BRNO UNIVERSITY OF TECHNOLOGY FACULTY OF INFORMATION TECHNOLOGY

Excel@FIT 2016
http://excel.fit.vutbr.cz

Light Propagation Volumes

Tomáš Růžička*

Abstrakt
Light Propagation Volumes je metoda, která dovoluje v oblasti počítačové grafiky počítat globální osvětlení scény v reálném čase. Lze ji využít jako zdroj nepřímého osvětlení scény spolu s konvenčními metodami přímého nasvícení a stínovými mapami. Pro výpočet nepřímého osvětlení nepotřebuje žádná předpřipravená data a je vhodná pro dynamické scény, např. ve videohrách. Implicitně podporuje měkké stíny a lze ji rozšířit o podporu většího počtu nepřímých světelných odrazů, případně výsledek využít pro simulaci lesklých povrchů v reálném čase nebo pro zobrazení mlhy. Svým charakterem se neřadí mezi tzv. Screen Space metody, a proto produkuje uvěřitelnější výsledky. Tato metoda vychází z předpokladu nízkofrekvenčního charakteru odraženého světla v prostoru, a proto je výpočet založen na distribuci světla vyjádřené pomocí koeficientů sférických harmonických funkcí nízkého řádu v trojrozměrné mřížce. Výsledná implementace dokáže spočítat globální osvětlení scény při zobrazení testovací scény Sponza s využitím 4 kaskád LPV a 8 propagací za 5.5 ms.

Klíčová slova: Light Propagation Volumes — Reflective Shadow Maps — sférické harmonické — globální iluminace — nepřímé osvětlení — real-time zobrazování

Přiložené materiály: Demonstrační obrázky — Stažitelný kód

*xruzic42@stud.fit.vutbr.cz, *Fakulta informačních technologií, Vysoké učení technické v Brně*

1. Úvod

Výpočet globálního osvětlení je v počítačové grafice náročný proces a není jednoduché navrhnout efektivní metodu pro vyřešení takového problému v reálném čase. Proto se v minulosti používaly a stále používají, z důvodu omezeného výkonu grafických adaptérů, zjednodušené techniky stínování. V tomto článku je představena metoda, která dovoluje efektivně počítat první odraz nepřímého osvětlení pomocí grafické karty v reálném čase.

Hlavním problémem globálního osvětlení je jeho komplexnost. Na rozdíl od lokálního osvětlení, kdy je zobrazovací rovnice [1] zjednodušena pouze na přímé šíření světla scénou, u globálního osvětlení je potřeba uvažovat s úplnou rovnicí, která definuje šíření paprsku světla ve všech možných směrech při dopadu na povrch objektu. Nicméně analytické řešení této rovnice není v reálném čase možné, a proto si real-time osvětlovací metody tuto rovnici zjednodušují na úkor výsledné kvality.

Existuje mnoho metod pro výpočet globálního osvětlení v reálném čase. Do skupiny metod, které pracují pouze v oblasti obrazovky lze zařadit např. *Screen Space Ambient Occlusion* [2] [3], *Horizon Based Ambient Occlusion* [4] a *Screen Space Direct Occlusion* [5]. Cílem metod je ztmavit nebo zesvětlit pouze sobě blízké povrchy, nicméně tyto metody produkují pouze omezenou iluzi globálního osvětlení. Další skupinou jsou metody, které využívají principu *virtuálních bodových zdrojů světla* (VPL). Tento kon-

cept zavádí metoda *Instant Radiosity* [6] [7] a využívá jej mj. technika *Reflective Shadow Maps* [8]. RSM vykresluje scénu z pozice primárních zdrojů světla pro získání VPL, které využívá pro určení světelných odrazů. Metoda je výpočetně náročnější při zobrazování geometricky složitých scén. Dále existuje paralelní implementace *Photon Mappingu* [9], která ale dosahuje kvalitních výsledků pouze při větším počtu fotonů, což se velmi negativně promítá do doby výpočtu. *Voxel Cone Tracing* [10] je technika pro výpočet globálního osvětlení, která je v dnešní době v herních enginech poměrně oblíbená. Základním principem je voxelizace scény, do struktury oktálového stromu. Do listových ulzů stromu jsou vloženy VPL a následně propagovány do vyšších úrovní. Při vykreslování scény jsou z pozic vzorků vrhány kužely do okolí s cílem posbírat světelnou intenzitu z různých úrovní stromu.

Tento článek je zaměřen na metodu *Light Propagation Volumes*, která využívá ke svému výpočtu dvě trojrozměrné mřížky. První z nich je určena k uložení a propagaci světelné intenzity (LPV) a druhá obsahuje zjednodušenou část geometrie pro zastínění (*Geometry Volume* – GV). Tyto hodnoty nejsou v mřížkách uloženy přímo, ale jsou přepočteny na koeficienty *sférických harmonických funkcí* (SH). Tím lze získat hrubou aproximaci směru šíření světla pro každou buňku mřížky LPV, resp. směr zastínění v případě GV. Celý výpočet se skládá ze tří části: injekce, propagace a vykreslení. V první části se využívá technika *Reflective Shadow Maps* pro vykreslení scény z pozice světel. Výsledkem jsou opět *virtuální bodové zdroje světla* (VPL), které jsou převedeny na SH koeficienty a vloženy do mřížek LPV a GV. Následuje iterativní proces propagace, kdy je intenzita každé buňky distribuována do sousedních buněk s využitím GV pro útlum v daném směru. Posledním krokem je převedení SH koeficientů z LPV na světelnou intenzitu pro každý vykreslovaný vzorek scény.

Následující sekce 2 se věnuje teoretické části metody *Light Propagation Volumes*. Sekce 3 se zabývá rozšířeními této techniky. Sekce 4 popisuje implementaci a v sekci 5 jsou zaznamenány naměřené hodnoty. Článek uzavírá sekce 6, kde jsou shrnuty dosažené výsledky.

2. Light Propagation Volumes

Tuto metodu výpočtu globální iluminace představilo studio *Crytek* v roce 2009 a integrovalo ji do svého enginu *CryEngine*. Nicméně v současné době byla tato technika nahrazena metodami *SSDO* a *Voxel Cone Tracing*.

Jak již bylo zmíněno v úvodu, metoda LPV využívá dva druhy mřížek: první pro světelnou intenzitu (LPV) a druhou pro blokaci propagace intenzity (GV). Buňky těchto mřížek obsahují směrovou distribuci světelné intenzity resp. útlumu (*blokující potenciál*) v SH koeficientech nízkého řádu. V případě LPV se pro uložení SH využívají tři na sobě nezávislé mřížky pro každý barevný kanál zvlášť. Pro lepší interpolaci blokujícího potenciálu jsou středy buněk z mřížky geometrie (GV) zarovnány s rohy buněk LPV. Při zobrazování rozsáhlých scén je vhodné použít kaskádu různě velkých mřížek pro zachycení drobných detailů osvětlení blízko kamery a hrubých detailů vzdálených objektů. Blíže je tato část popsána v sekci 3.

Samotný výpočet metody lze shrnout do následujících tří bodů:

- **inicializace** mřížek světla (LPV) a zastínění (GV)
- **iterativní propagace** světla v LPV s využitím GV
- **vykreslení** scény s využitím LPV při stínování

2.1 Inicializace mřížek

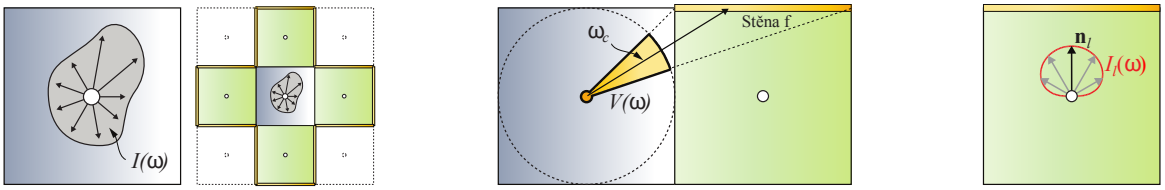
Prvním krokem při výpočtu je získání informací o sekundárním osvětlení scény a základní prostorové struktuře scény. Tyto informace jsou poté vloženy do mřížek LPV a GV. K tomuto účelu se využívá technika *Reflective Shadow Maps* (RSM), jejímž principem je vykreslení scény z pozice primárních zdrojů světla a z pozice kamery. Výstupem RSM je trojice textur obsahující ambientní osvětlení, pozici a normálu geometrie. Každý texel této trojice představuje virtuální bodový zdroj světla (VPL), který se nachází v místě scény dopadu paprsku přímého osvětlení. Všechna VPL jsou následně vložena na konkrétní místo v LPV, přičemž vkládání intenzity do LPV se provádí pouze pro RSM z primárních světel.

V případě GV jsou tato VPL použita jako směrový blokující potenciál podle normály původní geometrie. Kromě RSM z primárních světel je pro injekci GV použita také RSM z pozice kamery, z důvodu získání více informací o geometrii scény.

Pro plně dynamické scény, kde se pohybují světla i objekty, je potřeba, aby inicializace a propagace probíhala v každém snímku. Pro eliminaci nežádoucího osvětlení geometrie sebe samé je vhodné pozici všech VPL při vložení posunout o polovinu velikosti buňky LPV ve směru normály VPL.

Před injekcí je potřeba pro každé VPL nejprve vyjádřit směrovou distribuci světla $I_p(\vec{\omega})$ z pozice VPL, s normálou $\vec{n_p}$ a odraženou září Φ_p [11]:

$$I_p(\vec{\omega}) = \frac{\Phi_p}{\pi} \langle \vec{n_p} | \vec{\omega} \rangle_+ \tag{1}$$



zdrojová buňka | propagace do okolních buněk | zdrojová buňka | cílová buňka | reprojekce světla do buňky LPV

Obrázek 1. Postup výpočtu při propagaci SH koeficientů světelné intenzity v LPV. Vlevo: zdrojová buňka (modrá) obsahující SH koeficienty a cílové buňky (zelené). Uprostřed: reprojekce SH na stěnu sousední buňky. Vpravo: sevřený kosínový lalok při reprojekci SH.

Následně je potřeba převést distribuci na koeficienty *sférických harmonických funkcí* pro každou barevnou složku zvlášť. SH koeficienty zde představují intenzitu toku světla závislé na směru a pro jejich vyjádření jsou zvoleny polynomy SH stupně $l = 2$ (SH nízkého řádu), což odpovídá čtyřem koeficientům $y_l^m(\vec{\omega})$ pro popis těchto bázových funkcí. Bázové funkce v polynomiální formě lze zapsat tvarem [12]:

$$\begin{aligned} y_0^0(\vec{\omega}) &= \frac{1}{2\sqrt{\pi}} & y_1^{-1}(\vec{\omega}) &= -\frac{\sqrt{3}}{2\sqrt{\pi}}y \\ y_1^0(\vec{\omega}) &= \frac{\sqrt{3}}{2\sqrt{\pi}}z & y_1^1(\vec{\omega}) &= -\frac{\sqrt{3}}{2\sqrt{\pi}}x \end{aligned} \tag{2}$$

Pro získání těchto koeficientů SH lze využít *sevřený kosínový lalok* otočený ve směru normály VPL vyjádřený pomocí *zonálních harmonických funkcí* (ZH) [13]:

$$\begin{aligned} c_0 &= \frac{\sqrt{\pi}}{2} & c_1 &= -\sqrt{\frac{\pi}{3}}y \\ c_2 &= \sqrt{\frac{\pi}{3}}z & c_3 &= -\sqrt{\frac{\pi}{3}}x \end{aligned} \tag{3}$$

Výsledné koeficienty je nutné vynásobit jedním ze tří barevných kanálů vyzařovaného světla VPL a výsledek poté vložit do buňky LPV daného kanálu.

Inicializace mřížky GV blokujícím potenciálem v SH probíhá obdobně jako v případě LPV výše, nicméně při situaci vložení několika různých SH na stejné místo do GV je potřeba vybrat SH s největší hodnotou útlumu. Blokující potenciál $B(\vec{\omega})$, kterým jsou před vložením vynásobeny koeficienty SH, závisí na velikosti s mřížky GV, velikosti stěny buňky A_s a úhlem mezi jeho normálou $\vec{n}_s$ a směrem světla $\vec{\omega}$ [14]:

$$B(\vec{\omega}) = A_s s^{-2} \langle \vec{n}_s | \vec{\omega} \rangle_+ \tag{4}$$

2.2 Propagace intenzity

Propagace je iterativní část výpočtu v mřížce LPV, jejímž hlavním smyslem je distribuce světelné intenzity každé buňky do svého okolí (obr. 1). Pro správný proces propagace je nutné mezivýsledek každé iterace ukládat do prázdné LPV mřížky a výsledná intenzita je dána součtem všech těchto mřížek.

Hlavním principem výpočtu propagace je přenesení (reprojekce) koeficientů SH zdrojové buňky na nesdílené stěny okolních šesti buněk LPV. Výsledná intenzita cílové buňky je tak dána součtem výsledků třiceti reprojekcí z okolních buněk. Pro jednu stěnu buňky f lze celkovou intenzitu Φ_f analyticky určit pomocí [14]:

$$\Phi_f = \int_\Omega I(\vec{\omega}_c) V(\vec{\omega}) \mathrm{d}\vec{\omega} \tag{5}$$

Kde $I(\vec{\omega})$ je přibližná intenzita světla zdrojové buňky:

$$I(\vec{\omega}) \approx \sum_i c_l^m y_l^m(\vec{\omega}) \tag{6}$$

$V(\vec{\omega})$ reprezentuje *operátor viditelnosti* stěny f, pro paprsek vržený ze středu buňky:

$$V(\vec{\omega}) \approx \sum_i v_l^m y_l^m(\vec{\omega}) \tag{7}$$

Nicméně výpočet viditelnosti lze urychlit pomocí skalárního součinu koeficientů SH c_l^m s koeficienty viditelnosti v_l^m. Ovšem z důvodu nízké přesnosti těchto koeficientů je místo toho určena hodnota *pevného úhlu* $\Delta_{\vec{\omega}f}$ pro každou stěnu f cílové buňky:

$$\Delta_{\vec{\omega}f} = \int_\Omega V(\vec{\omega}) \mathrm{d}\vec{\omega} \tag{8}$$

Světelnou intenzitu $\Phi_{\vec{\omega}f}$ dopadající ve směru $\vec{\omega}$ na stěnu f lze nakonec vyjářit jako průměrnou intenzitu v oblasti pevného úhlu ve směru $\vec{\omega}_c$ dopadající na střed stěny:

$$\Phi_{\vec{\omega}f} = \frac{\Delta_{\vec{\omega}f}}{4\pi} I(\vec{\omega}_c) \tag{9}$$

Dle poznámek autora [11] lze rozlišit dva *pevné úhly*: pro stěnu otočenou přímo (0.4_{rad}) a pro stěnu natočenou bokem (0.4234_{rad}) ke zdrojové buňce.

Během propagace je pro každou stěnu f vytvořeno nové VPL na pozici středu cílové buňky, je otočeno na

danou stěnu a jeho intenzita Φ_l odpovídá intenzitě Φ_f [11]. Určit SH koeficienty tohoto VPL lze obdobně jako v případě inicializace LPV, a to vynásobením *sevřeného kosínového laloku* otočeného ve směru VPL s pevným úhlem $\Delta_{\omega f}$ a intenzitou Φ_l. Tu lze získat pomocí skalárního součinu SH koeficientů normály ve směru na střed cílové stěny (rovnice 2) s SH koeficienty zdrojové buňky. Výsledná intenzita cílové buňky je dána součtem všech SH koeficientů dílčích reprojekcí.

Během propagace SH koeficientů v LPV je také využívána mřížka GV pro správné určení útlumu intenzity. SH koeficienty blokujícího potenciálu jsou určeny pomocí interpolace nejbližších osmi koeficientů na pozici středu protínané stěny při propagaci. Získaným blokujícím potenciálem stačí vynásobit dané SH koeficienty intenzity.

Na konci každé iterace propagace SH koeficientů v LPV je tato mřížka připravena na vykreslení nebo na využití při další iteraci propagace.

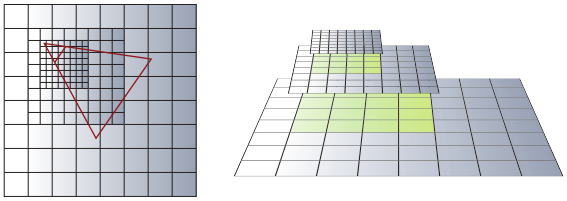
2.3 Využití při vykreslení

Po dokončení propagací je možné mřížku LPV využít při stínování scény způsobem, kdy je nejprve pro každý vykreslovaný vzorek geometrie zjištěna jeho pozice ve scéně a normála. Pomocí pozice lze získat trilineárně interpolovanou sadu SH koeficientů z LPV pro tři barevné kanály. Následně jsou SH koeficienty vyhodnoceny ve směru obrácené normály vzorku (rovnice 2) a pomocí skalárního součinu jsou vynásobeny s hodnotami z mřížky LPV. Provedením této operace nad všemi barevnými kanály lze získat výsledné globální osvětlení scény.

Při vykreslování však mohou nastat případy tzv. *světelného průsaku*, kdy se objeví světlé plochy na místech, kde by správně měly být plochy tmavé. Tento problém může být způsoben velkým rozměrem buněk LPV a tenkou zástěnou mezi buňkami. Autoři [14] přišli s řešením pomocí tzv. *faktoru útlumu*, který využívá směrovou změnu distribuce světelné intenzity. Pro každý vzorek s pozicí x a normálou $\vec{n}$ je potřeba určit interpolované SH koeficienty c_l^m a směrovou odchylku ve směru normály vzorku $\nabla_{\vec{n}} c_l^m$. Pokud je odchylka větší, potom jsou SH koeficienty c_l^m utlumeny.

3. Rozšíření techniky LPV

Metodu LPV lze rozšířit v mnoha ohledech. K základnímu rozšíření, které je nezbytné při zobrazování rozsáhlých scén je použití kaskády několika různě velkých mřížek pro LPV a GV – *Cascaded Light Propagation Volumes* (CLPV). Hlavním přínosem je možnost zachytit globální osvětlení malých ploch blízko kamery



Obrázek 2. Ukázka kaskády LPV. V tomto případě jsou úrovně zarovnány dle pohledového frusta.

a snížení rozlišení metody u ploch vzdálených ve scéně. Jednotlivé úrovně mřížek jsou zarovnány buď podle pohledového frusta nebo svým středem na pozici kamery s přesností na násobky velikosti jedné buňky (obr. 2). Postup výpočtu CLPV je podobný původní metodě: injekci SH koeficientů do mřížek je potřeba provádět pro všechny kasdády. Dále propagace SH koeficientů probíhá nezávisle pro každou mřížku a při vykreslení scény je dle vzdálenosti vzorku interpolováno mezi dvěma úrovněmi mřížek.

Jinými rozšířeními může být využití LPV pro zobrazení odlesků na povrchu objektu pomocí *ray marchingu* v LPV mřížce nebo rozšíření výpočtu o více světelných odrazů během propagace SH koeficientů.

Rozšíření, která navrhl autor článku jsou: *zachování původního směru šíření světelné intenzity*, *vložení a rozptyl světla z pozadí scény* a *simulace průchodu světla průsvitnými objekty*. První z navržených rozšíření řeší problém zpětného nasvícení ploch při větším počtu propagací (2 a více), jejichž části generují VPL. Tento problém je způsoben využitím SH funkcí nízkého řádu s pouze čtyřmi koeficienty. Problém lze vyřešit použitím SH vyššího řádu s větším počtem koeficientů, nicméně takové řešení by výrazně zvýšilo počet čtení dat z grafické paměti. Proto byl navržen postup ukládat a distribuovat do okolí LPV buněk, během propagace, také původní sevřený kosínový lalok, který je použitý jako maska pro získané SH koeficienty během propagace.

Rozšíření *vložení a rozptylu světla z pozadí scény* bylo navrženo z důvodu zdokonalení grafického výstupu metody. Technika se liší od původní metody LPV ve dvou bodech:

- injekce VPL do LPV je provedena pouze do nejvyšší úrovně buněk ve svislém směru při pohledu na scénu kolmo dolů
- propagace využívá multiplikativní konstantu pro zamezení útlumu světla při rozptylu

Vzhledem k odlišnému výpočtu během propagace je potřeba provádět výpočet v separátní mřížce LPV.

Poslední rozšíření zabývající se *průchodem světla skrz průsvitné materiály* je kombinací původní metody LPV a předchozího rozšíření:

- injekce je prováděna standardním způsobem
- propagace využívá sevřené kosínové laloky pro usměrnění šíření světla a násobitel pro sousední SH koeficienty závisející na hustotě materiálu objektu.

4. Implementace

Demonstrační aplikace je implementována v jazyce C++ s využitím knihoven Qt, SDL a grafické knihovny OpenGL. Pro výpočet globálního osvětlení je implementována metoda LPV rozšířená o 4 kaskády (stejný počet pro RSM). Lokální osvětlení scény zajišťuje *Blinn – Phong* osvětlovací model a pro zobrazení stínů je zvolena technika *Cascaded Shadow Mapping* (CSM) s 6 kaskádami.

Pro násobné vykreslování scény u technik RSM a CSM je použita funkce instancování geometrie. Uložení výstupu bylo nejprve implementováno s využitím *geometry shaderu* a 2d textury pole pro úrovně kaskád, nicméně z důvodu výrazného propadu výkonu jsou nyní tyto úrovně ukládány jako dlaždice do jedné 2d textury s využitím ořezových rovin.

Pro metodu LPV jsou zvoleny SH řádu 2, čímž lze získat 4 SH koeficienty. Tento počet je vhodný pro uložení na grafické kartě, protože pomocí jedné operace čtení z textury lze získat všechny SH koeficienty. Proces inicializace a propagace LPV je implementován, pro zvýšení kompatibility, ve dvou variantách: GS – s využitím grafického zobrazovacího řetězce (OpenGL 3.2+) a CS – pomocí *compute shaderů* (OpenGL 4.3+). Každá z těchto variant je naprogramována metodami sběru a rozptylu. Metoda sběru používá každou prováděnou buňku LPV jako cílovou a okolní buňky jako zdrojové, ze kterých počítá reprojekce SH koeficientů na stěny cílové buňky. Zatímco metoda rozptylu počítá propagaci standardním způsobem, jaký byl představen v části 2.2.

V případě GS varianty jsou použity 2 *framebuffery* (FBO) každý se 7 3D texturami ve formátech `GL_RGBA32F`. Tyto textury slouží pro uložení výsledků poslední propagace v LPV, uložení všech propagací a uložení blokujícího potenciálu. Pro každou texturu jsou jednotlivé kaskády LPV nebo GV uloženy vedle sebe v ose X. Proces inicializace LPV je proveden pomocí *point renderingu* s aktivovaným additivním mícháním, kdy každý vykreslovaný bod představuje jedno VPL z dané úrovně kaskády RSM. Tento bod je v *geometry shaderu* transformován na konkrétní pozici VPL a ve *fragment shaderu* jsou určeny jeho SH koeficienty. Postup propagace je prováděn v cyklu s využitím postupného přepínání obou FBO. V každé iteraci je aktivován jeden FBO, který je vymazán a následně je vykreslena geometrie tak, aby invokovala grafický řetězec pro všechny buňky LPV. V případě metody sběru jsou vykresleny *quady* pro všechny vrstvy textur a pro metodu rozptypu je opět proveden *point rendering*.

CS varianta s *compute shadery* využívá, z důvodu omezeného počtu *image jednotek*, pouze čtyři textury: dvě pro LPV propagaci, třetí pro akumulaci propagace a poslední pro GV. Vzhledem k využití atomických operací pro injekci a propagaci, které jsou definovány pouze nad celočíselným typem textur, jsou SH koeficienty v textuře prokládány. Na rozdíl od GS varianty propagace LPV, kde jsou jednotlivé iterace řízeny na straně CPU, v případě této varianty je veškerá propagace včetně čištění a přepínání v režii *compute shaderu*.

Z důvodu časového omezení nebyly v době psaní článku prozatím implementovány autorem navržené rozšíření techniky LPV.

5. Výsledky

Měření výsledků bylo prováděno na následující testovací sestavě: CPU: AMD Athlon 64 X2 3800+, RAM: 2 GB DDR2, GPU: GeForce GTX 960 2 GB. Pro testovací scénu byl zvolen model Sponza, ve scéně bylo umístěno jedno směrové světlo a pro zobrazení CSM stínů bylo zvoleno 6 kaskád o rozlišení 1024^2.

Obrázek 3 zachycuje lokální a globální osvětlení s využitím různého počtu propagací. Z obrázku lze pozorovat kvadratickou závislost počtu propagací na výsledném globálním osvětlení z důvodu vysokého rozptylu světelné intenzity. Obrázek 4 ukazuje jednotlivé varianty implementace propagace LPV. U variant, které využívají *geometry* a *fragment shader* pro výpočet, lze pozorovat artefakty projevující se zpětným osvětlením ploch v oblasti původních VPL (např. zem). Nejlepší výsledky propagace se objevují u CS varianty počítané metodou rozptylu.

Následující tabulky shrnují měření času pro různé situace. V tabulce 1 je zachycen vliv počtu VPL na čas vykreslení scény do RSM a inicializace LPV pomocí implementace s použitím *geometry shaderu* a *compute shaderu*. Z měření je patrné, že zvyšování počtu VPL nemá velký vliv na vytvoření RSM, nicméně u injekce lze pozorovat kvadratickou závislost a CS varianta je výrazně pomalejší.

Tabulky 2 a 3 obsahují chování implementace algoritmu propagace v časové oblasti, v prvním případě na základě počtu propagací a ve druhém případě na velikosti LPV mřížky. Měření bylo prováděno pro všechny čtyři implementace a dle výsledků je první zcela nejrychlejší. Výrazné zpomalení GS varianty

Obrázek 3. Vlevo nahoře: lokální osvětlení se stíny. Ostatní: globální osvětlení počítané pomocí varianty GS metodou sběru s využitím 2, 8 a 32 propagací. Počet VPL: 64^2, velikost LPV: 32^3, počet LPV kaskád: 4.

Velikost RSM	Kreslení	Injekce GS	Injekce CS
32^2	2.7	0.21	0.4
64^2	2.75	0.27	1.04
128^2	2.85	0.34	3.61
256^2	3.17	0.62	14.1
512^2	4.23	1.62	56.227

Tabulka 1. Meření času (v ms) vytvoření RSM a její použití při injekci LPV pro varianty s geometry shaderem a compute shaderem v závislosti na její velikost (počtu VPL). Počet RSM a LPV kaskád: 4, velikost LPV: 32^3.

rozptylu lze přičítat vysokému počtu generovaných bodů v *geometry shaderu* a zpomalení CS variant je pravděpodobně způsobeno nevhodným rozvržením dat v texturách a vysokým počtem atomických operací.

Pro srovnání, implementace autorů této metody [14] provedla výpočet techniky LPV v čase 3.7 ms při využití 3 kaskád o rozměru 32^3 a 8 propagací pomocí grafické karty GeForce GTX 285. Způsob propagace SH koeficientů odpovídal variantě sběru vykonanou grafickým řetězcem.

Propagací	GS sběr	GS rozp.	CS sběr	CS rozp.
1	0.77	5.05	12.17	13.84
2	1.52	10.03	14.05	17.26
4	2.88	19.96	18.14	22.99
8	5.46	40.20	25.86	38.68
16	11.58	88.52	42.54	61.82
32	24.66	229.60	72.45	114.71

Tabulka 2. Meření času (v ms) propagace SH koeficientů v LPV pro varianty s geometry shaderem a compute shaderem a pro každou metodou sběru a rozptylu. Počet LPV kaskád: 4, velikost LPV: 32^3.

Obrázek 4. Porovnání jednotlivých implementací LPV propagace. Nahoře: GS sběr a rozptyl, uprostřed: CS sběr, dole: CS rozptyl. Počet VPL: 64^2, velikost LPV: 32^3, počet propagací: 8, počet LPV kaskád: 4.

Velikost LPV	GS sběr	GS rozp.	CS sběr	CS rozp.
8^3	1.03	1.39	0.95	1.18
16^3	1.88	5.77	3.36	4.57
32^3	5.65	40.35	25.99	38.95
64^3	34.48	312.28	116.79	119.34

Tabulka 3. Meření času (v ms) propagace SH koeficientů v LPV pro varianty s geometry shaderem a compute shaderem a pro každou variantu metodou sběru a rozptylu v závislosti na velikosti LPV mřížky. Počet LPV kaskád: 4, počet propagací: 8.

6. Závěr

V tomto článku byla představena a popsána metoda výpočtu globálního osvětlení v reálném čase Light Propagation Volumes. Tato metoda využívá pro výpočet dvě trojrozměrné mřížky pro uložení a propagaci světelné intenzity a blokujícího potenciálu v koeficientech sférických harmonických funkcí.

Metoda byla implementována ve čtyřech různých variantách: s využitím buď *geometry shaderů* nebo *compute shaderů* a různým přístupem k výpočtu propagace SH v LPV: metodou sběru a rozptylu. Dle výsledků měření je nejrychlejší varianta propagace pomocí *geometry shaderů* metodou sběru SH koeficientů, která dokáže aproximovat globální osvětlení scény v čase 5.5 ms pro 4 LPV kaskády s velikostí 32^3.

Budoucí vývoj bude zaměřen na optimalizaci výpočtu pomocí *compute shaderů* a implementaci navržených rozšíření.

7. Poděkování

Rád bych poděkoval Ing. Tomáši Miletovi za jeho podporu a odbornou pomoc, kterou mi poskytl při psaní tohoto článku a vedení diplomové práce na stejné téma.

Literatura

[1] James Kajiya. The rendering equation. In *Proceedings of the 13th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '86, pages 143–150, New York, NY, USA, 1986. ACM.

[2] Sergej Zhukov, Andrej Inoes, and Grigorij Kronin. An ambient light illumination model. In *Rendering Techniques '98*, Eurographics, pages 45–55. Springer Vienna, 1998.

[3] Martin Mittring. Finding next gen: Cryengine 2. In *ACM SIGGRAPH 2007 Courses*, SIGGRAPH '07, pages 97–121, New York, NY, USA, 2007. ACM.

[4] Rouslan Dimitrov Louis Bavoil, Miguel Sainz. Image-space horizon-based ambient occlusion. In *ACM SIGGRAPH 2008 Talks*, SIGGRAPH '08, pages 22:1–22:1, New York, NY, USA, 2008. ACM.

[5] Tobias Ritschel, Thorsten Grosch, and Hans-Peter Seidel. Approximating dynamic global illumination in image space. In *Proceedings of the 2009 Symposium on Interactive 3D Graphics and Games*, I3D '09, pages 75–82, New York, NY, USA, 2009. ACM.

[6] Alexander Keller. Instant radiosity. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '97, pages 49–56, New York, NY, USA, 1997. ACM Press, Addison-Wesley Publishing Co.

[7] Samuli Laine, Hannu Saransaari, Janne Kontkanen, Jaakko Lehtinen, and Timo Aila. Incremental instant radiosity for real-time indirect illumination. In *Proceedings of the 18th Eurographics Conference on Rendering Techniques*, EGSR'07, pages 277–286, Aire-la-Ville, Switzerland, 2007. Eurographics Association.

[8] Carsten Dachsbacher and Marc Stamminger. Reflective shadow maps. In *Proceedings of the 2005 Symposium on Interactive 3D Graphics and Games*, I3D '05, pages 203–231, New York, NY, USA, 2005. ACM.

[9] Michael Mara, David Luebke, and Morgan McGuire. Toward practical real-time photon mapping: Efficient gpu density estimation. In *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, I3D '13, pages 71–78, New York, NY, USA, 2013. ACM.

[10] Cyril Crassin, Fabrice Neyret, Miguel Sainz, Simon Green, and Elmar Eisemann. Interactive indirect illumination using voxel cone tracing. *Computer Graphics Forum*, 2011.

[11] Andreas Kirsch. Light propagation volumes – annotations. [online], 2010.

[12] Peter Sloan. Stupid spherical harmonics (sh) tricks. In *Archives of the Game Developers Conference*, GDC '08, 2008.

[13] Johan Elvek. Extending ogre with light propagation volumes. Diplomová práce, University of Gothenburg, Göteborg, Sweden, 2012.

[14] Anton Kaplanyan and Carsten Dachsbacher. Cascaded light propagation volumes for real-time indirect illumination. In *Proceedings of the 2010 ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, I3D '10, pages 99–107, New York, NY, USA, 2010. ACM.