega$ :

$$I = \frac{d\Phi}{d\omega}. \quad (1)$$

Pro prostorový úhel se používá jednotka steradián ( $sr$ ):  $1 sr$  reprezentuje úhel vymezený jednotkovou plochou na jednotkové sféře. Pro názornost: celá sféra představuje  $4\pi$  steradiánů a hemisféra  $2\pi$  steradiánů.



Obrázek 1: Příklad prostorového úhlu  $1 sr$

### 1.1.3 Intenzita vyzařování

Intenzita vyzařování ( $E$  [ $\frac{W}{m^2}$ ]) popisuje zářivý tok vztažený k ploše  $A$  – množství energie procházející jednotkovou plochou:

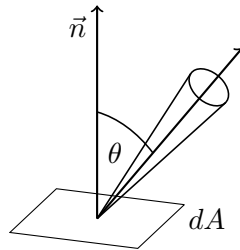
$$E = \frac{d\Phi}{dA}. \quad (2)$$

### 1.1.4 Zář

Zář ( $L$  [ $\frac{W}{m^2 \cdot sr}$ ]) kombinuje předchozí dvě veličiny a je definována jako zářivý tok vztažený k ploše a prostorovému úhlu:

$$L = \frac{d^2\Phi}{dA \cdot |\cos \theta| d\omega}, \quad (3)$$

kde  $dA \cdot |\cos \theta|$  vyjadřuje projekci plochy  $dA$  na rovinu kolmou k směru prostorovému úhlu a  $\theta$  je úhel mezi směrem prostorového úhlu a normálou roviny, ve které leží plocha  $dA$ .



Obrázek 2: Příklad záře

## 1.2 BRDF

Dvousměrová distribuční funkce odrazu (anglicky Bidirectional Reflectance Distribution Function, zkratka BRDF) se využívá k matematické reprezentaci chování materiálů vzhledem k odrazu dopadajícího světla. Je definována jako poměr odchozí záře ve směru  $\vec{\omega}_{out}$  a intenzity záření přicházejícího ze směru  $\vec{\omega}_{in}$  a vlnové délce  $\lambda$ :<sup>[1]</sup>

$$f(\vec{\omega}_{in}, \vec{\omega}_{out}, \lambda) = \frac{dL_{out}(\vec{\omega}_{out}, \lambda)}{dE(\vec{\omega}_{in}, \lambda)}. \quad (4)$$

Směr dopadu  $\vec{\omega}_{in}$  a směr odrazu  $\vec{\omega}_{out}$  lze reprezentovat buď pomocí sférických souřadnic  $\theta, \phi, r$  (zde vždy s pevnou vzdáleností  $r = 1$ ), a nebo pomocí kartézských souřadnic  $x, y, z$ .

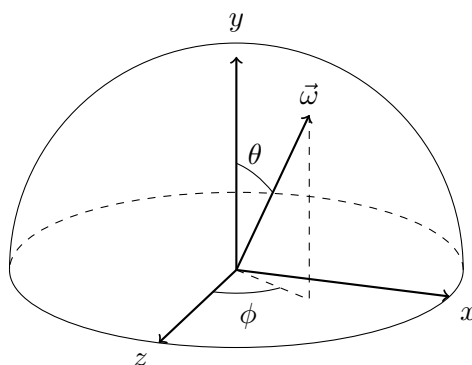
Kartézský souřadnicový systém používaný v rámci této práce (pokud není řečeno jinak) je shodný se souřadnicovým systémem používaným v OpenGL. Jedná se o pravotočivý souřadnicový systém s osou  $y$  směřující vzhůru (viz obrázek 3).

Sférický souřadnicový systém popisuje pozici bodu v prostoru pomocí umístění daného bodu určeného dvojicí úhlů  $\theta$  a  $\phi$  na sféru o poloměru  $r$ . Bod  $\vec{v}$  a osa  $y$  svírají úhel  $\theta$ , úhel  $\phi$  se nachází mezi osou  $z$  a projekcí  $\vec{v}$  na rovinu  $xz$ . Převod mezi sférickými a kartézskými souřadnicemi lze provádět pomocí následujícího vzorce:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} r \cdot \sin \theta \cdot \sin \phi \\ r \cdot \cos \theta \\ r \cdot \sin \theta \cdot \cos \phi \end{bmatrix}, \quad \begin{bmatrix} \theta \\ \phi \\ r \end{bmatrix} = \begin{bmatrix} \arccos\left(\frac{y}{\sqrt{x^2 + y^2 + z^2}}\right) \\ \arctan2(x, z) \\ \sqrt{x^2 + y^2 + z^2} \end{bmatrix}, \quad (5)$$

kde funkce  $\arctan2(b, a)$  je zobecněná varianta výrazu  $\arctan(\frac{a}{b})$  s definičním oborem  $[0, 2\pi)$ .

V následujících kapitolách jsou využívány dle situace oba souřadné systémy pro popis směrů dopadu a odrazu. Je tak učiněno s cílem čitelnějšího zápisu některých vzorců.



Obrázek 3: Příklad kartézských a sférických souřadnic

### 1.2.1 Směrově-hemisférická odrazivost (albedo)

Směrově-hemisférická odrazivost reprezentuje celkový poměr odraženého světla přicházejícího z daného směru do všech směrů na hemisféře:<sup>[1]</sup>

$$a(\vec{\omega}_{in}, \lambda) = \int_{\Omega} f(\vec{\omega}_{in}, \vec{\omega}_{out}, \lambda) \cdot \cos \theta_{out} d\vec{\omega}_{out}. \quad (6)$$

### 1.2.2 Dvou-hemisférická odrazivost (průměrné albedo)

Průměrné albedo je pohledově nezávislá varianta albeda. Je vyjádřena jako vážený průměr albeda přes všechny směry úhlu dopadu  $\theta_{in}$ :<sup>[1]</sup>

$$a_{mean}(\lambda) = \frac{1}{\pi} \int_{\Omega} a(\vec{\omega}_{in}, \lambda) \cdot \cos \omega_{in} d\vec{\omega}_{in}. \quad (7)$$

### 1.2.3 Symetrie

Jeden ze základních aspektů BRDF vycházející z fyzikálních vlastností světla popsaného Helmholtzovou reciprocitou je symetrie vůči směru dopadu a směru odrazu, tzn. je možné zaměnit mezi sebou zdroj světla a pozorovatele:<sup>[2]</sup>

$$\forall \vec{\omega}_{in}, \vec{\omega}_{out} \in \Omega : f(\vec{\omega}_{in}, \vec{\omega}_{out}, \lambda) = f(\vec{\omega}_{out}, \vec{\omega}_{in}, \lambda). \quad (8)$$

Některé existující analytické modely toto pravidlo nespĺňují (viz kapitola 2).

### 1.2.4 Zachování energie

Dalším z fyzikálních zákonů vztahující se k chování odrazivosti je zákon o zachování energie. Z něho vyplývá, že množství odražené energie nesmí přesáhnout množství dopadající energie záření.<sup>[2]</sup> Pomocí albeda lze tuto vlastnost vyjádřit jako:

$$\forall \vec{\omega}_{in} \in \Omega : a(\vec{\omega}_{in}, \lambda) \leq 1. \quad (9)$$

Stejně jako v případě symetrie, některé existující analytické modely tuto vlastnost nespĺňují a je nutné tento problém kompenzovat normalizační konstantou.

### 1.2.5 Izotropie

Dle chování materiálu vzhledem k úhlu  $\phi_{in}$  je možné je rozdělit do dvou kategorií: izotropní a anizotropní. Izotropní materiály respektují rotační symetrii, tj. nemění svojí odrazivost v závislosti na natočení vzorku podle osy definované normálou povrchu.<sup>[2]</sup> Tudíž splňují následující rovnost:

$$\begin{aligned} \forall \vec{\omega}_{in} := (\phi_{in}, \theta_{in}), \vec{\omega}_{out} := (\phi_{out}, \theta_{out}) : \\ f((\phi_{in}, \theta_{in}), (\phi_{out}, \theta_{out}), \lambda) = f((0, \theta_{in}), (\phi_{out} - \phi_{in}, \theta_{out}), \lambda). \end{aligned} \quad (10)$$

Z této rovnosti lze odvodit, že izotropní materiály mají pouze tři stupně volnosti (při dané vlnové délce), a proto klesá paměťová i časová náročnost měření.

Materiály, které nejsou izotropní nazýváme anizotropní. Nesplňují výše uvedenou rovnost a mají čtyři stupně volnosti. Většina přírodním materiálů je anizotropní.

### 1.2.6 ARS

ARS (z anglického Angular Resolved Scatter) je nepříliš používaná varianta BRDF kompenzovaná cosinem úhlu od normály:<sup>[3]</sup>

$$f_{ars}(\vec{\omega}_{in}, \vec{\omega}_{out}, \lambda) = f_{brdf}(\vec{\omega}_{in}, \vec{\omega}_{out}, \lambda) \cdot \cos \theta_{out}. \quad (11)$$

### 1.2.7 BTDF

Dvousměrová distribuční funkce lomu (anglicky Bidirectional Transmittance Distribution Function) je matematická reprezentace chování materiálu vzhledem k průsvitnosti světla přes materiál. Je definována stejným způsobem jako BRDF, liší se pouze povoleným rozsahem úhlu  $\theta_{out} \in [90^\circ, 180^\circ]$ .

## 1.3 Zobrazovací rovnice

Zobrazovací rovnice vyjadřuje výslednou zář odraženou do daného směru  $\vec{\omega}_{out}$ :<sup>[2]</sup>

$$L_{out}(\vec{\omega}_{out}, \lambda) = L_{emit}(\vec{\omega}_{out}, \lambda) + \int_{\Omega} f(\vec{\omega}_{in}, \vec{\omega}_{out}, \lambda) \cdot L_{in}(\vec{\omega}_{in}, \lambda) \cdot \cos \theta \, d\vec{\omega}_{in}. \quad (12)$$

Cílem globálních osvětlovacích modelů používaných v počítačové grafice je snaha o co nejuvěrnější aproximaci této rovnice. Z důvodů výskytu záře na obou stranách rovnice a nutnosti integrace přes hemisféru je přesné vyjádření příliš výpočetně náročné. Proto se používají techniky založené na Monte-Carlo numerické integraci<sup>[2]</sup>, popřípadě různé způsoby nahrazení části rovnice předpřipravenými daty a aproximací BRDF<sup>[4]</sup>.

## 2 Analytické modely BRDF

Jeden z možných způsobů vyjádření BRDF je pomocí explicitně definovaného analytického vzorce. Jednou z výhod toho přístupu je paměťová nenáročnost, protože jsou ovlivňovány parametry pouze v řádu jednotek. Naopak nevýhodou se stává menší variabilita modelu, pomocí kterého se dá reprezentovat pouze určitá skupina materiálů.<sup>[5]</sup>

Analytické modely lze rozdělit do dvou hlavních skupin: empirické a fyzikálně založené. Empirické modely vznikly za pomoci dat z reálného světa a následného hledání analytického vzorce aproximující tyto data. Fyzikálně založené modely využívají primárně znalostí z oblasti fyziky a optiky k formulaci daného vzorce.<sup>[2]</sup>

### 2.1 Ideální modely

Níže uvedené teoretické modely v této kapitole je vhodné zmínit zvlášť, vzhledem k tomu, že se jedná o základní modely, na které je často odkazováno při definici složitějších.

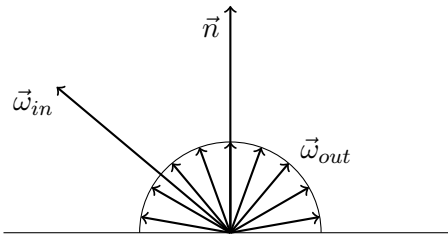
#### 2.1.1 Lambert

Lambertův model je ideální reprezentací difúzní odrazivosti, při kterém je příchozí světlo odraženo do všech směrů rovnoměrně. Je nezávislý na úhlu dopadu i odrazu paprsku a hodnota funkce je definována pomocí konstanty  $\rho_d$  jako:<sup>[2]</sup>

$$f(\vec{\omega}_{in}, \vec{\omega}_{out}, \lambda) = \frac{\rho_d}{\pi}, \quad (13)$$

kde  $\pi$  je normalizační konstanta zajišťující zachování energie a platnost vztahu  $\rho_d = a(\vec{\omega}_{in}, \lambda) = a_{mean}(\lambda)$ .

Z výše uvedeného vztahu je zřejmé, že se jedná o symetrický, izotropní model zachovávající energii (pro  $\rho_d \leq 1$ ).



Obrázek 4: Odrazivost Lambertova modelu

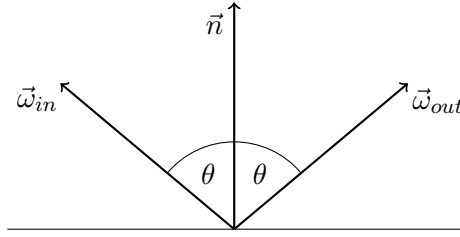
#### 2.1.2 Ideální odraz

Model ideálního odrazu odpovídá fungování zákona o odrazu. Přicházející paprsek je odražen pouze do jediného směru odpovídající zrcadlovému odrazu  $\vec{\omega}_{out} = r(\vec{\omega}_{in})$ . Zrcadlový odraz lze definovat v prostoru sférických souřadnic jako  $r(\phi, \theta) = (\phi + \pi, \theta)$ , popřípadě v kartézských souřadnicích jako  $r(\vec{x}) = \vec{x} - 2 \cdot (\vec{x} \cdot \vec{n}) \cdot \vec{n}$ , kde  $\vec{n}$  udává normálu povrchu.

BRDF ideálního odrazu je definováno pomocí funkce  $\rho_s(\vec{\omega}_{in})$  určující míru odrazivosti pro daný příchozí směr vynásobené funkcí nazývanou Diracovo delta:<sup>[2]</sup>

$$f(\vec{\omega}_{in}, \vec{\omega}_{out}, \lambda) = \rho_s(\vec{\omega}_{in}) \cdot \delta(\vec{\omega}_{out}, r(\vec{\omega}_{in})). \quad (14)$$

Stejně jako u Lambertova modelu je model ideálního odrazu symetrický, izotropní a respektuje zachování energie.



Obrázek 5: Ideální odraz

## 2.2 Empirické modely

### 2.2.1 Phong

Jeden z prvních analytických modelů popisující odrazivost povrchů, které neodpovídají chování Lambertova modelu, je Phongův model. Phong se ve své práci zabýval výpočtem výsledné záře osvětleného povrchu, kterou definoval jako:<sup>[6]</sup>

$$L_{out} = k_d \cdot (\cos(\theta_{in}) \cdot (1 - d) + d) + k_s \cdot (\vec{\omega}_{out} \cdot r(\vec{\omega}_{in}))^n, \quad (15)$$

kde  $k_d$  a  $k_s$  jsou koeficienty odrazivosti difúzní a spekulární složky,  $d$  vyjadřuje koeficient rovnoměrného zdroje světla z okolního prostředí a  $n \in [0, \infty)$  určuje matnost materiálu.

Matematickými úpravami lze tento vzorec rozložit na konstantní zář z okolí a BRDF skládající se z difúzní složky, odpovídající až na konstantu Lambertově modelu, a spekulární složky, využívající cosinus úhlu mezi směrem odrazu a zrcadlovým odrazem:

$$f(\vec{\omega}_{in}, \vec{\omega}_{out}, \lambda) = k_d \cdot (1 - d) + k_s \cdot \frac{\vec{\omega}_{out} \cdot r(\vec{\omega}_{in})^n}{\cos(\theta_{in})}. \quad (16)$$

Hlavní nevýhodou tohoto modelu je nerespektování zákona o zachování energie a nesplnění předpokladu symetrie.<sup>[5]</sup>

### 2.2.2 Lafortune-Phong

Lafortune se zaměřil na problematické aspekty Phongova modelu a přišel s upravenou variantou. Spekulární složka BRDF obsahuje normalizační konstantu zajišťující zachování energie, společně s odstraněním koeficientu převrácené hodnoty cosinem úhlu dopadu:<sup>[7]</sup>

$$f(\vec{\omega}_{in}, \vec{\omega}_{out}, \lambda) = \frac{k_d}{\pi} + k_s \cdot \frac{n + 2}{2\pi} \cdot (\max\{\vec{\omega}_{out} \cdot r(\vec{\omega}_{in}), 0\})^n. \quad (17)$$

Díky těmto změnám tento model splňuje zachování energie i předpoklad symetrie.<sup>[7]</sup>

### 2.2.3 Ward

Wardův model byl vytvořen s důrazem na jednoduchost nalezení parametrů z naměřených dat. Skládá se z difúzní a spekulární složky. Difúzní složka odpovídá Lambertově modelu. Spekulární složka je založena na Gaussově (normálním) rozdělení s parametry určující standardní odchylku  $\alpha$  a míru intenzity  $k_s$ .<sup>[8]</sup>

$$f_s(\vec{\omega}_{in}, \vec{\omega}_{out}, \lambda) = \frac{k_s}{\sqrt{\cos(\vec{\theta}_{in}) \cos(\vec{\theta}_{out})}} \frac{e^{-\tan^2(\frac{\vec{h} \cdot \vec{n}}{\alpha^2})}}{4\pi\alpha^2}, \quad (18)$$

kde  $\vec{h}$  vyjadřuje vektor mezi směrem dopadu a směrem odrazu a je definován jako  $\vec{h} = \|\vec{v}_{in} + \vec{v}_{out}\|$ . Koeficient  $4\pi\alpha^2$  normalizuje hodnoty Gaussova rozdělení, přesto výsledný model nezaručuje zachování energie.



Pro anizotropní materiály existuje rozšířená varianta, která využívá eliptického Gaussova rozdělení určeného dvěma koeficienty  $\alpha_x, \alpha_y$  pro vyjádření standardní odchylky ve dvou kolmých směrech ( $\vec{x}$  a  $\vec{y}$  v tečném prostoru):<sup>[8]</sup>

$$f_s(\vec{\omega}_{in}, \vec{\omega}_{out}, \lambda) = \frac{k_s}{\sqrt{\cos(\vec{\theta}_{in}) \cos(\vec{\theta}_{out})}} e^{-\frac{(\frac{\vec{h} \cdot \vec{x}}{\alpha_x})^2 + (\frac{\vec{h} \cdot \vec{y}}{\alpha_y})^2}{(\vec{h} \cdot \vec{n})^2}}, \quad (19)$$

Wardův model respektuje symetrii, ale jak už bylo zmíněno výše, nespĺňuje zachování energie. Následně byly objeveny alternativní normalizační koeficienty, které řeší problém zachování energie.<sup>[2]</sup>

## 2.3 Fyzikálně založené modely

### 2.3.1 Fresnelovy rovnice a Schlickova aproximace

Fresnelovy rovnice popisují, jak se mění poměr odraženého a lomeného světla na rozhraní mezi dvěma prostředími s různými indexy lomu v závislosti na úhlu dopadu. Chování pro nepolarizované světlo dopadající na rozhraní materiálu s indexem lomu  $\eta$  odpovídá následujícímu vzorci:<sup>[9]</sup>

$$F = \frac{(g - c)^2}{2(g + c)^2} \cdot \left( 1 + \frac{(c \cdot (g + c) - 1)^2}{(c \cdot (g - c) + 1)^2} \right), \quad (20)$$

$$c = \vec{\omega}_{out} \cdot \vec{h}, \quad (21)$$

$$g = \sqrt{\eta^2 + c^2 - 1}. \quad (22)$$

Vzhledem k tomu, že se jedná o výpočetně náročný vzorec, Schlick ve své práci popisuje aproximovanou variantu:<sup>[10]</sup>

$$F = F_0 + (1 - F_0)(1 - \cos \theta_d)^5, \quad (23)$$

kde  $\theta_d$  je úhel mezi směrem dopadu a vektorem  $\vec{h}$ , konstanta  $F_0$  určuje sílu Fresnelova efektu pro směr odpovídající normále povrchu.

### 2.3.2 Cook-Torrance

Cook-Torranceův model, stejně jako mnohé jiné, je rozdělen na difúzní a spekulární složku. Pro jednotlivé složky definuje váhový parametr  $k_d$  a  $k_s$ :<sup>[9]</sup>

$$f(\vec{\omega}_{in}, \vec{\omega}_{out}, \lambda) = k_d \cdot f_d(\vec{\omega}_{in}, \vec{\omega}_{out}, \lambda) + k_s \cdot f_s(\vec{\omega}_{in}, \vec{\omega}_{out}, \lambda) \quad (24)$$

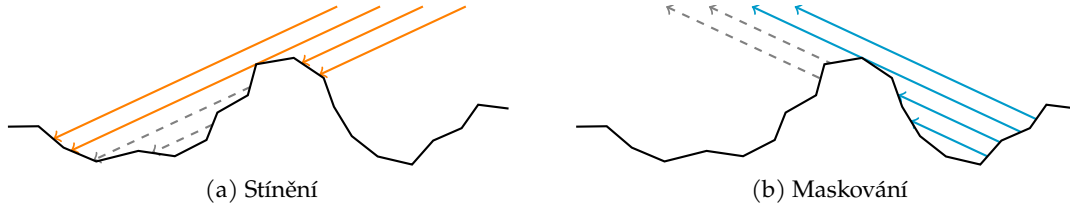
a platí  $k_d + k_s = 1$ . Pro difúzní složku je využit Lambertův model.

Spekulární složka je založena na teorii micro-plošek, tj. povrch materiálu se skládá z mnoha malých, náhodně natočených plošek, kde každá z nich se chová jako zrcadlo. Vzhledem k tomu chování lze předpokládat, že se plošky odrážejí pouze v případě, že platí  $\vec{\omega}_{out} = r(\vec{\omega}_{in})$ , resp.  $\vec{h} = \vec{n}$ . Vzorec je rozdělen na součin třech částí:  $F$  určuje Fresnelovu rovnici popisující odrazivost každé z plošek,  $G$  určuje útlum způsobený geometrií povrchu a  $D$  distribuční funkce pravděpodobnosti orientace plošek:<sup>[9]</sup>

$$f_s(\vec{\omega}_{in}, \vec{\omega}_{out}, \lambda) = \frac{F}{\pi} \frac{G \cdot D}{(\vec{n} \cdot \vec{\omega}_{in})(\vec{n} \cdot \vec{\omega}_{out})}. \quad (25)$$

Geometrický útlum  $G$  je způsobem dvěma rozdílnými zdroji: stíněním a maskováním. Stínění je způsobeno zakrytím přicházejícího paprsku jinými ploškami materiálu a maskování je způsobeno zakrytím odraženého paprsku. Vzorec používaný v tomto modelu je:<sup>[9]</sup>

$$G = \min \left\{ 1, \frac{2(\vec{n} \cdot \vec{h})(\vec{n} \cdot \vec{\omega}_{out})}{\vec{\omega}_{out} \cdot \vec{h}}, \frac{2(\vec{n} \cdot \vec{h})(\vec{n} \cdot \vec{\omega}_{in})}{\vec{\omega}_{out} \cdot \vec{h}} \right\}. \quad (26)$$



Obrázek 6: Příklad geometrického útlumu

Pro distribuční funkci  $D$  existuje více variant. Jedna z možných je Beckmannova distribuční funkce:<sup>[9]</sup>

$$D = \frac{1}{m^2 \cdot \cos^4(\vec{h} \cdot \vec{n})} \cdot e^{-\left(\frac{\tan(\vec{h} \cdot \vec{n})}{m}\right)^2}, \quad (27)$$

kde  $m$  je parametr určující hrubost materiálu.

Cook-Torrancův model je určen pro izotropní materiály a respektuje symetrii, ale nesplňuje zachování energie.<sup>[5]</sup>

### 2.3.3 Oren-Nayar

Oren a Nayar se ve svém výzkumu zaměřili na difúzní model pro hrubé materiály. Tento model je také založen na teorii mikro-plošek. Skládá se z náhodně orientovaných rýh ve tvaru písmene V. Mikro-plošky orientované směrem přicházejícího paprsku část odrazí zpět. Kromě stínění a maskování tento model bere v úvahu odrazy mezi jednotlivými ploškami. Výsledný používaný vzorec je pouze aproximací kvůli menší výpočetní náročnosti:<sup>[11]</sup>

$$f(\vec{\omega}_{in}, \vec{\omega}_{out}, \lambda) = \frac{\rho_d}{\pi} (A + B \cdot \max(0, \cos(\phi_{in} - \phi_{out})) \cdot \sin(\max(\theta_{in}, \theta_{out})) \cdot \tan(\min(\theta_{in}, \theta_{out}))), \quad (28)$$

$$A = 1 - \frac{0.5 \cdot \sigma^2}{\sigma^2 + 0.33}, \quad (29)$$

$$B = \frac{0.45 \cdot \sigma^2}{\sigma^2 + 0.09}, \quad (30)$$

kde  $\sigma$  vyjadřuje matnost materiálu. Pro hodnotu  $\sigma = 0$  odpovídá Lambertově modelu.

Oren-Nayarův model je izotropní, respektuje symetrii a zachování energie.<sup>[5]</sup>

### 2.3.4 Walter (GGX)

Walterův model počítá kromě odrazivosti i lomivost světla. Využívá podobné rozdělení spekulární složky na  $F$ ,  $G$ , a  $D$  jako u Cook-Torrancova modelu, ale obsahuje jiný normalizační faktor (4 místo  $\pi$ ):<sup>[12]</sup>

$$f_s(\vec{\omega}_{in}, \vec{\omega}_{out}, \lambda) = \frac{F}{4} \frac{G \cdot D}{(\vec{n} \cdot \vec{\omega}_{in})(\vec{n} \cdot \vec{\omega}_{out})}. \quad (31)$$

Fresnelův vzorec zůstává stejný, jak byl definován v Cook-Torrancově modelu. Naopak pro člen  $D$  vytvořili novou distribuční funkci pojmenovanou jako GGX:<sup>[12]</sup>

$$D = \frac{\alpha^2 \cdot \max(0, \vec{h} \cdot \vec{n})}{\pi \cdot \cos^4 \theta_h (\alpha^2 + \tan^2 \theta_h)^2}, \quad (32)$$

kde  $\theta_h$  vyjadřuje úhel mezi normálou a vektorem  $\vec{h}$  ( $\theta_h = \arccos(\vec{n} \cdot \vec{h})$ ). Parametr  $\alpha$  určuje šířku spekulárního odrazu.

Na rozdíl od geometrické funkce  $G$  použité v Cook-Torrancova modelu, která byla nezávislá na použité distribuční funkci, v tomto modelu dochází k aproximaci stínění a maskování pomocí rozdělení na dva podproblémy:<sup>[12]</sup>

$$G(\vec{\omega}_{in}, \vec{\omega}_{out}) \approx G_1(\vec{\omega}_{in}) \cdot G_1(\vec{\omega}_{out}), \quad (33)$$

$$G_1(\vec{\omega}) = \max\left(0, \frac{\vec{\omega} \cdot \vec{h}}{\vec{\omega} \cdot \vec{n}}\right) \cdot \frac{2}{1 + \sqrt{1 + \alpha^2 \cdot \tan^2 \theta}}. \quad (34)$$

Walterův model je izotropní, respektuje symetrii a zachování energie.<sup>[5]</sup>

### 2.3.5 Disney

Hlavním cílem u modelu vytvořeného společností Disney je intuitivnost parametrů a požadavek, aby zároveň odpovídal naměřeným datům z databáze MERL<sup>[13]</sup>. Vychází z Walterova modelu, ale přináší několik změn. Všechny vstupní parametry vyjadřuje pro rozsah  $[0, 1]$ . Pro Fresnelovu rovnici  $F$  je použita Schlickova aproximace.<sup>[14]</sup>

$G$  je shodná s Walterovým modelem, ale parametr  $\alpha$  je omezen na rozsah  $[0.5, 1]$  jako  $\alpha = (\frac{1+\alpha}{2})^2$ . Pro distribuční funkci mikro-plošek  $D$  se používají dvě varianty: pro základní povrch nezměněná varianta Walterova modelu, pro vrstvu povrchového lesku je použita generalizovaná varianta bez kvadratického exponentu.<sup>[14]</sup>

Další velkou změnou je difúzní složka. Původní Lambertův model nepočítá s Fresnelovým efektem, a proto je zde model vylepšen o jeho zohlednění. Stejně jako v případě spekulární složky je použita Schlickova aproximace:<sup>[14]</sup>

$$f_d(\vec{\omega}_{in}, \vec{\omega}_{out}, \lambda) = \frac{\rho_d}{\pi} \cdot (1 + (F_{D90} - 1)(1 - \cos \theta_{in})^5) \cdot (1 + (F_{D90} - 1)(1 - \cos \theta_{out})^5), \quad (35)$$

$$F_{D90} = 0.5 + 2\alpha \cdot \cos^2 \theta_d. \quad (36)$$

Fresnelův efekt je zde zohledněn dvakrát: jednou pro směr dopadu a jednou pro směr odrazu.



### 3 Formáty pro ukládání BRDF

Naměřená data odrazivosti vzorků je nutné ukládat pro další zpracování a využití. V současné době bohužel různá měřicí zařízení a softwary používají různé vlastní souborové formáty. Často k těmto formátům neexistuje specifikace popisující jejich obsah a strukturu.

U každého z formátů lze rozlišovat několik aspektů:

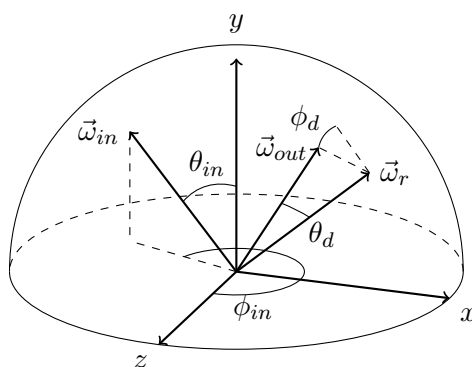
- univerzálnost – zda je možné uložit různé typy naměřených dat (BRDF, BTDF) a podpora pro ukládání dodatečných parametrů (vlnové délky, polarizace, apod.),
- typ ukládání – zda se jedná o textový nebo binární soubor,
- podpora metadat – zda je možné kromě samotných naměřených dat ukládat i informace o měření (typ vzorku, konfigurace měřícího přístroje, autora apod.),
- velikost souboru – kolik výsledný soubor zabírá úložiště vzhledem k ostatním možným variantám.

V následujících podkapitolách jsou popsány tři existující souborové formáty. Formát společnosti Zemax byl vybrán kvůli vlastnímu měření prováděnému v rámci této práce, formát používaný v databázi MERL je zde zmíněn kvůli její známosti a formát BiRD kvůli tomu, že byl vytvořen s cílem vytvořit univerzální formát pro tento typ dat.

#### 3.1 Zemax

Společnost Zemax pro svůj ekosystém produktů vytvořila textový formát pro ukládání tabularizovaných dat. Nabízí zápis buď hodnot BRDF, nebo BTDF. Dále umožňuje respektování symetrie podle osy  $yz$ , ukládání dat izotropních i anizotropních materiálů a rovněž disponuje podporou ukládání dat pro jednu nespecifikovanou vlnovou délku, nebo pro tři vlnové délky dle standardu CIE XYZ.<sup>[15]</sup>

Jako souřadnicový systém používá sférické souřadnice definované vůči dvěma různým prostorům. Pro příchozí paprsky využívá tečný prostor. Pro odchozí paprsky se jedná o prostor, ve kterém osa  $z$  odpovídá zrcadlovému odrazu pro BRDF a směru přímému průchodu (bez lomu) materiálem pro BTDF. Osa  $x$  směřuje k normále povrchu (resp. její záporné hodnotě pro BTDF) – jedná se o normalizovanou projekci normály do roviny definované osou  $z$ .<sup>[15]</sup>



Obrázek 7: Příklad souřadnicového systému formátu firmy Zemax

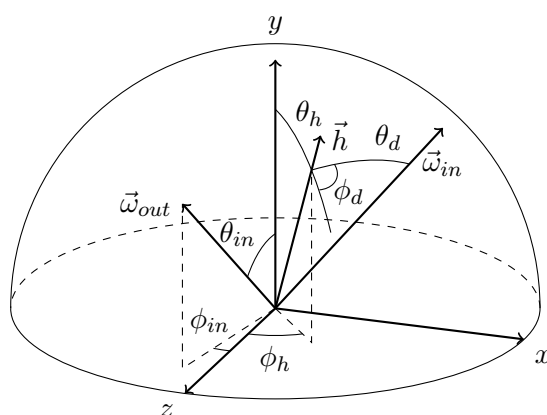
Samotná data jsou uložena jako uspořádaná množina matic. Každá z matic reprezentuje hodnoty odrazivosti pro daný směr příchozích paprsků. Hodnoty pro odchozí směr mimo definovaný poloprostor (tj. pod vzorkem pro BRDF, nad vzorkem pro BTDF) jsou neplatné a musí být ignorovány.<sup>[15]</sup>

Z důvodů využití sférických souřadnic dochází k částečné duplicitě dat (pro  $\theta = 0$ ) jak v jednotlivých prvcích, tak i pro celé matice v případě anizotropních dat. Přesto přináší výhody v podobě vyšší hustoty naměřených dat pro oblast okolo zrcadlového odrazu. Hlavní nevýhodou tohoto formátu je absence podpory metadat, které lze zapsat pouze nestrukturovaně pomocí komentářů. Velikost souboru není nezanedbatelná kvůli textové podobě, vzhledem k ostatním textovým formátům se ale jedná spíše o úspornější reprezentaci.

### 3.2 MERL databáze

Pro účely uložení dat v rámci svého výzkumu Matusik s kolektivem vytvořili binární formát pro izotropní materiály. Jedná se o omezený formát neumožňující přidání metadat a je definován pro trojici barev RGB. Výhodou je nízká paměťová náročnost a jednoduché načítání souboru.<sup>[13]</sup>

Tento formát používá Rusinkiewiczův souřadnicový systém. Ten se skládá celkem ze čtyř souřadnic:  $\theta_h$ ,  $\theta_d$ ,  $\phi_h$  a  $\phi_d$ .  $\theta_h$  a  $\phi_h$  jsou sférické souřadnice vektoru  $\vec{h}$ ,  $\theta_d$  a  $\phi_d$  jsou sférické souřadnice příchozího směru  $\vec{\omega}_{in}$  v bázi, kde  $\vec{h}$  reprezentuje osu  $z$ .<sup>[13]</sup>



Obrázek 8: Příklad Rusinkiewiczova souřadnicového systému

Data jsou uložena v trojrozměrném poli.  $\phi_h$  je ignorováno, protože se předpokládá izotropie vzorku. Pro výpočet indexu  $\theta_h$  hodnoty v poli se používá nelineární mapování  $i_{\theta_h} = n \cdot \sqrt{2\theta_h/\pi}$ . U  $\phi_d$  se předpokládá symetrie měřených dat a proto může nabývat hodnot  $[0, \pi]$ .<sup>[13]</sup>

Stejně jako v případě formátu společnosti Zemax, i zde dochází k stejným aspektům při použití sférických souřadnic, ale zde použité nelineární mapování ještě zvyšuje hustotu dat okolo zrcadlového odrazu. Velikost souboru je výrazně menší díky binárnímu zápisu a úzce specifickému účelu formátu.

### 3.3 BiRD JSON

Jako součást projektu BiRD byl vytvořen univerzální formát pro ukládání BRDF založený na formátu JSON. Data jsou ukládána pomocí uspořádaného pole naměřených hodnot a dodatečných polí stejné délky určující parametry pro dané naměřené hodnoty. Konkrétně se jedná o směr, vlnovou délku a polarizaci dopadajícího a odraženého světla, ale je možné zavést i vlastní parametry. Díky tomuto způsobu je možné ukládat nezpracovaná naměřená data bez nutnosti jejich tabularizace.<sup>[16]</sup>

Dále podporuje ukládání velké škály metadat ohledně autora, času a místa pořízení, využití vybavení a softwaru, podmínek měření, informací o vzorku a licence k naměřeným datům. Mezi nevýhody patří absence informace ohledně izotropie a symetrie vzorku a značná velikost souboru, která je způsobena univerzálností a použitím textového formátu.

## 4 MiniDiff v2

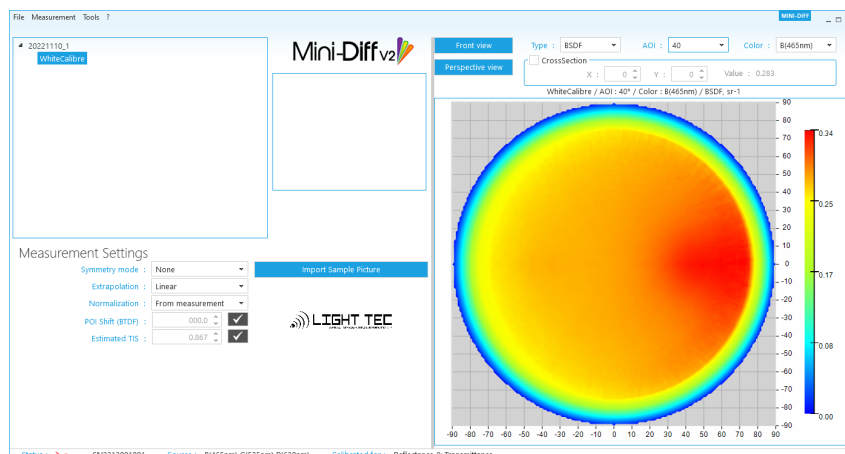
Pro skenování odrazivosti materiálů byl použit přístroj MiniDiff v2 od společnosti Synopsys (dříve Lightec). Jedná se o kontaktní scatterometer umožňující měřit odrazivost a průsvitnost vzorku. Pro měření odrazivosti používá jako zdroj světla čtyři RGB LED v úhlech  $0^\circ$ ,  $20^\circ$ ,  $40^\circ$  a  $60^\circ$  od normály povrchu vzorku. Tyto LED mají definovanou vlnovou délku 630 nm pro červenou, 525 nm pro zelenou a 465 nm pro modrou barvu. V každém z úhlů dopadu měří odrazivost na kulové výšeči v rozsahu  $\phi \in [0^\circ, 360^\circ)$  a  $\theta \in [0^\circ, 75^\circ]$  s rozlišením  $1^\circ$ . Na měření průsvitnosti vzorku je potřeba připojit dodatečný modul, ve kterém se nachází další čtveřice RGB LED pokrývající úhly dopadu  $120^\circ$ ,  $140^\circ$ ,  $160^\circ$  a  $180^\circ$ . Kvůli dané konstrukci zařízení je možné skenovat pouze izotropní materiály.<sup>[3]</sup>



Obrázek 9: Přístroj MiniDiff v2

### 4.1 Obslužný software

Softwarová část přístroje zajišťuje kromě samotného skenování a kalibrace také správu relací, vizualizaci, nastavení extrapolace a rovinné symetrie, průměrování, ukládání a export naměřených dat.



Obrázek 10: Snímek okna softwaru Minidiff v2

#### 4.1.1 Kalibrace

Před zahájením každé relace je nutné provést kalibraci pomocí přiložených standardizovaných vzorků. Pro odrazivost se jedná o vzorek bílý (průměrné albedo blízké 1, odrazivost velice podobná Lambertovu modelu) a černý (průměrné albedo blízké 0). Pro průsvitnost se jedná o vzorek průsvitný.

Kalibraci není možné žádným způsobem uložit a znovu použít při opětovném spuštění obslužného programu. Veškeré měřené vzorky provedené v určité relaci mají k sobě pevně vztahenou konkrétní kalibraci a není možné na nezpracovaná naměřená data aplikovat kalibraci z jiného měření.

#### 4.1.2 Vizualizace

Naměřená data jsou zobrazena pomocí 2D kartézské mřížky nebo 3D pohledu. Pro kartézské souřadnice platí vzorec:

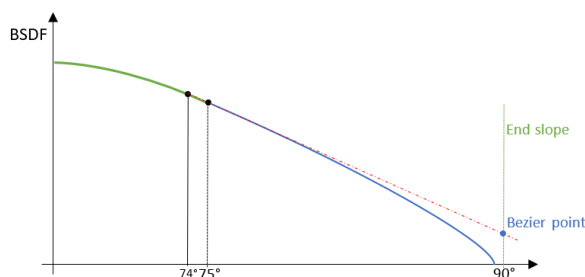
$$\begin{aligned}x &= \theta \cdot \cos \phi, \\y &= \theta \cdot \sin \phi, \\z &= f(\omega_{in}, \vec{\omega}_{out}, \lambda).\end{aligned}$$

Při zobrazování lze zvolit mezi nepravými barvami určující naměřené BRDF nebo ARS a barvami CIE 1931, 1976 a RGB.

#### 4.1.3 Extrapolace a symetrie

Vzhledem k situaci, že přístroj umožňuje měřit pouze pro úhly dopadu a odrazu  $\theta \leq 75^\circ$ , pro zbývající kulovou vrstvu ( $\theta \in (75^\circ, 90^\circ]$ ) je nutné data extrapolovat. Součástí obslužného softwaru přístroje MiniDiff v2 jsou následující možnosti extrapolace:

1. bez extrapolace – data mimo dostupný rozsah jsou rovna 0,
2. lineární – data jsou lineárně interpolovaná mezi poslední dostupnou naměřenou hodnotou pro  $\theta = 75^\circ$  a 0 pro  $90^\circ$ ,
3. konstantní – data mimo rozsah jsou rovny nejbližší naměřené hodnotě,
4. Beziérova křivka – data jsou interpolovaná pomocí Beziérovky křivky s počátečním bodem naměřené hodnoty pro  $\theta = 75^\circ$ , 0 pro  $90^\circ$  a s derivací v počátečním bodě odpovídající tečně naměřených dat (viz obrázek 11).



Obrázek 11: Extrapolace Beziérovou křivkou v software MiniDiff v2 (převzato z manuálu<sup>[3]</sup>)

Dále lze nastavit, zda mají být naměřená data symetrická podle roviny XZ. V takovém případě jsou hodnoty z opačných poloprostorů zprůměrovány.

#### 4.1.4 Export dat

Software přístroje MiniDiff v2 umožňuje export dat do více možných formátů, ale pro mnoho z nich není veřejně dostupná specifikace. Proto pro následné využití a zpracování dat byl v této práci použit formát společnosti Zemax.



## 5 Měření BRDF

Požadavkem práce bylo změření aspoň 200 vzorků různých materiálů. Naměřená data slouží v další části práce pro realistické vykreslování v herních enginech.

Před samotným měřením vzorků materiálů je nutné provést kalibraci, jak je popsáno v kapitole 4.1.1. Protože jsou naměřená data pevně svázána s kalibrací, je vhodné po kalibraci provést testovací měření známých vzorků. Jako jeden z možných ověřovacích vzorků lze použít i bílý kalibrační vzorek. Je ale nutné změnit jeho orientaci (pootočit).



Obrázek 12: Přístroj MiniDiff v2 v průběhu měření

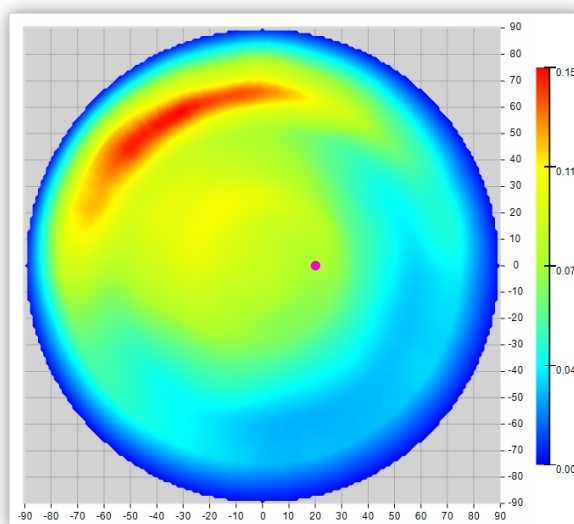
Měření vzorku spočívá v následujících krocích:

- Umístění přístroje na měřený vzorek – Rovný vzorek musí být umístěn na pevné rovné podložce. Na vzorek se následně umístí přístroj stranou s optickou soustavou dolů tak, aby měřící apertura byla položena na vzorku a přístroj stál rovně a stabilně.
- Pořizování snímků povrchu a výpočet BRDF – Tato část je v režii obslužného softwaru přístroje a trvá u jednoho vzorku přibližně 2 minuty a 15 sekund.
- Export – Z obslužného softwaru se exportují naměřená data, pro každou barevnou složku samostatně, do formátu firmy Zemax. Tyto soubory mohou být následně využity při zpracování vytvořenou aplikací `brdf_gui`.

### 5.1 Časté problémy

#### 5.1.1 Nerovnosti povrchu

Jedním z hlavních problémů při měření je situace, kdy se vyskytují nerovnosti povrchu vzorku. Měřící přístroj předpokládá, že normála měřeného povrchu směřuje k čočce přístroje. Pokud tento předpoklad neplatí, dochází k podobným jevům jako na obrázku 13.



Obrázek 13: Nevalidní měření způsobené zvlněním měřeného vzorku (fialový bod určuje očekávaný směr zrcadlového odrazu)

Tento problém může být způsoben několika aspekty: fyzickým tvarem pevného vzorku nebo strukturou vzorku. V případech fyzického tvaru se nabízí možnost vyzkoušení jiné části vzorku bez těchto vad, v případě nevhodné struktury vzorku je nutné tento vzorek vyřadit z měření.

Obzvláště velký problém nastává v případě výskytu vady nerovnosti povrchu na kalibračním vzorku, protože pak jsou touto vadou zatíženy všechny následně měřené vzorky a je nutné jejich opětovné přeměření s použitím kalibračního vzorku bez vady.

### 5.1.2 Průsvitnost vzorku

Dalším problematickým aspektem měření může být částečná průsvitnost vzorku. V tomto případě dochází k nežádoucímu částečnému měření prosvítající podložky nebo dopadajícího světla z okolí. Proto je potřeba tyto vzorky umístit na podložku s minimální odrazivostí (tj. černá matná podložka) a omezit zdroje světla z okolí.

### 5.1.3 Měkkost vzorku

Mezi časté problémy při měření patří i problémy způsobené měkkostí a poddajností měřeného vzorku. Protože se jedná o kontaktní způsob měření, na vzorek působí nemalá síla z hmotnosti přístroje, která může vzorek negativně ovlivnit a změnit jeho tvar. Tyto vzorky většinou není možné správně změřit. Příkladem takového vzorku je látka zvaná softshell (viz obrázek 14, dvě čáry vznikly položením závaží s obdélníkovou podstavou na vzorek).

## 5.2 Výběr vzorků

Vzhledem k skutečnostem zmíněným v kapitole 5.1 je výběr vhodných vzorků značně omezen. Z měření je nutno úplně vyloučit měkké vzorky, které neposkytují pro další zpracování použitelná data. U pevných vzorků je nutné vybrat ty, jejichž povrch je rovný, čistý a mechanicky nepoškozený. Při měření je vhodné se vzorky manipulovat opatrně, aby nedošlo k poškození nebo zašpinění jejich povrchů.

### 5.2.1 Příklady nevhodných vzorků

#### Látky

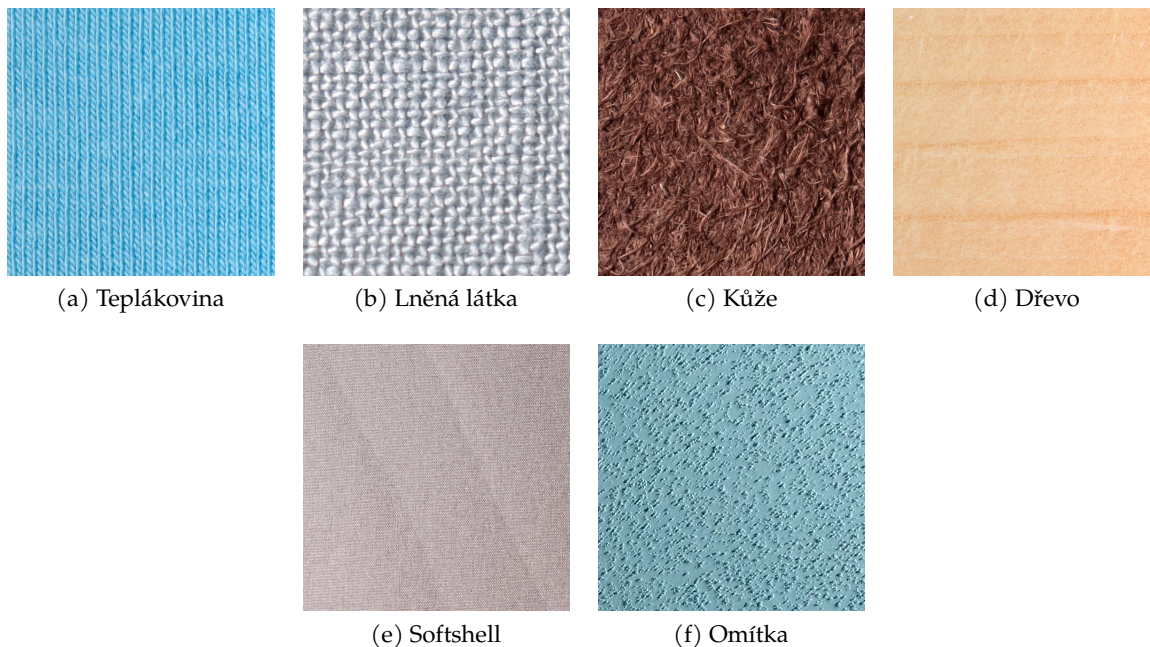
Problematickým aspektem látek je jejich typ vazby a tkaní. V drtivé většině případů jsou vlákna příliš velká nebo příliš od sebe vzdálena, dochází k situaci, ve které nerovnosti povrchu vedou k nesprávnému výsledku měření. Další z důvodu, proč není možné správně naměřit odrazivost látek, může být jejich měkkost a poddajnost.

## Dřevo

Mnoho dřevěných materiálů má příliš výrazné letokruhy narušující povrch a tím i výsledek měření. Částečně je možné opravit vzorek hoblováním, ale tím je i změněna odrazivost vzorku.

## Omítka

Povrch omítky obsahuje jednotlivé částičky, které nejsou dostatečně zanedbatelně malé a negativně ovlivňují výsledek měření.

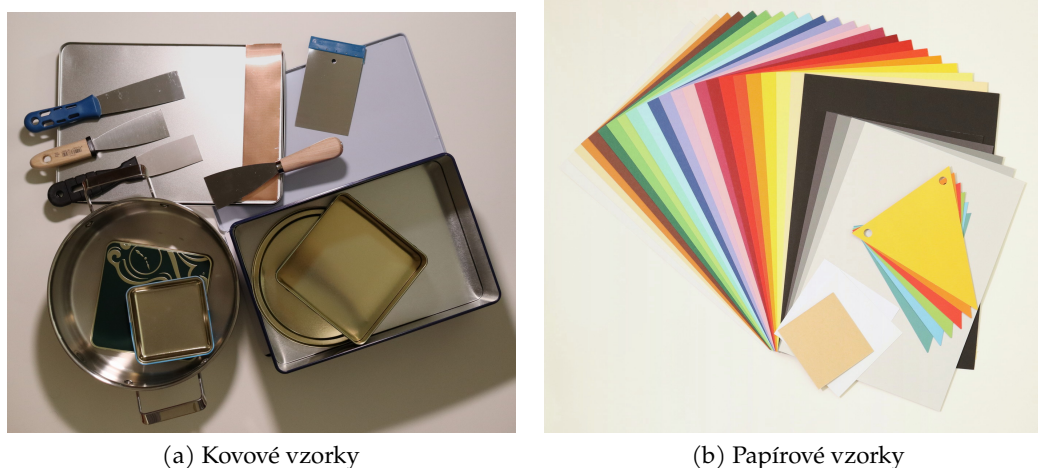


Obrázek 14: Nevhodné typy vzorků

### 5.2.2 Použité vzorky

Pomocí přístroje MiniDiff v2 byly měřeny nejrůznější materiály, ale pro další zpracování bylo vybráno 217 vhodných vzorků. U anizotropních materiálů byly změřeny dvě varianty pro  $\phi_{in} = 0$  a  $\phi_{in} = 90^\circ$ .

Výčet těchto vzorků je uveden v příloze A. V seznamu jsou rozděleny do kategorií podle typů materiálů.



Obrázek 15: Fotografie naměřených vzorků



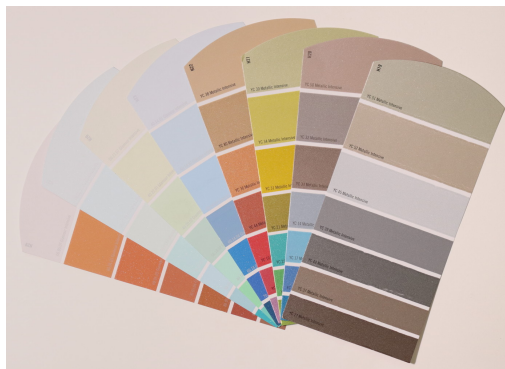
(c) Mix vzorky



(d) Gumové vzorky



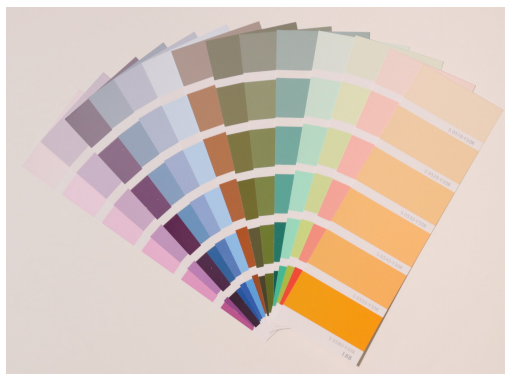
(e) Plstěné vzorky



(f) Metalický vzorník



(g) Vzorník PPG



(h) Vzorník NCS



(i) Plastové vzorky

Obrázek 15: Fotografie naměřených vzorků (pokračování)

## 6 Implementace vizualizačního softwaru

Součástí práce je vytvoření knihovny `brdf_parser` a grafické aplikace `brdf_gui` pro účely vizualizace a zpracování naměřených dat. Aplikace umožňuje volbu extrapolace, náhled na měřená data pomocí laloku odrazivosti, aplikování dat na 3D model osvětlený mapou okolí a export do formátu použitelného pro vykreslování v herních enginech. Dále byly pomocí tohoto nástroje vytvořeny krátká videa ilustrující použití naměřených dat na zobrazeném 3D modelu.

### 6.1 Knihovna `brdf_parser`

Jedná se o knihovnu zajišťující načítání, ukládání a zpracování měřených BRDF dat. Je napsaná v jazyce C++ s využitím knihovny `OpenEXR`<sup>[17]</sup> pro manipulaci s formátem souborů `EXR`. Tento formát je důležitý pro použití v herních enginech.

Knihovna byla navržena s cílem vysoké míry obecnosti nad rámec podpory pro zařízení `MiniDiff v2`. Je tedy pamatováno na snadnou dodatečnou implementaci podpory pro i jiné formáty souborů měřených dat, a tak bylo možné případně pracovat i s daty v podobě mračna bodů. Z tohoto důvodu byla již na počátku vývoje zvolena paměťová reprezentace umožňující práci s mračny bodů.

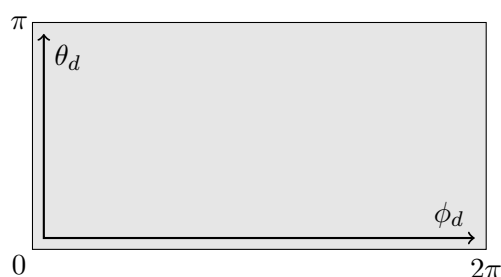
Mezi podporované operace zpracování patří možnost programově přidávat a odebírat jednotlivé naměřené body a provádět operaci hledání nejbližších sousedů. Tyto operace jsou následně využity pro výpočet interpolace a extrapolace dat (využito vyhledávání) a spojování jednotlivých barevných složek (využito přidávání).

#### 6.1.1 Podporované formáty

Knihovna `brdf_parser` pracuje s těmito formáty pro čtení a zápis BRDF dat:

- formát společnosti Zemax (viz kapitola 3.1),
- obrazový formát `EXR`, využívaný pro následné vykreslování.

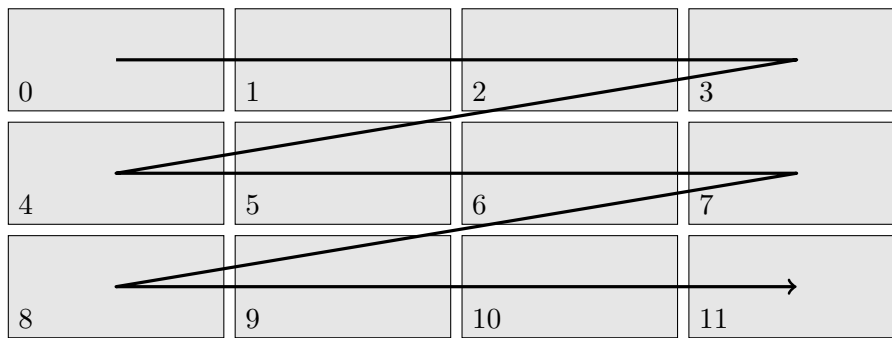
Obrazový formát `EXR` umožňuje efektivní reprezentaci dat vysokého dynamického rozsahu s možností ukládání dat po dlaždicích a podporou různých datových typů pro metadata.<sup>[17]</sup> Data jsou rozdělena do rovin (dlaždic) s daným úhlem dopadu. Vzhledem k tomu, že následné vykreslování je omezeno na izotropní materiály, úhel dopadu je popisován pouze pomocí úhlu od normály  $\theta_{in}$ . Úhel odrazu  $\theta_{out}$  v jednotlivých dlaždicích je vztažen, shodně s formátem společnosti Zemax, ke směru zrcadlového odrazu. Dlaždice používají ekvidistantní válcovou projekci: na ose  $x$  jsou rovnoměrně namapované hodnoty  $\phi_d \in [0, 2\pi]$ , na ose  $y$  hodnoty  $\theta_d \in [0, \pi]$ .



Obrázek 16: Souřadnicový systém v dlaždici

K přiřazení úhlu dopadu k jednotlivým dlaždicím jsou využita metadata, konkrétně uspořádané pole `bsdfTileAngles` určující  $\theta_{in}$ . Dlaždice jsou indexovány po řádcích a nezávisí na počtu dlaždic na řádek.

Mezi ostatní ukládaná metadata patří `bsdfMethod` popisující způsob porízení hodnot BRDF, `bsdfSymmetry` definující možnou symetrii dat a `bsdfType` určující, jestli se jedná o BRDF nebo BTDF.



Obrázek 17: Pořadí indexace dlaždic

### 6.1.2 Slučování barevných složek

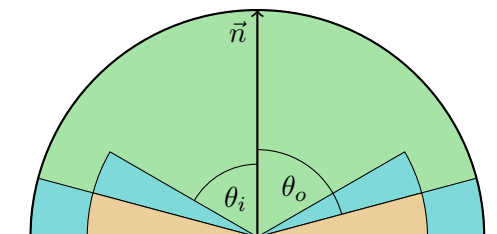
Obslužný software k měřicímu přístroji MiniDiff v2 umožňuje exportovat data pouze pro každou barevnou složku zvlášť (červená, zelená, modrá). Pro následné zpracování a využití není tento přístup praktický a proto je vhodné jednotlivé barevné složky sloužit do společného souboru.

Pro každý bod z jednotlivých barevných souborů se vyhledá v ostatních barevných složkách příslušná hodnota. Pokud neexistuje naměřená hodnota v tomto bodě, je nutné použít odhad pomocí interpolace, resp. extrapolace.

### 6.1.3 Interpolace a extrapolace

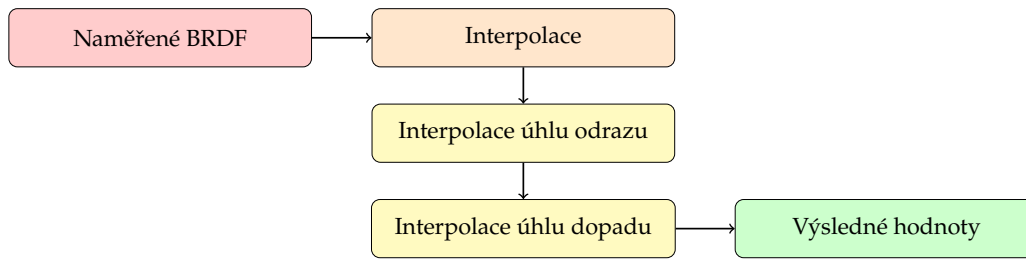
Z technologických důvodů není možné naměřit hodnoty odrazivosti pro všechny možné kombinace úhlů dopadů a odrazů. Jednak z důvodů omezeného rozlišení přístroje a jednak z optického řešení neumožňující pohled pro úhly  $\theta$  blízké  $90^\circ$ . Proto je nutné získaná data interpolovat a extrapolovat.

Cílem interpolace je vytvořit spojitou funkci vytvořenou za pomoci množiny diskretních bodů na podprostoru definovaném těmito body. Naopak extrapolace definuje spojitou funkci mimo tento podprostor. V případě extrapolace BRDF dat, je vhodné extrapolaci rozdělit na dva podproblémy: pro úhel odrazu a úhel dopadu. V případě extrapolace pro úhel odrazu lze využít existujících dat v rovině pro daný úhel dopadu a dochází k extrapolaci pouze pro dvě dimenze. Na rozdíl při extrapolaci pro úhel dopadu dochází k situaci, kde neexistuje žádná naměřená hodnota v žádné z existujících rovin dle úhlu odrazu, která by se dala při výpočtu použít, tj. je nutné provést extrapolaci přes všechny čtyři dimenze. Jedna z možností, jak tento problém vyřešit, je využití již extrapolovaných dat úhlů odrazu a převést na problém dvou dimenzí.



Obrázek 18: Rozdělení hemisféry na oblasti interpolace (zelená), extrapolace odrazu (modrá) a dopadu (oranžová).

Ve vytvořené knihovně `brdf_parser` je tato problematika vyřešena pomocí systému zřetězených instancí třídy `Estimator`. Každá instance má definovanou oblast, pro kterou provádí dopočítávání dat, a vnořenou další instanci třídy `Estimator`, popřípadě objekt reprezentující samotná BRDF data, pokud se jedná o počáteční instanci.



Obrázek 19: Obvyklé zřetězení tříd

Při výpočtu hodnoty pro specifickou dvojici úhlů dochází k postupnému vyhodnocování následujícím způsobem:

1. Pokud daný bod odpovídá přiřazené oblasti, provede se interpolace (resp. extrapolace) za pomoci získaných dat z vnořené třídy.
2. Jinak bez jakýkoliv změn předá požadavek dál vnořené třídě.

Mezi třídy implementované v knihovně `brdf_parser` patří následující:

- Třídy pro interpolaci
  - `NeighborsInterpolator`: Pomocí nejbližších sousedů mračna bodů se vypočítá Shepardova interpolace (viz níže).
  - `LinearAxisInterpolator`: Z mračna bodů vybere nejbližší sousedy v osách souřadného systému sférických souřadnic, následně mezi těmito hodnotami udělá lineární interpolaci.
  - `GridInterpolator`: Vytvoří matici hodnot pro roviny dle úhlu dopadu, následně mezi těmito hodnotami provádí lineární interpolaci.
- Třídy pro extrapolaci
  - `ClampExtrapolator`: Vrátí nejbližší hodnotu v interpolované oblasti.
  - `GradientExtrapolator`: Vypočítá interpolaci kvadratickou Beziérovou křivkou mezi nejbližší hodnotou v interpolované oblasti a jejím procentuálním poměrem (viz níže).

### Shepardova interpolace

`NeighborsInterpolator` používá pro výpočet hodnoty v daném bodě  $\vec{q}$  Shepardovu interpolaci. V mračnu bodů se vyhledá maximálně  $k$  bodů  $\vec{x}_i$  se vzdáleností menší než  $d_{max}$ . Výsledná hodnota BRDF je vyjádřena jako vážený průměr těchto bodů:<sup>[18]</sup>

$$f(\vec{q}) = \begin{cases} f(\vec{x}_i) & \text{pro } \exists x_i : \vec{q} = \vec{x}_i, \\ \frac{\sum_{i=1}^k f(\vec{x}_i) \cdot \|\vec{q} - \vec{x}_i\|^{-2}}{\sum_{i=1}^k \|\vec{q} - \vec{x}_i\|^{-2}} & \text{jinak.} \end{cases} \quad (37)$$

Aby se odstranila nutnost podmínky, je vhodné k vzdálenosti bodů přičíst malé  $\epsilon > 0$ :

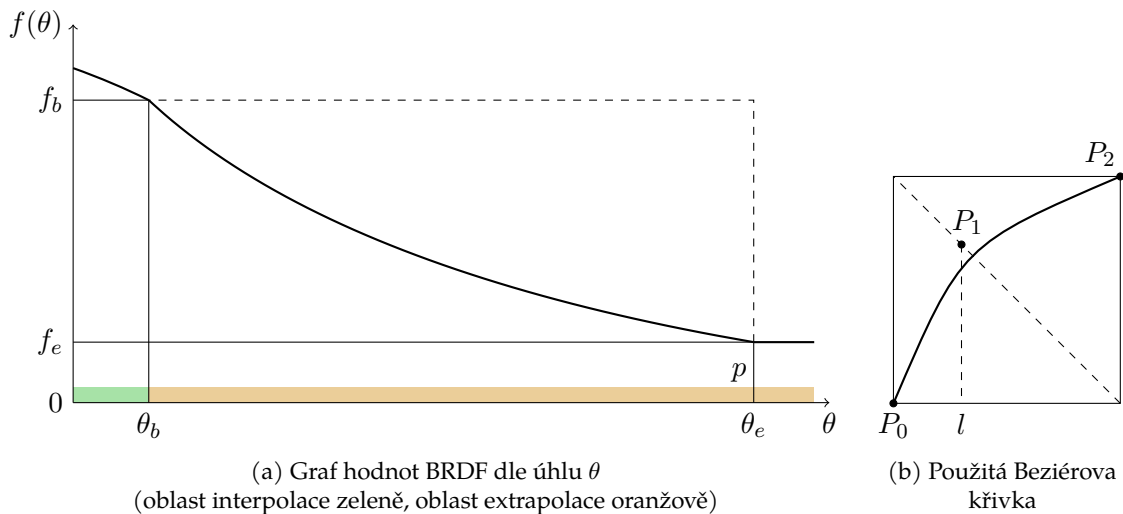
$$f(\vec{q}) = \frac{\sum_{i=1}^k f(\vec{x}_i) \cdot (\|\vec{q} - \vec{x}_i\| + \epsilon)^{-2}}{\sum_{i=1}^k (\|\vec{q} - \vec{x}_i\| + \epsilon)^{-2}}. \quad (38)$$

## Extrapolace pomocí gradientu

GradientExtrapolator provádí interpolaci mezi nejbližší známou hodnotou  $f_b$  a její procentuální částí  $f_e$ :

$$\begin{aligned} f(\vec{\omega}) &= (1 - t) \cdot f(\vec{\omega}_b) + t \cdot (p \cdot \vec{\omega}_b), \\ t &= b\left(\frac{\|\vec{\omega} - \vec{\omega}_b\|}{m}, l\right), \end{aligned} \quad (39)$$

kde  $\vec{\omega}_b$  je souřadnice nejbližšího naměřeného bodu,  $m$  definuje, do jaké maximální vzdálenosti je počítán gradient,  $p$  určuje poměr hodnoty koncového bodu vůči počátečnímu a parametr  $l$  upravující chování Beziérovy křivky.



Obrázek 20: Znárodnění extrapolace gradientem ( $p = 0.2, l = 0.3$ )

Funkce  $b(x, l)$  reprezentuje upravenou verzi dvoudimenzionální kvadratické Beziérovy křivky. Počáteční  $\vec{P}_0$  a koncový  $\vec{P}_2$  bod je pevně stanoven na  $(0, 0)$ , resp.  $(1, 1)$ . Bod  $\vec{P}_1$  leží na přímce kolmé je spojnicí bodů  $\vec{P}_0, \vec{P}_2$  a je ovlivněn parametrem  $l$  odpovídající  $x$  souřadnici tohoto bodu (tj.  $\vec{P}_1 = (l, 1 - l)$ ). Parametr  $t$  Beziérovy křivky není přímo určen, ale dochází k jeho hledání za předpokladu pevně stanoveného výsledné  $x$  souřadnice. Hodnota funkce  $b(x, l)$  odpovídá  $y$  souřadnici nalezeného bodu Beziérovy křivky:

$$b(x, l) = y \text{ za předpokladu } \exists t : \begin{bmatrix} x \\ y \end{bmatrix} = 2 \cdot (1 - t) \cdot t \cdot \begin{bmatrix} l \\ 1 - l \end{bmatrix} + t^2. \quad (40)$$

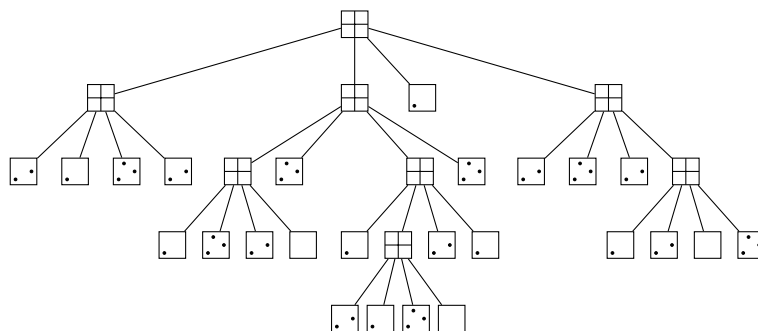
### 6.1.4 Datová struktura pro vyhledávání nejbližších sousedů

Zpracování naměřených dat vyžaduje časté vyhledávání. Převážně jde o hledání  $n$  nejbližších sousedů. Pro úpravu dat musí být umožněno přidávat a odstraňovat body, reprezentující naměřené hodnoty. Pro tyto operace se využívá zobecněná datová struktura vycházející z rovnoměrného (PR) kvadrantového stromu (quadtree)<sup>[19]</sup>.

Pro popis umístění bodu je použit kartézský souřadnicový systém. Vzhledem k závislosti vektorů popisující směr úhlů dopadu a odrazu ( $\|\vec{v}\| = 1$ ), je možné redukovat počet dimenzí na čtyři vyřazením vertikální osy  $y$ , která lze zpětně dopočítat jako  $y = \sqrt{1 - x^2 - z^2}$ . Touto operací sice dochází k omezení na podprostor  $y \geq 0$ , ale pro ukládání BRDF dat to nepředstavuje problém.



Stromová vyhledávací struktura se skládá z větví a listů. Každý vrchol reprezentuje osově zarovnanou oblast, do které patří body z jeho podgrafu. Listy grafu obsahují maximálně  $m$  bodů, při překročení limitu je nutné podgraf rozdělit. Větve grafu uložené body neobsahují a obsahují pouze seznam  $2^d$  potomků, kde  $d$  je počet dimenzí. Každému z potomků náleží oblast s poloviční délkou ve všech osách, od rohu k středu  $s$  nadřazené oblasti, celkově vyplňují celou oblast nadřazeného vrcholu. Potomci jsou očíslovány binárním číslem určující pozici oblasti: každá číslice představuje jednu osu, 0 pokud oblast leží v podprostoru  $x_i \leq s_i$ , jinak 1.



Obrázek 21: Strom vyhledávací struktury

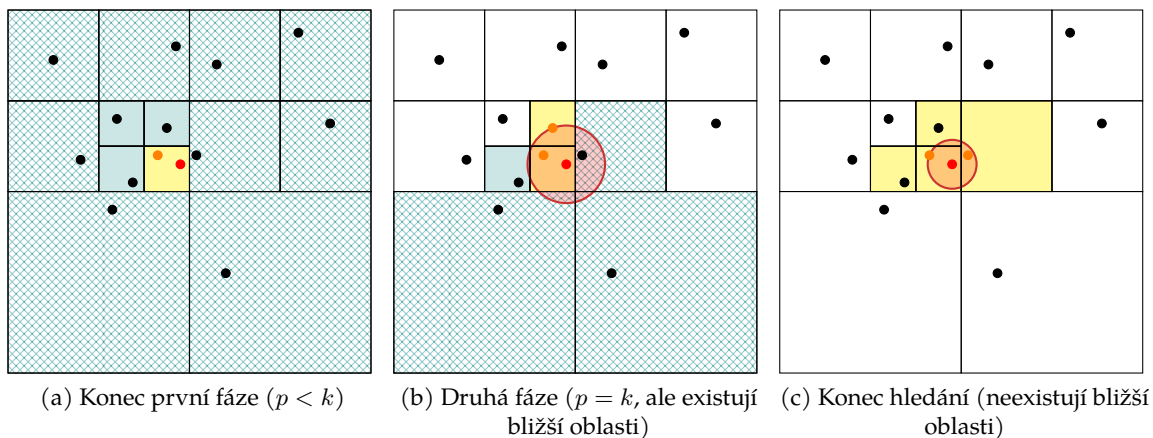
Při vyhledávání jednoho konkrétního bodu dochází k postupnému procházení stromu, u větvi se vybírá potomek v jehož oblasti leží hledaný bod. V listu se následně prohledá seznam uložených (maximálně  $m$ ) bodů. Tato operace má u rovnoměrně rozloženého stromu složitost  $O(\log_{2^d} n + \frac{m}{2})$ .

Pro přidání bodu do stromu se vyhledá list s oblastí náležící nově přidávanému bodu a dojde k přidání do seznamu. V případě překročení limitu  $m$  bodů je nutné list nahradit větví a provést nové uložení bodů bývalého listu. Tato operace má u rovnoměrně rozloženého stromu amortizovanou složitost odpovídající složitosti vyhledávání listu + konstanta (vlození do listu  $O(1)$ , pro každý  $m$ . prvek je potřeba provést vložení  $m$  prvků do nových listů).

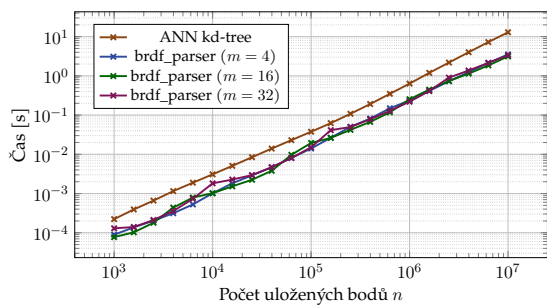
Odstranění bodu probíhá obdobně. Nejdříve dojde k vyhledání příslušného listu a odstranění bodu ze seznamu. Pokud by touto operací vznikla větev, obsahující ve svém podstromu méně než  $l < m$  bodů, všechny body podstromu se vloží do nového listu, který následně nahradí existující větev. Amortizovaná časová složitost je shodná s přidáním prvku.

Vyhledávání  $k$  nejbližších sousedů probíhá pomocí hybridního algoritmu<sup>[20]</sup> adaptovaného pro  $k > 1$ . Pro účely ukládání nalezených sousedů se používá uspořádaný seznam (počet prvků označme  $p$ ) dle vzdálenosti od hledaného vrcholu, v situaci  $p > k$  dochází k odstranění  $p - k$  nejvzdálenějších prvků. V první fázi se nalezne vrchol stromu (větev nebo list) obsahující méně nebo rovno  $k - p$  bodů. Všechny tyto body se přidají seznamu. V druhé fázi dochází u nadřazených vrcholů ke kontrole, jestli  $p < k$  nebo vzdálenost nejvzdálenějšího prvku seznamu je větší než vzdálenost k neprozkoumanému vrcholu. V takovém případě je nutné provést vyhledávání nejbližších sousedů na jejich podstromech. Setříděním potomků dle vzdálenosti od hledaného bodu lze podpořit pravděpodobnost nalezení nejbližších bodů před prozkoumáním vzdálených potomků. Pro rovnoměrně rozložený strom dochází průměrně k prozkoumání o řád více prvků.

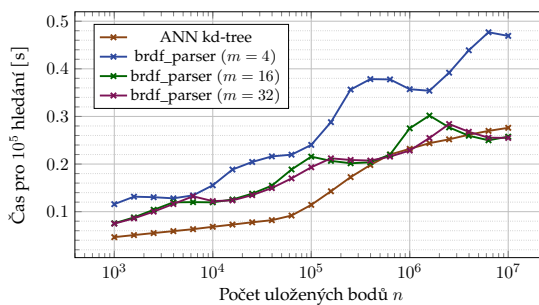
Pro porovnání výkonu této datové struktury byla vybrána implementace kd-stromu knihovny ANN<sup>[21]</sup>. Porovnávání bylo provedeno ve třech kategoriích: čas na vytvoření stromu, vyhledání  $k$  nejbližších sousedů a paměťová náročnost (výsledky testů viz obrázek 23). Výkyvy časové a paměťové náročnosti jsou způsobeny rozdílnou průměrnou zaplněností listů. Testování bylo provedeno na procesoru Intel i5-9600K @ 4.5 GHz, Clang 15.0.7 -03.



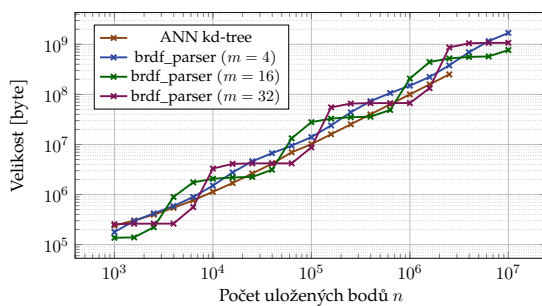
Obrázek 22: Průběh hledání  $k = 2$  nejbližších sousedů. Prozkoumané oblasti žlutě, tyrkysově potenciální oblasti (šrafované oblasti mimo aktuálně rozhodovaný vrchol).



(a) Vytváření stromu



(b) Hledání  $k = 10$  sousedů



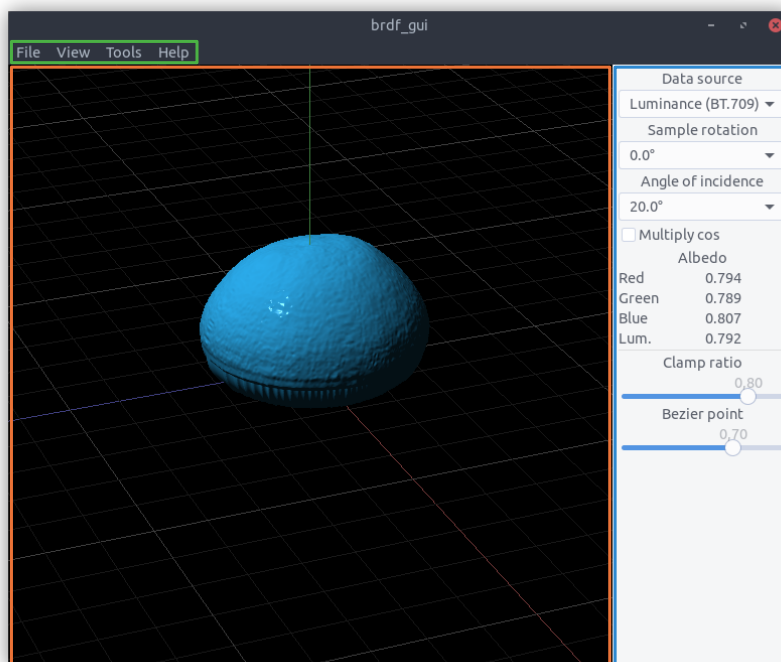
(c) Velikost zabrané paměti RAM

Obrázek 23: Porovnání ANN a implementované datové struktury pro různá  $m$ .

## 6.2 Aplikace brdf\_gui

brdf\_gui je grafická nadstavba nad knihovnou brdf\_parser umožňující načíst, zobrazit, upravit a exportovat naměřená BRDF data. Je vytvořena pomocí knihovny Gtkmm<sup>[22]</sup> a vykreslování probíhá pomocí OpenGL.

Po spuštění aplikace se zobrazí hlavní okno s výchozím BRDF odpovídající Lambertově modelu. Otevřené okno (viz obrázek 24) se skládá ze tří částí: horní menu (zelený rámeček), hlavní zobrazení (oranžový rámeček) a postranní panel (modrý rámeček).

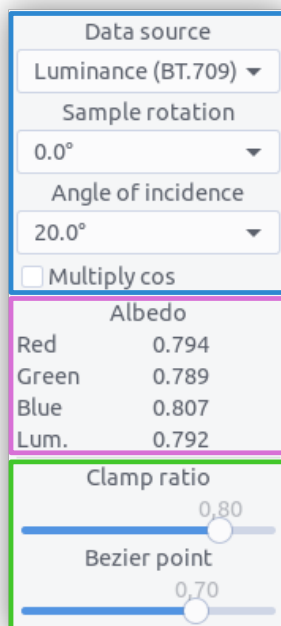


Obrázek 24: Hlavní okno aplikace brdf\_gui

Horní menu je obsahuje možnosti pro manipulaci se soubory, zobrazením, nastavením a je rozděleno do čtyř kategorií:

- File
  - Open (**Ctrl** + **O**) – Otevře naměřená BRDF data ve formátu firmy Zemax.
  - Export (**Ctrl** + **E**) – Exportuje aktuálně otevřené BRDF do formátu OpenEXR.
  - Settings – Zobrazí okno s nastavením (viz kapitola 6.2.4).
  - Quit (**Ctrl** + **Q**) – Ukončí aplikaci.
- View
  - Reflection blob (**F1**) – Přepne režim zobrazení na zobrazení laloku (viz kapitola 6.2.1).
  - Environment (**F2**) – Přepne režim zobrazení na zobrazení modelu osvětleného mapou okolí (viz kapitola 6.2.2).
- Tools
  - Merge (**Ctrl** + **M**) – Zobrazí okno pro spojení jednotlivých barevných složek měřených dat (viz kapitola 6.2.5).
- Help
  - About – Zobrazí okno s informacemi o této aplikaci.

Postranní panel (viz obrázek 25) obsahuje nastavení aktuálního zobrazení (modrý rámeček) a informace (růžový rámeček) aktuálního zobrazení a ovládání parametrů extrapolace (zelený rámeček).



Obrázek 25: Postranní panel

Ovládání parametrů extrapolace je zobrazováno pouze, pokud je zvolena extrapolace pomocí třídy `GradientExtrapolator`. V ostatních případech není zobrazováno nic.

Pro `GradientExtrapolator` lze měnit dva parametry:

- **Clamp ratio** – poměr hodnoty koncového bodu gradientu vůči počátečnímu bodu (parametr  $p$  dle kapitoly 6.1.3).
- **Bezier point** – vychýlení Beziérovky křivky (parametr  $l$  dle kapitoly 6.1.3), 0.5 odpovídá lineární interpolaci, 1 odpovídá plně vypouklému tvaru a 0 plně vydutému tvaru.

### 6.2.1 Zobrazení laloku

Zobrazení laloku (viz obrázek 26) slouží k přímé vizualizaci naměřených BRDF dat pomocí zdeformované hemisféry. Jedná se o pohled pro daný úhel dopadu. Každý bod hemisféry popisuje určitý úhel odrazu pomocí směrového vektoru od počátku souřadného systému. Vzdálenost bodu od počátku odpovídá hodnotě BRDF v daném směru. Protože zpracovávaná data mohou být určena pro tři barevné složky RGB, je nutné z této trojice barev vyhodnotit skalární hodnotu – buď jako jednu z barev, jejich průměr nebo vypočítaný jas. Jako trojúhelníková síť se používá dělená plocha (subdivision surface) dvacetistěnu.

Pomocí levého tlačítka a pohybu myši lze měnit úhel pohledu kamery, pomocí pravého tlačítka úhel osvětlení zobrazovaného laloku. Kolečkem myši se mění zorné pole kamery.

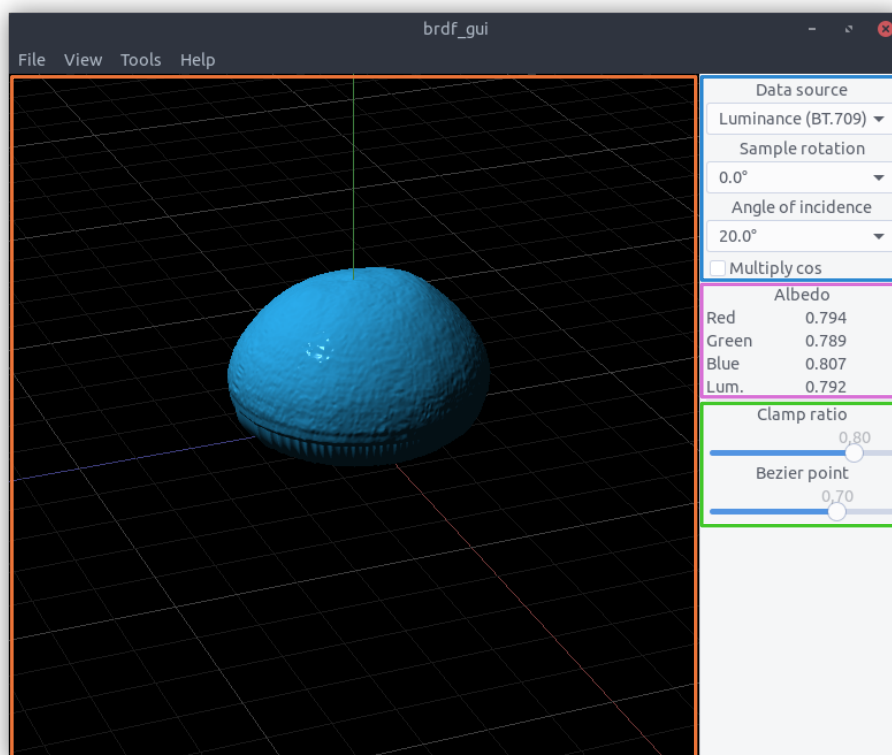
V kategorii nastavení zobrazení v horní části postranního panelu se nachází následující parametry:

- **Data source** – Výběr převodní funkce barevných složek na skalární hodnotu vzdálenosti bodů od počátku souřadného systému. Pro monochromatická data je tento výběr neaktivní.
- **Luminance (BT.709)** – Vrací hodnotu jasu dle standardu BT.709:<sup>[23]</sup>

$$c(r, g, b) = 0.2126 \cdot r + 0.7152 \cdot g + 0.0722 \cdot b.$$

- Red – Vrací hodnotu červené složky:  $c(r, g, b) = r$ .
- Green – Vrací hodnotu zelené složky:  $c(r, g, b) = g$ .
- Blue – Vrací hodnotu blue složky:  $c(r, g, b) = b$ .
- Average – Vrací průměr barevných složek:  $c(r, g, b) = \frac{r+g+b}{3}$ .
- Sample rotation – Výběr aktivního úhlu  $\phi$  směru odrazu. Jsou k dispozici pouze úhly pro které existuje dostatek naměřených bodů.
- Angle of incidence – Výběr aktivního úhlu  $\phi$  směru odrazu. Jsou k dispozici pouze úhly pro které existuje dostatek naměřených bodů.
- Multiply cos – Určuje, jestli BRDF data mají být vynásobeny  $\cos \theta_{out}$ .

Mezi informacemi zobrazení se nachází vypočítané albedo pro aktuální úhel odrazu, pro jednotlivé barevné složky a pro vypočítaný jas. Při změně parametrů extrapolace nebo pohledu dochází na pozadí k přepočítání.



Obrázek 26: Zobrazení laloku

### 6.2.2 Zobrazení modelu

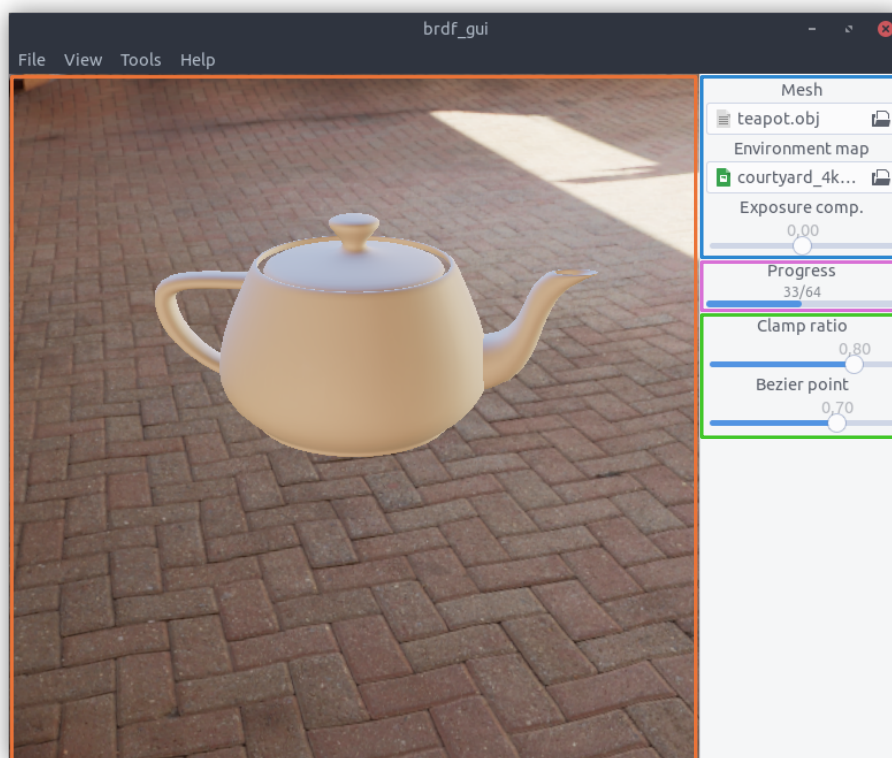
Zobrazení modelu (viz obrázek 27) slouží k náhledu BRDF aplikovaného na 3D model. Jako zdroj světla se používá mapa okolí zobrazená pomocí ekvidistantní válcové projekce. Technický popis vykreslování je popsán v kapitole 7.2.

Stejným způsobem jako u zobrazení lze pomocí levého tlačítka a pohybem myši měnit úhel pohledu kamery. Pomocí kláves 1, 2, 3, 4, 5 lze přepínat mezi mapami okolí definovaných v nastavení. Klávesové zkratky Shift + 1, Shift + 2, Shift + 3, Shift + 4, Shift + 5 přepínají 3D modely.

Kategorie nastavení zobrazení obsahuje možnosti ohledně výběru 3D modelu, mapy okolí a expozice:

- Mesh – Tlačítko pro výběr 3D modelu ve formátu obj. Změnou 3D modelu pomocí klávesové zkratky dojde k změně i této komponenty.
- Environment map – Tlačítko pro výběr mapy okolí ve formátu exr. Změnou mapy okolí pomocí klávesové zkratky dojde k změně i této komponenty.
- Exposure comp. – Kompenzace expozice osvětlení.

V sekci informací zobrazení se nachází ukazatel průběhu vykreslování.



Obrázek 27: Zobrazení modelu

### 6.2.3 Vytvoření animace zobrazení

V zobrazení modelu je možné vytvořit snímky animace zobrazení. Po stisknutí klávesy **R** se zobrazí dialog výběru složky, do které se následně vykreslí 120 snímků modelu otáčející se kolem svislé osy. Tyto snímky ve formátu EXR mohou dle nastavení obsahovat data vysokého dynamického rozsahu (HDR) bez tónového mapování nebo zpracovaná data ve standardním dynamickém rozsahu (SDR) s tónovým mapováním.

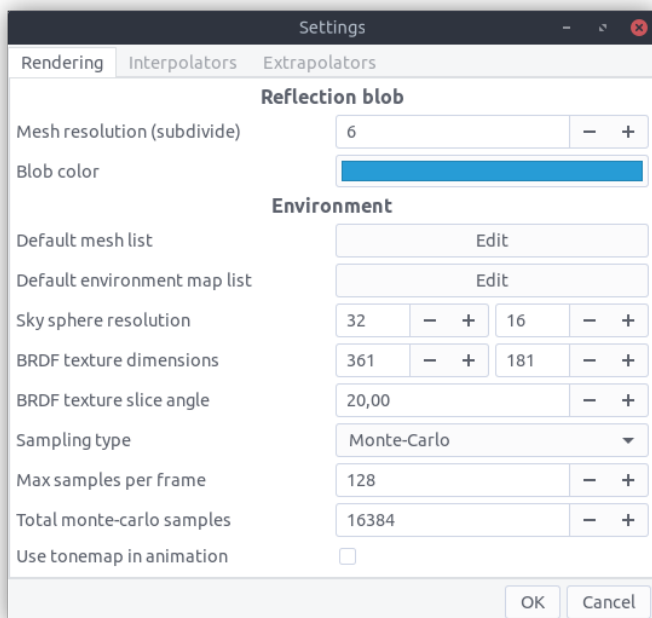
Pomocí tohoto nástroje byly vytvořena krátká videa zobrazení dle zadání této práce (viz přílohy B a C).

### 6.2.4 Nastavení

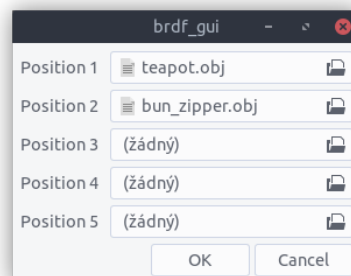
Nastavení obsahuje možnosti upravující celkové chování aplikace. Změny v nastavení se ukládají do konfiguračního souboru uloženého v podsložce domovského adresáře uživatele. Pro Linux se jedná o `~/.config/brdf_gui/config.json`.

## Nastavení vykreslování (viz obrázek 28)

- Reflection blob – nastavení zobrazení laloku
  - Mesh resolution – Počet iterací operace dělení plochy (subdivision surface). V každé iteraci dochází k nahrazení každého trojúhelníku čtyřmi menšími s nově vzniklými vrcholy ve středech hran původního trojúhelníku.
  - Blob color – Barva trojúhelníkové sítě laloku. Jedná se pouze o vizuální aspekt zobrazení a nemá vliv na BRDF data.
- Environment – nastavení zobrazení modelu
  - Default mesh list – Tlačítko na výběr modelů, které jsou možné v zobrazení načíst klávesovými zkratkami.
  - Default environment map list – Tlačítko na výběr map okolí, které jsou možné v zobrazení načíst klávesovými zkratkami.
  - Sky sphere resolution – Rozlišení trojúhelníkové sítě sféry mapy okolí.
  - BRDF texture dimensions – Rozlišení dlaždice BRDF textury (viz kapitola 6.1.1) používané pro vykreslování modelu.
  - BRDF texture slice angle – Úhel  $\theta_{in}$  mezi jednotlivými dlaždicemi BRDF textury.
  - Sampling type – Výběr druhu vzorkování (viz kapitola 7.2).
    - Deterministic – Deterministická integrace přes každý pixel dlaždice BRDF textury.
    - Monte-Carlo – Monte-Carlo integrace s distribuční funkcí rovnoměrného rozdělení.
  - Max samples per frame – Maximální počet vzorků z mapy okolí získaných v jednom snímku.
  - Total monte-carlo samples – Konečný počet vzorků při Monte-Carlo vzorkování.
  - Use tonemap in animation – Volba, zda uložené snímky animace mají být v nezpracované HDR podobě, nebo mají mít aplikované tónové mapování.



(a) Založka vykreslování

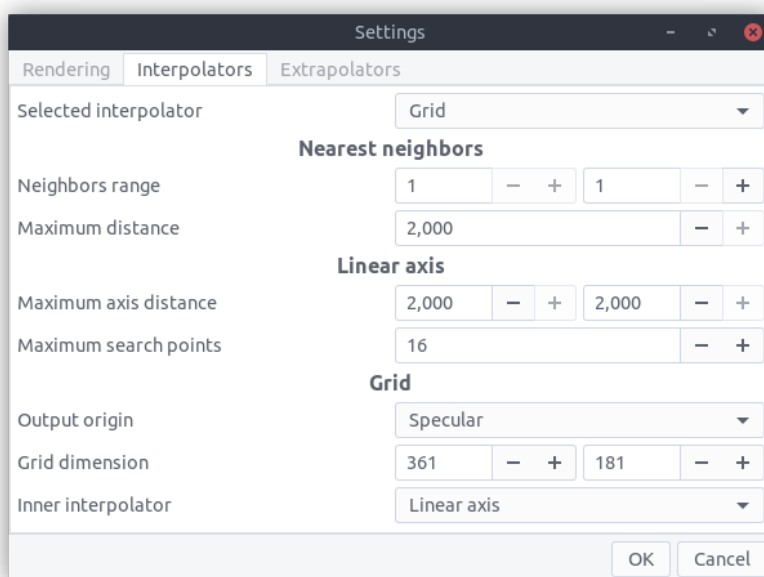


(b) Výběr předdefinovaných modelů/map okolí

Obrázek 28: Nastavení vykreslování

## Nastavení interpolace (viz obrázek 29)

- Selected interpolator – Vybraná třída interpolace používaná pro všechny otevřené soubory.
- Nearest neighbors
  - Neighbors range – Rozsah (minimální a maximální hodnota) počtu nejbližších sousedů. Pokud v zadaném rozsahu nebyl nalezen požadovaný minimální počet bodů, považuje se hodnota hledaného bodu za neexistující a je nulová.
  - Maximum distance – Maximální povolená vzdálenost bodu při hledání nejbližších sousedů.
- Linear axis
  - Maximum axis distance – Maximální vzdálenost hledaných bodů v jednotlivých osách. První hodnota ovlivňuje vzdálenost pro  $\theta_{in}$  a  $\phi_{in}$ , druhá hodnota ovlivňuje  $\theta_{out}$  a  $\phi_{out}$ .
  - Maximum search points – Maximální počet hledaných bodů.
- Grid
  - Output origin – Určuje směrový vertikální vektor definující počátek sférického souřadnicového systému.
    - Up – Vektor shodný s osou  $y$  pro libovolné úhly dopadu.
    - Specular – Vektor odpovídající zrcadlovému směru odrazu.
  - Grid dimension – Rozměry dlaždice pro daný směr dopadu.
  - Inner interpolator – Výběr třídy interpolace používané pro výpočet bodů mřížky.



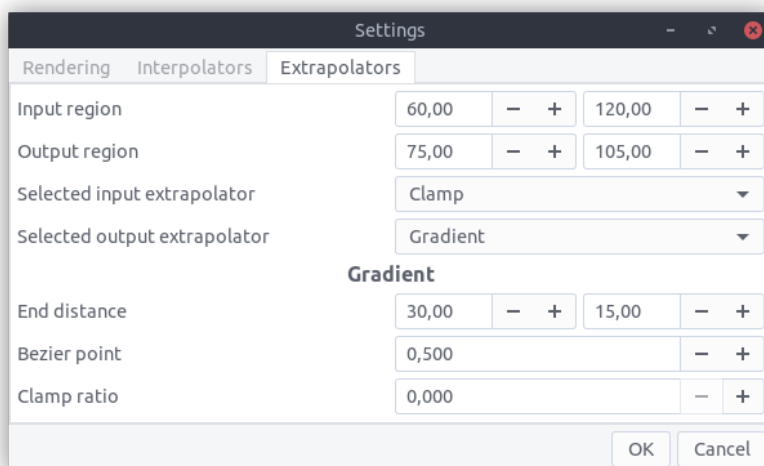
Obrázek 29: Nastavení interpolace

## Nastavení extrapolace (viz obrázek 30)

- Input region – Rozsah úhlů dopadu  $\theta_{in}$  (ve stupních) pro který dochází k extrapolaci dat.
- Output region – Rozsah úhlů odrazu  $\theta_{out}$  (ve stupních) pro který dochází k extrapolaci dat.
- Selected input extrapolator – Vybraná třída extrapolace používaná pro úhly dopadu  $\theta_{in}$ .



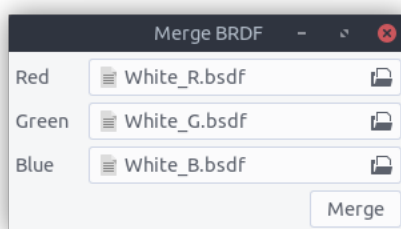
- Selected output extrapolator – Vybraná třída extrapolace používaná pro úhly odrazu  $\theta_{out}$ .
- Gradient
  - End distance – Vzdálenost koncového bodu gradientu od počátečního bodu (parametr  $m$  dle kapitoly 6.1.3).
  - Bezier point – Výchozí hodnota interaktivně měnitelného parametru  $p$ .
  - Clamp ratio – Výchozí hodnota interaktivně měnitelného parametru  $l$ .



Obrázek 30: Nastavení extrapolace

### 6.2.5 Slučování barevných složek

Tento dialog (viz obrázek 31) slouží k spojení souborů pro jednotlivé barevné složky do jednoho souboru obsahující RGB. Po výběru jednoho ze souborů dochází k pokusu o nalezení souborů pro ostatní barvy. Pro tento účel se předpokládá, že názvy slučovaných souborů obsahují podřetězec R nebo Red pro červenou barvu, G nebo Green pro zelenou a B nebo Blue pro modrou barvu. Po kliknutí na tlačítko Merge dojde k zobrazení dialogu pro výběr umístění cílového souboru a následnému spojení způsobem definovaným v kapitole 6.1.2.

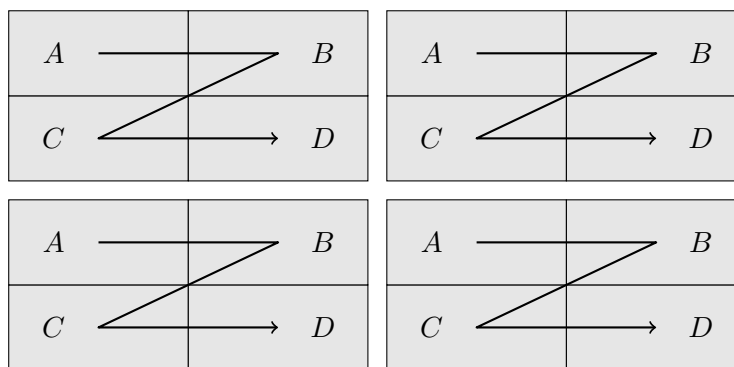


Obrázek 31: Okno slučování barevných složek

### 6.2.6 Vytváření atlasu BRDF

Atlas BRDF je technika, pomocí které je možné uložit více jednotlivých vzorků naměřených BRDF dat do jednoho obrázku ve formátu EXR. Tento způsob ukládání dat je používaný ve vytvořeném shaderu pro Unreal Engine (viz kapitola 9.2).

Atlas je stejným způsobem, jako při ukládání jediného vzorku naměřených BRDF dat, rozložen na jednotlivé dlaždice. Každá dlaždice se ale skládá z  $n \times m$  menších (vnitřních) dlaždic obsahující BRDF data jednotlivých vzorků se shodným  $\theta_{in}$ . Indexace vnitřních dlaždic probíhá stejným způsobem jako u vnějších dlaždic (viz obrázek 32).



Obrázek 32: Pořadí indexace vnitřních dlaždic atlasu

Aktuálně lze atlas vytvořit pouze z příkazové řádky a není možné tuto operaci provést pomocí grafického rozhraní.

### 6.2.7 Volání z příkazové řádky

Aplikace umožňuje provést základní operace přes příkazovou řádku bez uživatelského rozhraní. Jedná se konkrétně o operace slučování barevných složek, exportu dat do formátu EXR a vytvoření atlasu potřebného pro vykreslování v Unreal Engine. Pro všechny tyto operace se používají výchozí parametry z nastavení.

```
./brdf_gui [typ] [input] [output 1] [output 2] [...]
```

První argument určuje typ operace (merge, export nebo atlas), druhý definuje výstupní soubor a ostatní argumenty jsou vstupy závislé na daném typu operace.

#### Slučování barevných složek

```
./brdf_gui merge output.bsdf red.bsdf green.bsdf blue.bsdf
```

Vstupní argumenty jsou cesty vstupních dat pro červenou, zelenou a modrou barvu (v tomto pořadí).

#### Export

```
./brdf_gui export output.exr input.bsdf
```

Jediný vstupní argument je cesta k souboru měřených dat ve formátu firmy Zemax.

#### Atlas

```
./brdf_gui atlas atlas.exr input1.exr input2.exr [...]
```

Tato operace podporuje libovolné množství (minimálně jeden) vstupních argumentů v podobě cest k souborům dat ve formátu EXR. Tyto vstupní soubory jsou spojeny do atlasu ve stejném pořadí.

## 7 Implementace shaderů

Pojmem shader se označuje uživatelsky definovaný program upravující chování určitých částí vykreslovacího řetězce, běžícího na grafické kartě. Konkrétně se jedná o vertex shader zajišťující transformaci bodů trojúhelníkové sítě (obecně primitiv, může se jednat i o úsečky nebo body) z prostoru objektu do normalizovaného prostoru pohledu; a fragment shader zajišťující výpočet barvy pixelů (obecně hodnoty fragmentů, jeden pixel se může skládat z více vzorků i více výstupních parametrů) zaujímající prostor jednotlivých primitiv. Existují i jiné varianty shaderů, ale jejich popis je nad rámec této práce.<sup>[24]</sup>

Obvyklým cílem fragment shaderu je zjednodušený výpočet zobrazovací rovnice (viz kapitola 1.3) pomocí hodnot BRDF a přichozícího osvětlení. Implementovaný shader umožňuje zobrazit izotropní materiály pomocí naměřených dat BRDF.

V rámci této práce jsou BRDF data reprezentovaná pomocí trojrozměrné textury vytvořené z obrázku ve formátu EXR (viz kapitola 6.1.1). Pro správné vytvoření 3D textury je nutné splnit dodatečné podmínky: úhel  $\theta_{in}$  první dlaždice musí být 0 a rozdíl mezi každými dvěma dlaždicemi je shodný:

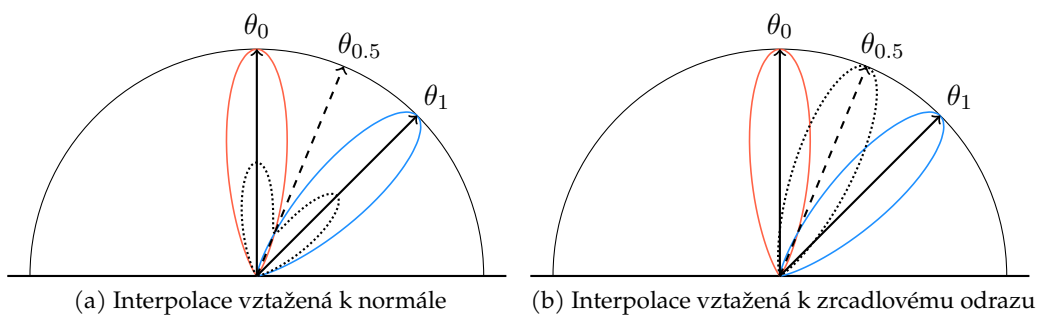
$$\begin{aligned} \theta_0 &= 0, \\ \theta_{i+1} - \theta_i &= k; k \in \mathbb{R}^+. \end{aligned} \quad (41)$$

Díky těmto podmínkám je možný převod ze sférických souřadnic do prostoru normalizovaných souřadnic textury pomocí vztahu:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \frac{\phi_d}{2\pi} \\ \frac{\theta_d}{\pi} \\ \frac{\theta_{in}}{\theta_{in_m}} \end{bmatrix}, \quad (42)$$

kde  $\theta_{in_m}$  vyjadřuje úhel dopadu pro poslední z dlaždic textury.

Souřadnicový systém dlaždic textury vztažený k zrcadlovému odrazu přináší výhodu v podobě možnosti použití lineární interpolace mezi jednotlivými dlaždicemi textury.



Obrázek 33: Porovnání lineární interpolace dlaždic vztažených k normále a k zrcadlovému odrazu

### 7.1 Přímé osvětlování

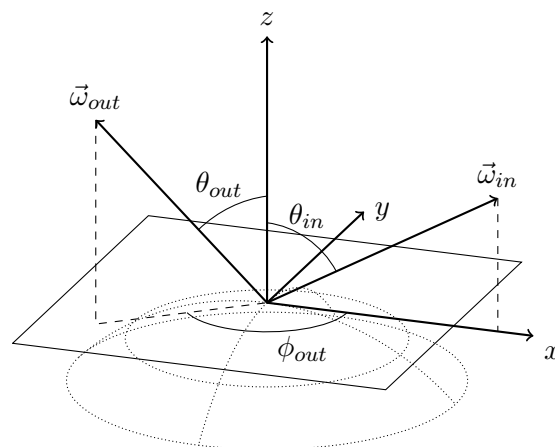
Výpočet přímého osvětlování spočívá ve vynásobení specifické hodnoty BRDF k danému směru dopadu  $\vec{\omega}_{in}$  (směr od zdroje světla ve světových souřadnicích – world space) a směru odrazu  $\vec{\omega}_{out}$  (směr ke kameře ve světových souřadnicích) intenzitou daného zdroje světla:

$$L_{out}(\lambda) = f(\vec{\omega}_{in}, \vec{\omega}_{out}, \lambda) \cdot L_{in}(\lambda) \cdot \cos \theta. \quad (43)$$

Oproti zobrazovací rovnici se zde nenachází integrál přes hemisféru, zdroje světla se předpokládají za nekonečně vzdálené (pro směrové světlo – directional light), resp. nekonečně malé (pro směrové světlo) a neuvažují se odrazy od objektů ve scéně. Proto je tento způsob osvětlování velice rychlý (je nutné vyhodnotit BRDF jenom pro jedny souřadnice), ale neposkytuje příliš realistický výsledek (právě kvůli zanedbání mnoha aspektů).

Pro nalezení hodnoty odrazivosti pro daný směr je vhodné nejdříve převést směr dopadu a odrazu do tečného prostoru, aby se zjednodušily následné operace. Používaný tečný prostor je založen na normalizovaném vektoru projekce směru dopadu  $\vec{t}$  do tečné roviny. Tento tečný prostor, na rozdíl od kartézského souřadnicového systému používaného v ostatních kapitolách, je pravotočivý s osou  $z$  směřující vzhůru. Transformační matice je definovaná jako:

$$T = \begin{pmatrix} \vec{t} & \vec{n} \times \vec{t} & \vec{n} \end{pmatrix}. \quad (44)$$



Obrázek 34: Tečný prostor využívaný při vykreslování

Směr dopadu v tečném prostoru se následně převede na sférickou souřadnici  $\theta_{in} = \cos(z)$  ( $\phi_{in}$  je vždy nulový). Pomocí sférických souřadnic zrcadlového odrazu  $\vec{\omega}_r = (\pi, \theta_{in})$  se vytvoří transformační matice z tečného prostoru do prostoru vztaženého k zrcadlovému odrazu:

$$S = (R_z(\pi) \cdot R_y(\theta_{in}))^T = \begin{pmatrix} -\cos \theta_{in} & 0 & -\sin \theta_{in} \\ 0 & -1 & 0 \\ -\sin \theta_{in} & 0 & \cos \theta_{in} \end{pmatrix}. \quad (45)$$

Zde je vhodné upozornit, že tato rotační matice je symetrická, tzn. inverzní matice  $S^{-1}$  je shodná s maticí  $S$ .

Výsledné souřadnice  $\theta_d$  a  $\phi_d$  (viz obrázek 7) lze získat z vektoru úhlu odrazu v tečném prostoru transformovaného maticí  $S$ . Celkově tedy souřadnice textury se vypočítají jako:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \frac{f_\phi(S \cdot T \cdot \vec{v})}{2\pi} \\ \frac{f_\theta(S \cdot T \cdot \vec{v})}{\pi} \\ \frac{f_\theta(T \cdot \vec{l})}{\theta_{inm}} \end{bmatrix}, \quad (46)$$

kde funkce  $f$  převádí z kartézských do sférických souřadnic,  $\vec{v}$  reprezentuje vektor směru odrazu ve světových souřadnicích a  $\vec{l}$  vektor směru dopadu ve světových souřadnicích.

## 7.2 Osvětlování mapou okolí

Při osvětlování mapou okolí dochází k opačnému procesu, než při osvětlování přímým zdrojem světla – pro definovaný bod BRDF textury se hledá směr dopadu  $\vec{\omega}_{in}$  a příslušná intenzita mapy okolí. Celý proces odpovídá zobrazovací rovnici (při  $L_{emit}(\vec{\omega}_{out}, \lambda) = 0$ ) s několika omezeními: neuvažují se odrazy od objektů ve scéně, jediným zdrojem světla je mapa okolí a integrál je tedy nahrazen numerickou integrací pomocí vážené sumy:

$$L_{out}(\vec{\omega}_{out}, \lambda) = \sum_{i=0}^{i_{max}} \sum_{j=0}^{j_{max}} w_{i,j} \cdot f(\vec{\omega}_{i,j}, \vec{\omega}_{out}, \lambda) \cdot L_{in}(\vec{\omega}_{i,j}, \lambda) \cdot \cos \theta_j, \quad (47)$$

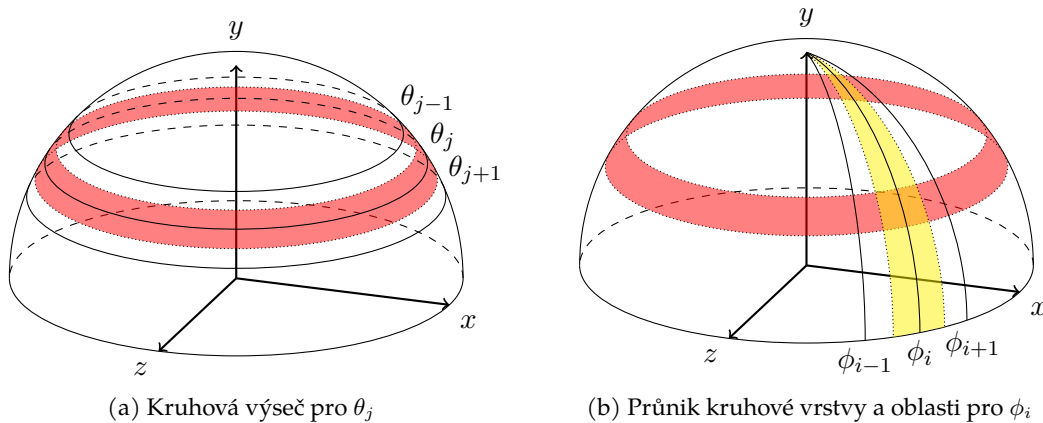
kde  $i$  je index pixelů BRDF textury v ose  $x$ ,  $j$  index v ose  $y$  a  $w_{i,j}$  určuje váhu pro daný bod textury.

Váha pro každý bod BRDF textury je vyjádřena jako velikost prostorového úhlu, který zaujímá daný pixel. Nejdříve se vypočítá povrch kulové vrstvy pro  $\theta_j$ . Tato kulová vrstva je ohraničena pomocí průměrné hodnoty s předchozím  $\theta_{j-1}$  úhlem, resp. následujícím  $\theta_{j+1}$ . Pro krajní případy  $j = 0$  a  $j = j_{max}$  se uvažuje za hranici úhel  $\theta_j$ :

$$S_j = 2\pi \cdot \left[ \left( 1 - \frac{\cos(\theta_j + \theta_{\min\{j+1, j_{max}\}})}{2} \right) - \left( 1 - \frac{\cos(\theta_j + \theta_{\max\{j-1, 0\}})}{2} \right) \right]. \quad (48)$$

Obsah kulové vrstvy se následně stejným způsobem rozdělí na sektory podle úhlu  $\phi_i$  (tj. rovnoměrně při konstantním rozdílu mezi sousedními  $\phi_i$  a  $\phi_{i+1}$ ) a vznikne úseč sféry o obsahu odpovídající váze pro daný bod:

$$w_{i,j} = S_j \cdot \frac{\phi_{\min\{i+1, i_{max}\}} - \phi_{\min\{i-1, 0\}}}{2 \cdot 2\pi} \left( \text{resp.} = \frac{S_j}{i_{max}} \right). \quad (49)$$



Obrázek 35: Rozdělení sféry při výpočtu váhy bodu BRDF

Při výpočtu dochází k výběru určité roviny  $xy$  (dlaždice nebo interpolace mezi dvěma dlaždicemi) z trojrozměrné textury dle  $z$  vypočítané stejným způsobem jako pro bodové světlo. V každé iteraci sumy dochází k převodu souřadnic textury do úhlu dopadu ve světových souřadnicích pomocí inverzní funkce k funkci  $z$  rovnice 46:

$$\vec{v} = T^T \cdot S^T \cdot f_{\phi, \theta}^{-1} \begin{bmatrix} 2\pi x \\ \pi y \end{bmatrix}. \quad (50)$$

Protože je numerická integrace výpočetně náročná, je vhodné ji rozdělit jako výpočet přes více snímků. Vytvořený shader nabízí dvě varianty: Monte-Carlo vzorkování pomocí distribuční funkce rovnoměrného rozložení a deterministický způsob integrace přes každý pixel dlaždice BRDF textury.

### Monte-Carlo integrace

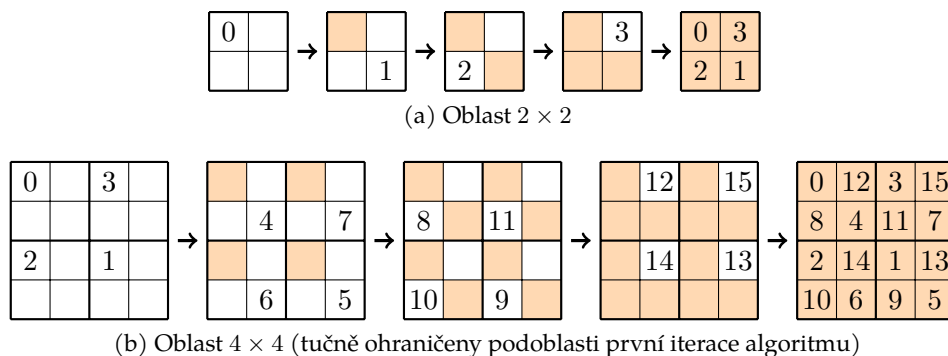
Generování sférických souřadnic vzorků pro Monte-Carlo integraci probíhá pomocí pseudo-náhodného generátoru XorWow<sup>[25]</sup> v každé instanci fragment shaderu izolovaně a pro každou osu sférických souřadnic zvlášť. Počáteční stav generátoru je vytvořen na základě souřadnice fragmentu a pseudonáhodného čísla specifického pro daný snímek pomocí generátoru Split-Mix32.

### Deterministická integrace

Deterministická integrace provádí postupnou iteraci přes celou dlaždici BRDF textury. Výhodou toho způsobu je rychlejší výpočet jednoho vzorku, naopak hlavní nevýhodou je existence systematické chyby zobrazení.

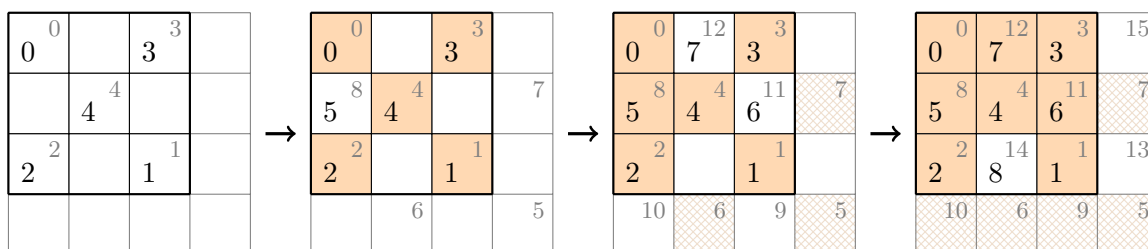
Aby mezisoučet mezi jednotlivými snímky odpovídal přibližnému výsledku, je zde využit princip prokládání. Rovina textury se rozdělí pomocí mřížky na  $k$  stejně velkých oblastí o velikosti  $n \times n$  pixelů. V každé z těchto oblastí se pro daný snímek vybere právě jeden vzorek.

Výběr pozice vzorku v oblasti je založen na rekurzivním algoritmu. Dochází k postupnému rozdělování oblasti na čtyři podoblasti. Podle čísla snímku  $t$  (číslovaného od 0) se vybírá aktivní podoblast do té doby, než obsahuje právě jeden vzorek (tj. velikost podoblasti je  $1 \times 1$ ). O výběru podoblasti rozhoduje zbytek po dělení 4: 0 – levý dolní roh, 1 – pravý horní roh, 2 – levý horní roh, 3 – pravý dolní roh.



Obrázek 36: Pořadí výběru vzorku v oblasti

Pro oblasti, pro které nemají velikost odpovídající mocnině dvou, se vytvoří fiktivní oblast o velikosti hrany nejbližší větší mocnině dvou. Dále je nutné vytvořit fiktivní číslo snímků, které je inkrementováno spolu s reálným číslem snímku. V situacích, kdy vybraný vzorek fiktivní oblasti mimo oblast reálnou, dochází k zvýšení fiktivního čísla o 1 a opakování hledání.



Obrázek 37: Pořadí vzorků u oblasti  $3 \times 3$  (tučně ohraničena) s fiktivní oblastí  $4 \times 4$ . V levém dolním rohu číslo snímku, v pravém horním rohu fiktivní číslo snímku.

## 8 Vykrasovací řetězce herních enginů

### 8.1 Unity

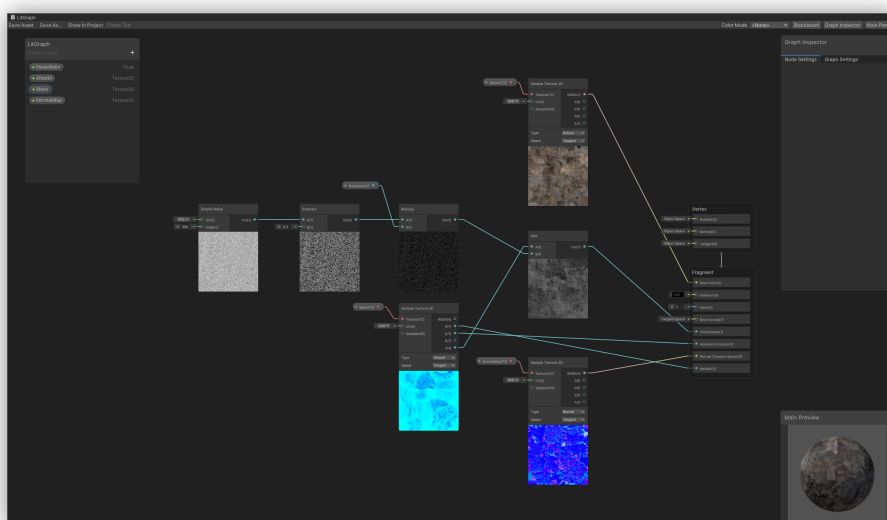
Unity v současné době poskytuje tři různé vykreslovací řetězce: Vestavěný, Universal Render Pipeline (URP) a High Definition Render Pipeline (HDRP). Poslední dva ze zmiňované trojice jsou postaveny na Scriptable Render Pipeline. Jedná se o mezivrstvu v jazyce C#, pomocí které lze ovládat proces vykreslování.<sup>[26]</sup>

Pro výpočet odrazivosti používá Unity upravený Walterův model s Schlickovou aproximací Fresnelova vzorce a Disney difúzním modelem. Na část výpočtů jsou využity předintegrované hodnoty BRDF. Nepřímé osvětlování je vytvořeno pomocí kombinace L2 sférických harmonických funkcí nebo předvypočítané hodnoty (v případě statických objektů) pro difúzní složku a lokalizované mapy prostředí pro spekulární složku, popřípadě pomocí techniky zvané Screen Space Reflection.<sup>[4]</sup>

#### 8.1.1 Způsoby tvorby shaderů

Tvorba shaderů je v Unity možná dvěma způsoby: pomocí ShaderLab kódu nebo graficky v Shader Graph editoru. ShaderLab je nadstavba nad programovacím jazykem HLSL. Umožňuje kromě samotného kódu shaderu (vertex, fragment, apod.) definovat parametry viditelné v grafickém prostředí editoru a měnit chování při vykreslování (například způsob nakládání s Z-bufferem, míchání barev, vyzařování stěn dle orientace, prioritu ve frontě vykreslování).<sup>[27]</sup> Díky tomu je možné napřímo určovat výsledné hodnoty pixelů vykresleného obrazu. Oficiálně je doporučován pouze pro vestavěný vykreslovací řetězec, přestože je vnitřně používán u všech řetězců, a to kvůli komplexnosti vykreslovacího procesu a neexistenci dokumentace v případě URP a HDRP.<sup>[28]</sup>

Shader Graph je nástroj na vizuální programování shaderů, pomocí kterého lze měnit parametry předdefinovaných vykreslovacích modelů. Jedná se o acyklický graf skládající se z jednotlivých bloků (vrcholů). Vrcholy představují specifickou operaci (manipulace s texturami, matematickou operaci, atd.) nebo vstupní data (informace ohledně geometrie, externí parametry v podobě skalárů, vektorů a textur). Jednotlivé vrcholy jsou spojeny hranami vedoucí z výstupu jednoho vrcholu do vstupu druhého. Tento nástroj je podporován všemi vykreslovacími řetězci. Hlavní nevýhodou je izolace od samotného procesu vykreslování, kdy není možné měnit předdefinované osvětlovací modely ani získat informace o zdrojích světla.<sup>[29]</sup>



Obrázek 38: Nástroj Shader Graph v Unity

### 8.1.2 Možnosti implementace vlastní BRDF

V případě použití vestavěného vykreslovací řetězce lze bez potíží vytvořit fragment shader využívající vlastní funkci zajišťující výpočet BRDF. Pro URP a HDRP není možné tento postup napřímo aplikovat. Lze ale využít vnitřního fungování nástroje Shader Graph, který pro následné vykreslování generuje ShaderLab kód. Díky okolnosti, že jsou zdrojové kódy shaderů používané v Unity veřejně dostupné, je možné izolovat funkce zajišťující výpočet BRDF a následně je nahradit vlastní implementací. Celý proces lze navíc plně automatizovat pomocí skriptů rozšiřující samotný editor.<sup>[30]</sup>

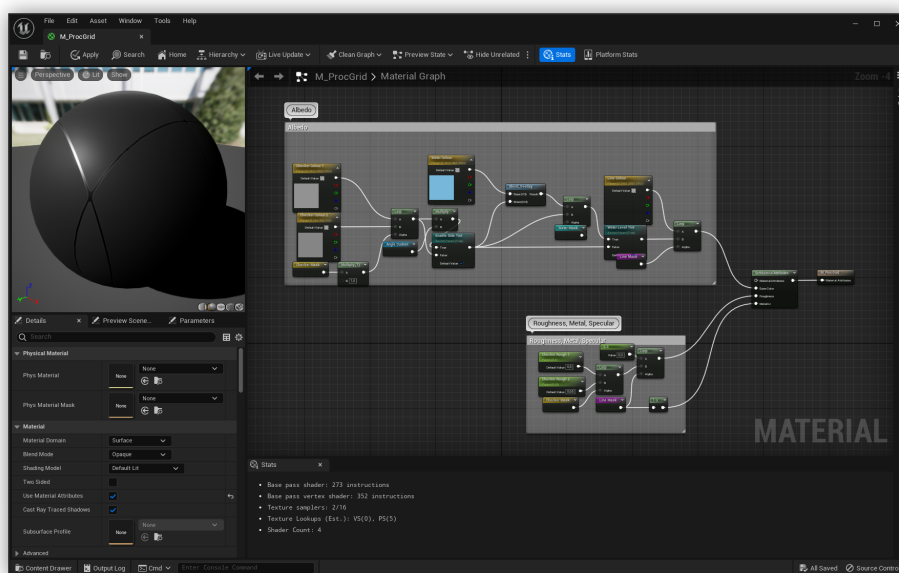
## 8.2 Unreal Engine 5

Unreal Engine primárně pro výpočet osvětlení využívá techniku odloženého vykreslování, které se skládá ze dvou fází. V první fázi nedochází k výpočtu osvětlení, ale pouze k naplnění tzv. G-bufferu. Jedná se o vyrovnávací paměť, do které se pro každý fragment obrazu uloží informace potřebné pro výpočet BRDF v daném bodě (tj. pozice, normála, typ BRDF modelu, hodnota albeda a jiné parametry analytických modelů). V druhé fázi dochází k výpočtu osvětlení pro každý fragment a již není potřebná znalost geometrie scény nad rámec uložených dat v G-bufferu. Seznam podporovaných BRDF modelů musí být předem globálně definován.

Výchozím BRDF model používaným v Unreal Engineu je upravený Walterův model s Schlickovu aproximací Fresnelova vzorce a aproximovanou variantou geometrického útlumu.<sup>[31]</sup> Pro výpočet nepřímého osvětlování se používá technologie Lumen, kombinující techniku Screen Space Reflection a Mesh Distance Field s mezipamětí povrchů.<sup>[32]</sup> Technologie Lumen používá nezávisle na vybraném BRDF modelu přímého osvětlování zjednodušený Walterův model bez Fresnelova členu a geometrického útlumu.<sup>[31]</sup>

### 8.2.1 Způsoby tvorby shaderů

Unreal Engine poskytuje pro tvorbu shaderů pro materiály pouze nástroj Material Editor podobný nástroji Shader Graph v Unity. Pomocí tohoto nástroje lze změnit chování první fáze odloženého vykreslování a lze nastavovat parametry G-bufferu. Z tohoto důvodu nelze získat informace o umístění zdrojů světla a měnit chování BRDF.<sup>[33]</sup>



Obrázek 39: Nástroj Material Editor v Unreal Engine



## 8.2.2 Možnosti implementace vlastní BRDF

Pro úpravu chování odloženého vykreslování z hlediska BRDF je nutné změnit druhou fázi vykreslování. Ta je pevně určena ve zdrojových kódech samotného enginu. Upravit zdrojové kódy je možné, protože jsou dostupné pod speciální licenci v neveřejné skupině na platformě GitHub.<sup>[31]</sup> Pro přidání nového BRDF modelu je nutné přidat novou položku do seznamu podporovaných modelů, zpřístupnit výběr v uživatelském rozhraní a implementovat výpočet BRDF jako součást funkce `IntegrateBxDF` v zdrojovém souboru shaderu `ShadingModels.ush`.<sup>[34]</sup>

Protože do G-bufferu není možné ukládat přímo celé textury, je nutné předávat do druhé fáze vykreslování pouze index atlasu BRDF textur (viz kapitola 6.2.6), který je centrálně definovaný nezávisle na jednotlivých objektech scény. Z toho plyne omezení v podobě maximálního počtu různých naměřených vzorků BRDF, které lze ve scéně použít.



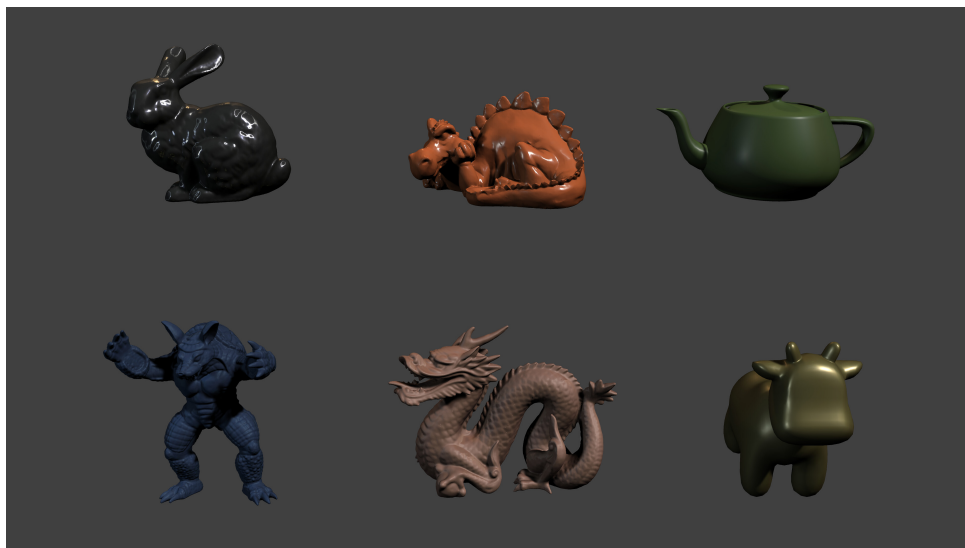
## 9 Použití měřených dat v herních enginech

V herních enginech Unity a Unreal Engine 5.1.1 byl v rámci práce vytvořen shader využívající naměřená BRDF data dle kapitoly 7 a shader s Lafortune-Phong analytickým modelem, použitým jako reference k měření rychlosti vykreslování.

V obou herních enginech byla vytvořena totožná testovací scéna (viz obrázky 40 a 42) skládající se z několika 3D modelů, každý z jiným vzorkem naměřených BRDF dat. Konkrétně se jedná o modely Stanford Bunny<sup>[35]</sup>, Stanford Armadillo<sup>[35]</sup>, Phlegmatic Dragon<sup>[36]</sup>, Stanford Dragon<sup>[35]</sup>, Utah Teapot<sup>[37]</sup> a Spot<sup>[38]</sup>. Rozdíl mezi zobrazením testovací scény v Unity a Unreal Engine je způsoben primárně odlišným tónovým mapováním.

### 9.1 Unity

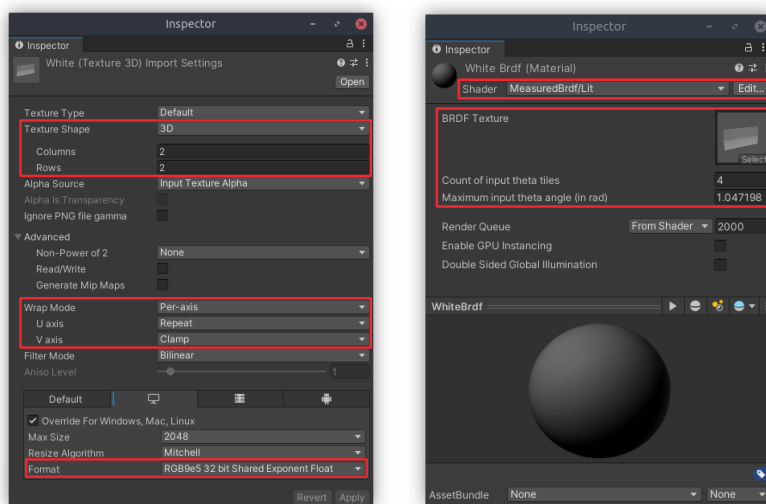
Implementovaný shader pro Unity je určen pro vestavěný vykreslovací řetězec. Podporuje pouze přímé zdroje osvětlení (směrové, bodové a kuželová světla), nepodporuje nepřímé osvětlování z map okolí ani pomocí sférických harmonických funkcí, protože tyto operace by vyžadovaly numerickou integraci nebo komplikované operace v prostoru sférických harmonických funkcí.



Obrázek 40: Vykreslená testovací scéna z Unity

Pro vytvoření příslušného materiálu je nutné importovat naměřená data ve formátu EXR. U této textury je nutné upravit nastavení importu, aby se jednalo o 3D texturu (Texture Shape), správně vyplnit počet řádků a sloupců mřížky (Rows, Columns) a vybrat formát (Format v sekci nastavení, která jsou závislá na platformě). Je nutno vybrat formát, který podporuje ukládání dat s vysokým dynamickým rozsahem (RGBA Half, RGB9e5 32bit Shared Exponent Float). Bohužel formát RGB HDR Compressed BC6H není možné použít, protože podporuje pouze textury s rozlišením dělitelným 4, tudíž pro exportovaná data není použitelný. Posledním parametrem, který je nutné změnit, je typ chování při čtení z textury mimo její rozsah (Wrap Mode): Osa U na Repeat a osa V na Clamp.

V nastavení materiálu je potřeba zvolit shader MeasuredBrdf • Lit. Poté se zobrazí možnosti shaderu v podobě výběru textury měřených dat a číselných polí pro určení počtu dlaždic textury a úhlu dopadu poslední dlaždice v radiánech. Číselná pole podporují zápis matematických výrazů, např. „60/180\*pi“.



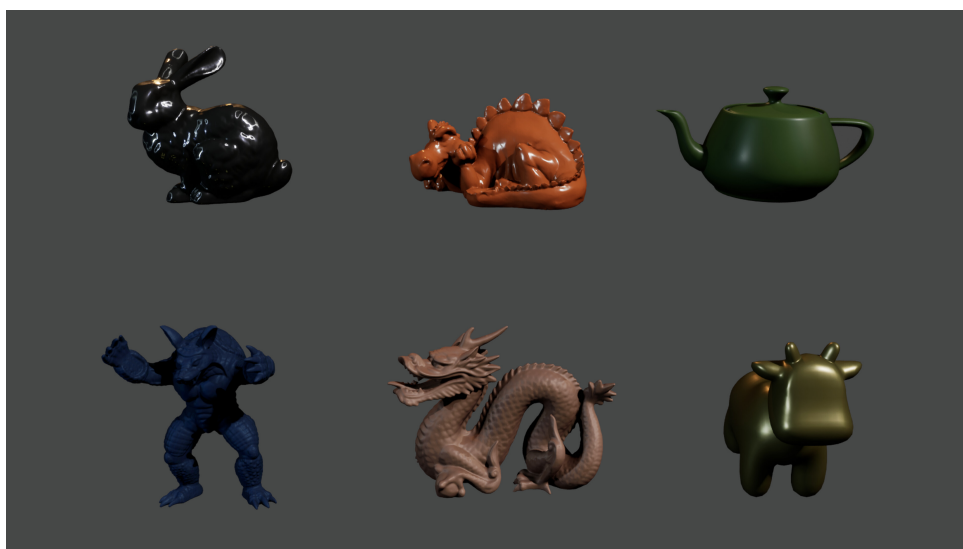
(a) Nastavení importu

(b) Nastavení materiálu

Obrázek 41: Import 3D BRDF textury v Unity

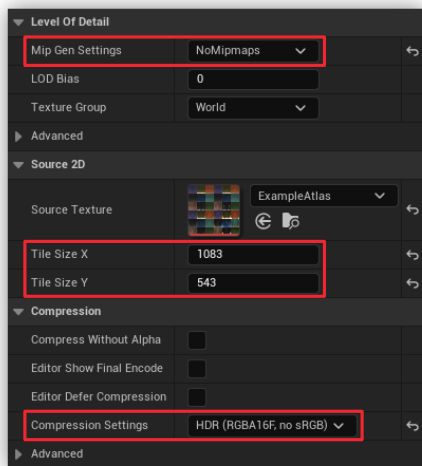
## 9.2 Unreal Engine

Implementovaný shader pro Unreal Engine 5.1.1 je součástí vnitřního systému odloženého vykreslování. Stejně jako v Unity, i zde je podporováno pouze přímé osvětlování (vyjma obdélníkových zdrojů světla). Chování nepřímého osvětlování technologií Lumen nebylo možné upravit, protože využívá zjednodušený BRDF model nezávisle na vybraném modelu pro přímé vykreslování (viz kapitola 8.2.2).

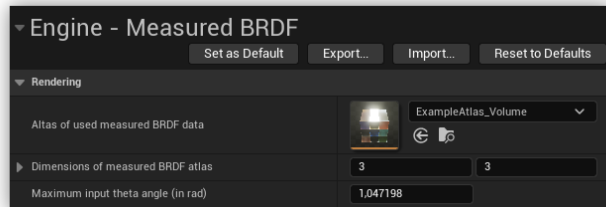


Obrázek 42: Vykreslená testovací scéna z Unreal Engine

Nejdříve je nutné vytvořit atlas jednotlivých naměřených BRDF vzorků pomocí volání aplikace `brdf_gui` z příkazové řádky (viz kapitola 6.2.7). Z importovaného atlasu je potřeba v Unreal Engine vytvořit 3D texturu (Volume Texture) a změnit následující nastavení importu: Určit velikost vnější dlaždice (Tile Size X, Tile Size Y), vypnout vytváření zmenšených variant textury (Mip Gen Settings na NoMipmaps) a nastavit formát komprese, který podporuje ukládání dat s vysokým dynamickým rozsahem (HDR (RGBA16F, no sRGB), HDR High Precision (RGBA32F) nebo HDR Compressed (RGB, BC6H, DX11)).



(a) Nastavení importu

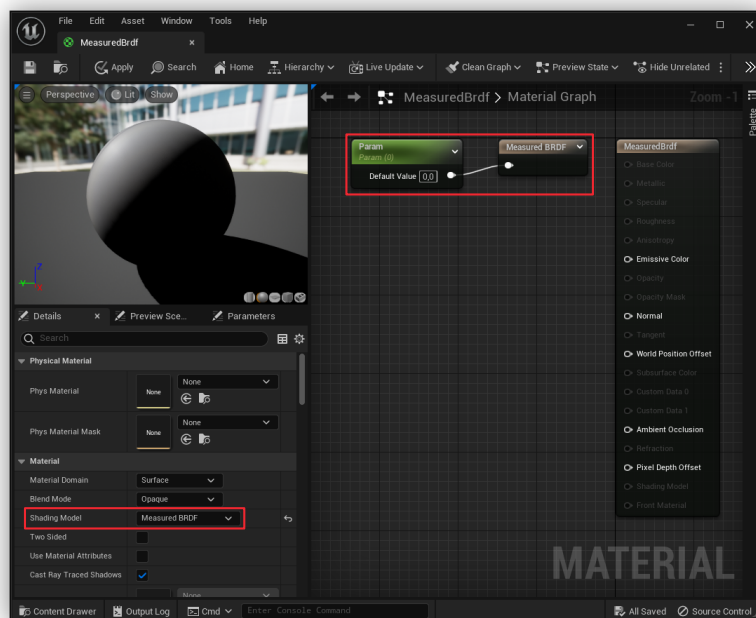


(b) Nastavení projektu

Obrázek 43: Nastavení importu atlasu v Unreal Engine

Následně je potřeba 3D texturu atlasu přiřadit v nastavení projektu (sekce Engine ▶ Measured BRDF), společně s počty sloupců a řádků v mřížce vzorků, aby bylo možné ji použít při vykreslování.

V editoru materiálu, v sekci Material panelu Detail se nachází výběr aktivního modelu. Mezi vestavěnými možnostmi se nachází položka Measured BRDF. Při výběru této možnosti se přepne materiál do módu využívající měřená BRDF data. Pro správné fungování je nutné vytvořit nový vrchol Measured BRDF přijímající skalární hodnotu a do tohoto vstupu připojit parametr obsahující celočíselnou hodnotu indexu vzorku z atlasu (číslované od 0). Uživatelsky není kontrolováno, jestli se jedná o platnou hodnotu, tj. celé číslo v rozsahu 0 do  $n - 1$ , kde  $n$  je počet vzorků atlasu.



Obrázek 44: Nastavení materiálu v Unreal Engine

### 9.3 Porovnání rychlosti vykreslování

Vytvořené shadery využívající naměřená BRDF data byla porovnána v prostředí Unity 2022.1.19f1 a Unreal Engine 5.1.1. Test byl proveden na zařízení s procesorem Intel i5-9600K @ 4.5 GHz, grafickou kartou nVidia GeForce GTX 1660. Oba herní enginy pro vykreslování využívají technologii Vulkan.

Každý shader byl porovnán s vytvořenou implementací Lafortune-Phong modelu v daném prostředí. Pro porovnání záměrně nebyl vybrán výchozí model daného herního enginu, protože na rozdíl od implementovaných shaderů podporuje výpočet nepřímého osvětlování.

Výsledky porovnání jsou uvedeny v tabulce 1. V obou prostředích je patrný rozdíl mezi shaderem využívající naměřená data a analytickým modelem Lafortune-Phong. Nicméně se nejedná o významný rozdíl, pouze v řádech jednotek procent, přičemž model Lafortune-Phong vychází vždy rychlejší.

	Lafortune-Phong		Naměřená BRDF	
Unity 2022.1.19f1	102,9 fps	9,72 ms	99,5 fps	10,05 ms
Unreal Engine 5.1.1	82,5 fps	12,12 ms	80,9 fps	12,37 ms

Tabulka 1: Porovnání rychlosti vykreslování

## 10 Závěr

Předmětem této práce bylo se seznámit s problematikou BRDF. Pomocí scatterometru MiniDiff v2 od společnosti Synopsys (dříve Lightec) naměřit odrazivost materiálů, následně tyto data zpracovat a využít pro vykreslování v herních enginech Unity a Unreal Engine.

Přístrojem MiniDiff v2 byla naměřena odrazivost u 217 vzorků. Výběr vzorků bylo nutné přizpůsobit existujícím omezením přístroje.

Speciálně pro využití naměřených dat byla vytvořena knihovna `brdf_parser` a grafické uživatelské rozhraní `brdf_gui` v jazyce C++ pomocí knihovny `Gtkmm` a `OpenGL`. Naměřená data byla pomocí této aplikace sloučena, zpracována a uložena do obrazového formátu `EXR` vhodného pro použití v herních enginech. Dále byly pomocí tohoto nástroje vytvořeny krátká videa ilustrující použití naměřených dat na zobrazeném 3D modelu.

V herních enginech Unity byl naprogramován shader využívající naměřená data pro výpočet přímého osvětlování. V prostředí Unreal Engine, kromě samotného shaderu, bylo nutné upravit vnitřní chování zobrazovacího řetězce. Rychlost vykreslování pomocí naprogramovaných shaderů byla porovnána s rychlostí vykreslování pomocí `Lafortune-Phong` BRDF modelu.





## Zdroje

1. NICODEMUS, F.E.; RICHMOND, J.C.; HSIA, J.J.; GINSBERG, W.I.; LIMPERS, T. Geometrical considerations and nomenclature for reflectance. *Applied Optics*. 1977, roč. 9, s. 1474–1475. Dostupné také z: <https://graphics.stanford.edu/courses/cs448-05-winter/papers/nicodemus-brdf-nist.pdf>.
2. SOLDADO, Rosana Montes; ALMAGRO, Carlos Ureña. *An Overview of BRDF Models*. 2012. Dostupné také z: <http://hdl.handle.net/10481/19751>.
3. SYNOPSISYS. *MiniDiff V2 User's Manual*. 2021. Dostupné také z: <https://www.synopsys.com/optical-solutions/scattering-measurements.html#MiniDiffV2>.
4. UNITY TECHNOLOGIES. *Unity Graphics Git Repository* [online]. [cit. 2022-12-27]. Dostupné z: <https://github.com/Unity-Technologies/Graphics/tree/1337d579a6e4aee397a811bdcfd17273ae1e9edb>.
5. GUARNERA, D.; GUARNERA, G.C.; GHOSH, A.; DENK, C.; GLENCROSS, M. BRDF Representation and Acquisition. *Computer Graphics Forum*. 2016, roč. 35, č. 2, s. 625–650. Dostupné z doi: <https://doi.org/10.1111/cgf.12867>.
6. PHONG, Bui Tuong. Illumination for Computer Generated Pictures. *Commun. ACM*. 1975, roč. 18, č. 6, s. 311–317. ISSN 0001-0782. Dostupné z doi: 10.1145/360825.360839.
7. LAFORTUNE, Eric; WILLEMS, Yves. *Using the Modified Phong Reflectance Model for Physically Based Rendering*. 1994. Dostupné také z: <https://www.cs.princeton.edu/courses/archive/fall103/cs526/papers/lafortune94.pdf>.
8. WARD, Gregory J. Measuring and Modeling Anisotropic Reflection. *SIGGRAPH Comput. Graph*. 1992, roč. 26, č. 2, s. 265–272. ISSN 0097-8930. Dostupné z doi: <https://doi.org/10.1145/142920.134078>.
9. COOK, R. L.; TORRANCE, K. E. A Reflectance Model for Computer Graphics. *ACM Trans. Graph*. 1982, roč. 1, č. 1, s. 7–24. ISSN 0730-0301. Dostupné z doi: 10.1145/357290.357293.
10. SCHLICK, Christophe. An Inexpensive BRDF Model for Physically-based Rendering. *Computer Graphics Forum*. 1994, roč. 13, č. 3, s. 233–246. Dostupné z doi: <https://doi.org/10.1111/1467-8659.1330233>.
11. OREN, Michael; NAYAR, Shree K. Generalization of Lambert's Reflectance Model. In: *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques*. New York, NY, USA: Association for Computing Machinery, 1994, s. 239–246. SIGGRAPH '94. ISBN 0897916670. Dostupné z doi: 10.1145/192161.192213.
12. WALTER, Bruce; MARSCHNER, Stephen R.; LI, Hongsong; TORRANCE, Kenneth E. Microfacet Models for Refraction through Rough Surfaces. 2007, s. 195–206. ISBN 9783905673524. Dostupné také z: <https://dl.acm.org/doi/10.5555/2383847.2383874>.
13. MATUSIK, Wojciech; PFISTER, Hanspeter; BRAND, Matt; MCMILLAN, Leonard. A Data-Driven Reflectance Model. *ACM Transactions on Graphics*. 2003, roč. 22, č. 3, s. 759–769. Dostupné také z: <https://www.merl.com/brdf/>.
14. BURLEY, Brent. *Physically-based shading at Disney*. 2012. Dostupné také z: <https://disneyanimation.com/publications/physically-based-shading-at-disney/>.

15. ZEMAX LLC. *BSDF Data Interchange file format specification* [online]. 2021. [cit. 2022-11-01]. Dostupné z: <https://support.zemax.com/hc/en-us/articles/1500005486801-BSDF-Data-Interchange-file-format-specification>.
16. LANEVSKI, Dmitri; FERRERO, Alejandro; PERALES, Esther; MANOOCHERI, Farshid; IKONEN, Erkki. Machine-readable universal data format for bidirectional reflectance distribution function and BiRDview—An open-source web-based application for viewing and comparing bidirectional reflectance data. *Color Research & Application*. 2022, roč. 47, č. 5, s. 1177–1192. Dostupné z DOI: <https://doi.org/10.1002/col.22790>.
17. OPENEXR CONTRIBUTORS. *OpenEXR* [online]. [cit. 2023-04-19]. Dostupné z: <https://github.com/AcademySoftwareFoundation/openexr/tree/v3.1.7>.
18. SHEPARD, Donald. A two-dimensional interpolation function for irregularly-spaced data. 1968. Dostupné z DOI: 10.1145/800186.810616.
19. SAMET, Hanan. The Quadtree and Related Hierarchical Data Structures. *ACM Comput. Surv.* 1984, roč. 16, č. 2, s. 187–260. ISSN 0360-0300. Dostupné z DOI: 10.1145/356924.356930.
20. ANDERSON, D. P. Techniques for Reducing Pen Plotting Time. *ACM Trans. Graph.* 1983, roč. 2, č. 3, s. 197–212. ISSN 0730-0301. Dostupné z DOI: 10.1145/357323.357327.
21. MOUNT, David M.; ARYA, Sunil. *ANN: A Library for Approximate Nearest Neighbor Searching* [online]. Department of Computer Science a Institute for Advanced Computer Studies University of Maryland, College Park, Maryland, 2010-01 [cit. 2023-04-19]. Dostupné z: <http://www.cs.umd.edu/~mount/ANN/>.
22. THE GNOME PROJECT AND CONTRIBUTORS. *Gtkmm: C++ Interfaces for GTK and GNOME* [online]. [cit. 2023-04-19]. Dostupné z: <https://gtkmm.org/>.
23. INTERNATIONAL TELECOMMUNICATION UNION. *Parameter values for the HDTV standards for production and international programme exchange* [online]. 2015. [cit. 2023-04-19]. ITU-R Recommendation, BT.709-6. ITU-R. Dostupné z: [https://www.itu.int/dms\\_pubrec/itu-r/rec/bt/R-REC-BT.709-6-201506-I!!PDF-E.pdf](https://www.itu.int/dms_pubrec/itu-r/rec/bt/R-REC-BT.709-6-201506-I!!PDF-E.pdf).
24. OPENGL WIKI. *Shader* [online]. 2019. [cit. 2023-05-16]. Dostupné z: <http://www.khronos.org/opengl/wiki/opengl/index.php?title=Shader&oldid=14593>.
25. MARSAGLIA, George. Xorshift RNGs. *Journal of Statistical Software*. 2003, roč. 8, č. 14. Dostupné z DOI: 10.18637/jss.v008.i14.
26. UNITY TECHNOLOGIES. *Unity Render Pipelines Documentation* [online]. [cit. 2023-01-18]. Dostupné z: <https://docs.unity3d.com/Manual/render-pipelines.html>.
27. UNITY TECHNOLOGIES. *ShaderLab Documentation* [online]. [cit. 2023-01-18]. Dostupné z: <https://docs.unity3d.com/Manual/SL-Reference.html>.
28. UNITY TECHNOLOGIES. *Render pipeline feature comparison* [online]. [cit. 2022-12-27]. Dostupné z: <https://docs.unity3d.com/2022.2/Documentation/Manual/render-pipelines-feature-comparison.html>.
29. UNITY TECHNOLOGIES. *Shader Graph Documentation* [online]. [cit. 2023-01-18]. Dostupné z: <https://docs.unity3d.com/Packages/com.unity.shadergraph@12.1/manual/index.html>.
30. ZGEB, Bronson. *Custom Lighting in URP with Shader Graph* [online]. 2021. [cit. 2022-12-26]. Dostupné z: <https://bronsonzgeb.com/index.php/2021/10/04/custom-lighting-in-urp-with-shader-graph/>.

31. EPIC GAMES, INC. *Downloading Unreal Engine Source Code* [online]. [cit. 2023-04-25]. Dostupné z: <https://docs.unrealengine.com/5.1/en-US/downloading-unreal-engine-source-code/>.
32. EPIC GAMES, INC. *Lumen Technical Details* [online]. [cit. 2023-04-25]. Dostupné z: <https://docs.unrealengine.com/5.0/en-US/lumen-technical-details-in-unreal-engine/>.
33. EPIC GAMES, INC. *Material Editor User Guide* [online]. [cit. 2023-04-25]. Dostupné z: <https://docs.unrealengine.com/5.1/en-US/unreal-engine-material-editor-user-guide/>.
34. ONE3Y3. *New shading models and changing the GBuffer* [online]. 2022-12. [cit. 2023-04-25]. Dostupné z: <https://dev.epicgames.com/community/learning/tutorials/2R5x/unreal-engine-new-shading-models-and-changing-the-gbuffer>.
35. STANFORD COMPUTER GRAPHICS LABORATORY. *The Stanford 3D Scanning Repository* [online]. [cit. 2023-05-20]. Dostupné z: <http://graphics.stanford.edu/data/3Dscanrep>.
36. FILIP, Jiří; HOLUB, Radek; HAVRAN, Vlastimil; KŘIVÁNEK, Jaroslav; SÝKORA, Daniel. *Phlegmatic Dragon* [online]. 2007. [cit. 2023-05-20]. Dostupné z: <https://dcgi.fel.cvut.cz/cgg/eg07/index.php?page=dragon>.
37. NEWELL, Martin. *Utah Teapot* [online]. 1975. [cit. 2023-05-20]. Dostupné z: <https://graphics.cs.utah.edu/teapot/>.
38. CRANE, Keenan. *Spot* [online]. [cit. 2023-05-20]. Dostupné z: <https://www.cs.cmu.edu/~kmcrane/Projects/ModelRepository/>.

## Seznam obrázků

1	Příklad prostorového úhlu $1\ sr$ . . . . .	1
2	Příklad záře . . . . .	2
3	Příklad kartézských a sférických souřadnic . . . . .	3
4	Odrazivost Lambertova modelu . . . . .	5
5	Ideální odraz . . . . .	6
6	Příklad geometrického útlumu . . . . .	8
7	Příklad souřadnicového systému formátu firmy Zemax . . . . .	11
8	Příklad Rusinkiewiczova souřadnicový systému . . . . .	12
9	Přístroj MiniDiff v2 . . . . .	13
10	Snímek okna softwaru Minidiff v2 . . . . .	13
11	Extrapolace Beziérovou křivkou v software MiniDiff v2 . . . . .	14
12	Přístroj MiniDiff v2 v průběhu měření . . . . .	15
13	Nevalidní měření způsobené zvlněním měřeného vzorku . . . . .	16
14	Nevhodné typy vzorků . . . . .	17
15	Fotografie naměřených vzorků . . . . .	17
15	Fotografie naměřených vzorků (pokračování) . . . . .	18
16	Souřadnicový systém v dlaždici . . . . .	19
17	Pořadí indexace dlaždic . . . . .	20
18	Rozdělení hemisféry na oblasti interpolace, extrapolace odrazu a dopadu . . . . .	20
19	Obvyklé zřetězení tříd . . . . .	21
20	Znázornění extrapolace gradientem . . . . .	22
21	Strom vyhledávací struktury . . . . .	23
22	Průběh hledání $k = 2$ nejbližších sousedů . . . . .	24
23	Porovnání ANN a implementované datové struktury pro různá $m$ . . . . .	24
24	Hlavní okno aplikace brdf_gui . . . . .	25
25	Postranní panel . . . . .	26
26	Zobrazení laloku . . . . .	27
27	Zobrazení modelu . . . . .	28
28	Nastavení vykreslování . . . . .	29
29	Nastavení interpolace . . . . .	30
30	Nastavení extrapolace . . . . .	31
31	Okno slučování barevných složek . . . . .	31
32	Pořadí indexace vnitřních dlaždic atlasu . . . . .	32
33	Porovnání lineární interpolace dlaždic . . . . .	33
34	Tečný prostor využívaný při vykreslování . . . . .	34
35	Rozdělení sféry při výpočtu váhy bodu BRDF . . . . .	35
36	Pořadí výběru vzorku v oblasti . . . . .	36
37	Pořadí vzorků u oblasti $3 \times 3$ s fiktivní oblastí $4 \times 4$ . . . . .	36
38	Nástroj Shader Graph v Unity . . . . .	37

39	Nástroj Material Editor v Unreal Engine . . . . .	38
40	Vykreslená testovací scéna z Unity . . . . .	41
41	Import 3D BRDF textury v Unity . . . . .	42
42	Vykreslená testovací scéna z Unreal Engine . . . . .	42
43	Nastavení importu atlasu v Unreal Engine . . . . .	43
44	Nastavení materiálu v Unreal Engine . . . . .	43
45	Model draka – použití naměřených dat od vybraných vzorků . . . . .	56
45	Model draka – použití naměřených dat od vybraných vzorků (pokračování) . . . . .	57

## Seznam tabulek

1	Porovnání rychlosti vykreslování . . . . .	44
2	Seznam kovových vzorků . . . . .	53
3	Seznam papírových vzorků . . . . .	53
4	Seznam plastových vzorků . . . . .	53
5	Seznam gumových vzorků . . . . .	54
6	Seznam plstěných vzorků . . . . .	54
7	Seznam vzorků vzorníku PPG . . . . .	54
8	Seznam vzorků metalického vzorníku . . . . .	54
9	Seznam vzorků vzorníku NCS . . . . .	55
10	Seznam mix vzorků . . . . .	55



# Přílohy

## A Seznam naměřených vzorků

### Kovové vzorky

kov-hrnec.exr	kov-hrnec_2.exr	kov-krabice.exr
kov-krabice_2.exr	kov-krabice-modra.exr	kov_matny.exr
kov_pseudozlaty.exr	kov_pseudozlaty_2.exr	kov-spachtle.exr
kov-spachtle_2.exr	kov-viko.exr	kov-viko_2.exr
kov_zeleny.exr	plech_bily.exr	spachtle40mm.exr
spachtle40mm_2.exr	spachtle50mm.exr	spachtle50mm_2.exr
spachtle-leskla.exr	spachtle-leskla_2.exr	paska-medena.exr
paska-medena_2.exr		

Tabulka 2: Seznam kovových vzorků (22 ks)

### Papírové vzorky

fotopapir.exr	karton-svetly.exr	papir120.exr
karton.exr	papir-zluta.exr	papir-modra.exr
papir-hneda.exr	papir-cerna.exr	papir-zelena.exr
papir-sediva.exr	papir-ruzova.exr	papir-hneda2.exr
papir-bezova.exr	papir-fialova.exr	papir-cervena.exr
papir-oranzova.exr	papir-tyrkysova.exr	papir-modrozelená.exr
papir-zluta_leskla.exr	papir-svetle-zluta.exr	papir-svetle-modra.exr
papir-svetle-hneda.exr	papir-modra_leskla.exr	papir-zelena_leskla.exr
papir-svetle-zelena.exr	papir-svetle-sediva.exr	papir-svetle-bezova.exr
papir-svetle-fialova.exr	papir-svetle-cervena.exr	papir-cervena_leskla.exr
papir-oranzova_leskla.exr	papir-modrozelená_leskla.exr	papir-bila.exr
papir-sediva1.exr	papir-sediva2.exr	papir-sediva3.exr
papir-sediva4.exr	papir-sediva5.exr	papir-tmave-cervena.exr
papir-tmave-zelena.exr		

Tabulka 3: Seznam papírových vzorků (40 ks)

### Plastové vzorky

plast-cerny.exr	plast-modry.exr	polystyren.exr
polystyren_2.exr	pytlík-stříbrny.exr	pytlík-stříbrny_2.exr
plast-zeleny.exr	plast-zlutý.exr	plast-cerveny.exr
plast-cerny-leskly.exr	plast-bily.exr	plast-svetle-modry.exr

Tabulka 4: Seznam plastových vzorků (12 ks)

## Gumové vzorky

guma-bila.exr	guma-cerna.exr	guma-cervena.exr	guma-fialova.exr
guma-hneda.exr	guma-modr.exr	guma-oranzova.exr	guma-sediva.exr
guma-tmave-zelena.exr	guma-tyrkysova.exr	guma-zelena.exr	guma-zluta.exr

Tabulka 5: Seznam gumových vzorků (12 ks)

## Plstěné vzorky

plst-bila.exr	plst-cervena.exr	plst-fialova.exr
plst-hneda.exr	plst-oranzova.exr	plst-svetle-modra.exr
plst-svetle-zelena.exr	plst-tmave-modra.exr	plst-zluta.exr

Tabulka 6: Seznam plstěných vzorků (9 ks)

## Vzorník PPG

ppg1222-5.exr	ppg1223-5.exr	ppg1224-5.exr	ppg1225-5.exr	ppg1226-5.exr
ppg1227-5.exr	ppg1228-5.exr	ppg1229-5.exr	ppg1230-5.exr	ppg1231-5.exr
ppg1232-5.exr	ppg1233-5.exr	ppg1234-5.exr	ppg1235-5.exr	ppg1236-5.exr
ppg1237-5.exr	ppg1238-5.exr	ppg1239-5.exr	ppg1240-5.exr	ppg1241-5.exr
ppg1242-5.exr	ppg1243-5.exr	ppg1244-5.exr	ppg1245-5.exr	ppg1246-5.exr
ppg1247-5.exr	ppg1248-5.exr	ppg1249-5.exr	ppg1250-5.exr	ppg1251-5.exr
ppg1252-5.exr	ppg1181_5.exr	ppg1182_5.exr	ppg1183_5.exr	ppg1184_5.exr
ppg1185_5.exr	ppg1186_5.exr	ppg1187_5.exr	ppg1188_5.exr	ppg1189_5.exr
ppg1190_5.exr	ppg1191_5.exr	ppg1192_5.exr	ppg1193_5.exr	ppg1194_5.exr
ppg1195_5.exr	ppg1196_5.exr	ppg1197_5.exr	ppg1198_5.exr	ppg1199_5.exr
ppg1200_5.exr	ppg1201_5.exr	ppg1202_5.exr	ppg1203_5.exr	ppg1204_5.exr
ppg1205_5.exr	ppg1206_5.exr	ppg1207_5.exr	ppg1208_5.exr	ppg1209_5.exr
ppg1210_5.exr	ppg1211_5.exr	ppg1212_5.exr	ppg1213_5.exr	ppg1214_5.exr
ppg1215_5.exr	ppg1216_5.exr	ppg1217_5.exr	ppg1218_5.exr	ppg1219_5.exr
ppg1220_5.exr	ppg1221_5.exr	ppg1227_5.exr	ppg1252_5.exr	

Tabulka 7: Seznam vzorků vzorníku PPG (74 ks)

## Metalický vzorník

H19-YC14.exr	H19-YC21.exr	H19-YC30.exr	H19-YC31.exr	H19-YC32.exr
H19-YC33.exr	H19-YC34.exr	H19-YC36.exr	H19-YC38.exr	H19-YC39.exr
H19-YC44.exr	H19-YC45.exr	H19-YC50.exr	H19-YC51.exr	H19-YC52.exr
H19-YC80.exr				

Tabulka 8: Seznam vzorků metalického vzorníku (16 ks)



## Vzorník NCS

S0580-Y30R.exr	S0580-Y90R.exr	S0850-E40B.exr	S1070-G50Y.exr
S1565-G.exr	S2060-R40B.exr	S3055-R50B.exr	S3060-R80B.exr
S3540-R90B.exr	S3560-R90B.exr	S4550-B90G.exr	S4550-G50Y.exr
S4550-Y40R.exr	S7020-G70Y.exr	S7020-R50B.exr	

Tabulka 9: Seznam vzorků vzorníku NCS (15 ks)

## Mix vzorků

bambus-krabice.exr	bambus-krabice_2.exr	cotton-panama-702-01.exr
cotton-panama-702-04.exr	cotton-panama-702-06.exr	cedulka-leskla.exr
cedulka-matna.exr	drevotriska.exr	drevo-viko.exr
kozenka-cerna.exr	kuze-prirodni-lic.exr	kuze-prirodni-rub.exr
preklizka.exr	preklizka_2.exr	preklizka_svetla.exr
preklizka_svetla_2.exr	talir.exr	

Tabulka 10: Seznam mix vzorků (17 ks)

## B Náhledové obrázky vyříznuté z vytvořených videí



(a) Vzorek spachtle-leskla.bsdf



(b) Vzorek papir-oranzovy-leskly.bsdf



(c) Vzorek plast-cerny.bsdf



(d) Vzorek guma-fialova.bsdf



(e) Vzorek plst-tmave-modra.bsdf



(f) Vzorek h21-yc21.bsdf

Obrázek 45: Model draka<sup>[35]</sup> – použití naměřených dat od vybraných vzorků



(g) Vzorek ppg1230\_5.bsdf



(h) Vzorek s4550-g50y.bsdf



(i) Vzorek preklizka.bsdf

Obrázek 45: Model draka<sup>[35]</sup> – použití naměřených dat od vybraných vzorků (pokračování)

## C Popis struktury přiložených souborů

- └─ app – Zdrojové kódy vytvořené knihovny a vizualizační aplikace
  - └─ brdf\_gui – Zdrojové kódy aplikace
  - └─ brdf\_parser – Zdrojové kódy knihovny
  - └─ README.md – Popis a návod na sestavení
- └─ data – Naměřená a zpracovaná BRDF data
  - └─ guma – 12 ks gumových vzorků
  - └─ kov – 22 ks kovových vzorků
  - └─ mix – 17 ks mix vzorků
  - └─ papir – 40 ks papírových vzorků
  - └─ plast – 12 ks plastových vzorků
  - └─ plst – 9 ks plstěných vzorků
  - └─ vzornik\_metalic – 16 ks vzorků metalického vzorníku
  - └─ vzornik\_ncs – 15 ks vzorků vzorníku NCS
  - └─ vzornik\_ppg – 74 ks vzorků vzorníku PPG
- └─ images – Náhledové obrázky videí a testovací scény v herních enginech
- └─ unity – Projekt pro Unity obsahující vytvořený shader a testovací scénu
- └─ unreal – Projekt a soubory změn Unreal Engine
  - └─ engine – Soubory změn provedených v zdrojovém kódu Unreal Engine
    - └─ new\_files – Nově přidané soubory do zdrojového kódu
    - └─ modified.patch – Soubor obsahující změny existujících souborů zdrojového kódu
    - └─ README.md – Návod na aplikování změn
  - └─ project – Projekt pro Unreal Engine obsahující testovací scénu
- └─ videos – Videá ilustrující použití naměřených dat na zobrazeném 3D modelu