

# Raytracing

I. Bobuskyy\*, D. Humpál\*\*, H. Jemelík\*\*\*, T. Pokorný\*\*\*

\*Gymnázium Benešov, \*\* Gymnázium Christiana Dopplera, \*\*\*Gymnázium Brno, tř. Kpt. Jaroše

xtompok@gmail.com

## Abstrakt

Raytracing je globální zobrazovací metoda, jejíž cílem je fotorealistické zobrazování. V rámci našeho miniprojektu jsme implementovali základní podobu algoritmu raytracingu a při renderování několika scén jsme ověřili její funkčnost.

## 1 Úvod

Pomocí raytracingu (tj. sledování paprsku) lze do značné míry fotorealisticky renderovat (zobrazovat) 3D scénu popsanou pomocí geometrických objektů. Podstata raytracingu vychází z geometrické optiky a jeho výhodou je jednodušnost algoritmu i snadná implementace. V určitých modifikacích se používá jako základní součást komplexních algoritmů pro fotorealistické zobrazování, jejichž výsledkem je mimo jiné zachycení co možná nejširší škály efektů.

## 2 Princip

Princip spočívá v tom, že každým pixelem průmětny vedeme paprsek směrem od pozorovatele (resp. středu promítání). Barva pixelu vznikne složením barev mnoha paprsků.

### Šíření světla (v základní verzi algoritmu)

Najdeme nejbližší povrch, který paprsek protíná (od kterého přichází). Světlo přicházející z toho bodu obecně tvoří:

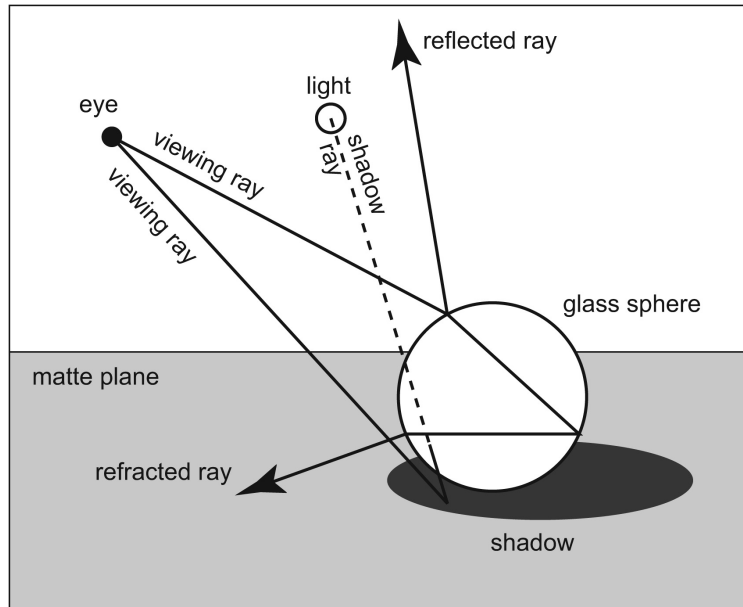
1. ambientní osvětlení objektu
2. difuzní odraz, tj. světlo vzniklé přímým difuzním odrazem (bodových) světelných zdrojů od povrchu
3. zrcadlový odraz světelného zdroje
4. světlo přicházející od jiného objektu a odražené v odpovídajícím směru
5. světlo procházející průhledným objektem

První tři složky popisuje tzv. Phongův osvětlovací model [2].

### Sledování paprsku

V algoritmu se sledují následující typy paprsků (Obr. 1):

1. primární paprsek (viewing ray) - paprsek směrem od pozorovatele
2. sekundární paprsky vycházejí z bodu dopadu primárního paprsku. Jedná se o
  - (a) stínový paprsek (shadow ray) - jeho úkolem je zjistit, zda se místo dopadu nachází ve stínu. Z místa dopadu je vyslán stínový paprsek ke každému světelnému zdroji. Pokud paprsek neprotíná žádný objekt, je tento bod daným světelným zdrojem osvětlen. V opačném případě je vůči tomuto zdroji ve stínu.



Obrázek 1: Schéma algoritmu sledování paprsku

- (b) odražený paprsek (reflected ray) - paprsek, který přichází ze směru, ze kterého by se mohl odrazit směrem k pozorovateli
- (c) lomený paprsek (refracted ray) - prochází rozhraním prostředí s různými indexy lomu a světlo se zde láme.

V bodě průsečíku primárního paprsku s povrchem objektu vypočítáme Phongův osvětlovací model pro nezastíněné světelné zdroje. K výsledné barvě přičteme s odpovídajícím koeficientem barvu odražených a lomených paprsků, které jsme sledovali rekurzivně stejnou procedurou jako paprsek primární. K zastavení rekurze stačí splnit alespoň jednu z následujících podmínek.

- Paprsek opustí scénu.
- Příspěvek barvy je nedostatečně velký.
- Je dosažena maximální hloubka rekurze.

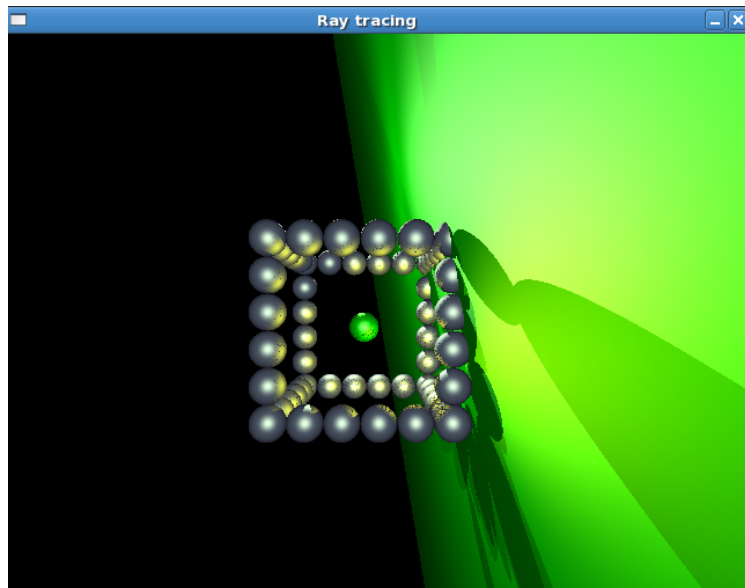
### 3 Implementace

Tvorba programu byla pojata jako týmový softwarový projekt a proto je celý algoritmus rozdělen do několika modulů, programovaných samostatně. Algoritmus jsme implementovali v jazyce C++ v prostředí linuxové distribuce CentOS. Ke grafickému vykreslení jsme využili knihovnu SDL. Náš program je napsán plně objektově, využívá dědičnosti a virtuálních metod. Implementovali jsme odrazy a průchody paprsků koulemi a rovinami, během další práce na projektu by bylo snadné jej doplnit o další třídy definující jiné objekty.

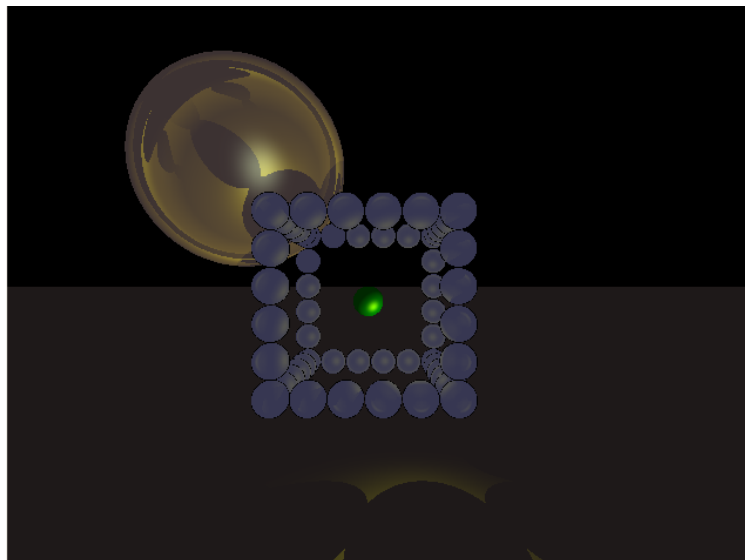
Při spuštění programu dojde k zavolání funkce `scena`, která inicializuje prostředí - objekty, kamery, zdroje světla. Následuje volání funkce `render`, jejímž úkolem je vyslat paprsek z pozice kamery postupně na všechny pixely promínaté plochy. K tomu se využívá funkce `SledujPaprsek`, která se rekurzivně volá a následuje dráhu paprsku. K tomuto účelu volá funkce `NejbliziPrusecik` a `PhongIllumination`, které zjistí nejbližší objekt na dané přímce a osvětlení daného bodu.

### 4 Výsledky

Pomocí našeho programu se nám podařilo vyrenderovat několik scén, z nichž dvě jsou uvedené v tomto příspěvku na Obr. 2 a Obr. 3.



Obrázek 2: Testovací scéna 1



Obrázek 3: Testovací scéna 2

## 5 Závěr

V rámci miniprojektu jsme se během dvou dnů seznámili s principem metody sledování paprsku pro počítačové zobrazování trojrozměrných scén a úspěšně jsme naprogramovali jejich implementaci. Přitom jsme si vyzkoušeli práci v týmu, moderní programovací techniky a další úkony, které bývají součástí dnešních softwarových projektů.

*Poděkování:* Rádi bychom poděkovali všem, kteří se podíleli na přípravě Týdne vědy na FJFI ČVUT a rovněž vedoucímu našeho miniprojektu Pavlu Strachotovi.

## Reference

- [1] P. Strachota: *Počítačová grafika* (přednášky). FJFI ČVUT, Praha, 2010
- [2] Žára, Beneš, Sochor, Felkel: *Moderní počítačová grafika*. Computer Press, 2005. ISBN: 80-251-0454-0
- [3] T. Whitted: *An improved illumination model for shaded display*. Comm. ACM, 23(6), (1980), 343-349