

---

# Fotonové mapy

---

Jaroslav Křivánek, KSVI, MFF, UK

[Jaroslav.Krivanek@mff.cuni.cz](mailto:Jaroslav.Krivanek@mff.cuni.cz)

# Globální osvětlení



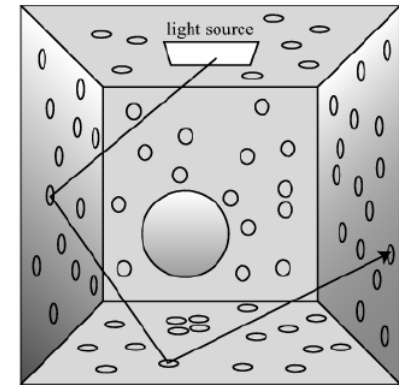
# Fotonové mapy

- **Cesty se sledují od zdrojů světla i od kamery**
  - Světla jsou zdroji radiance
  - Kamera reprezentuje důležitost (potenciál)
- Podobné obousměrnému sledování cest
  - Generování cest ze světla a z kamery jsou však dva oddělené procesy
- Přepoužití stejných světelných cest pro všechny pixely
  - Fotonová mapa = „cache cest od světla“
- Při stejné kvalitě je obvykle mnohem rychlejší než M-C techniky
- **Ztráta nestrannosti !**
  - ale konzistentní (konverguje při zvětšování počtu fotonů)

# Fáze výpočtu

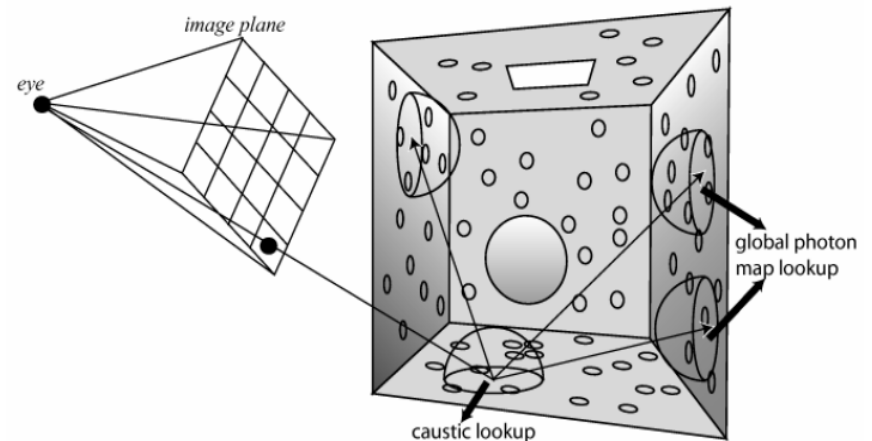
## 1. Rozmístění fotonů

- ❑ „Fotony“ jsou emitovány světelnými zdroji
- ❑ Propagují se do scény (a la light tracing)
- ❑ Ukládají se do „fotonových map“

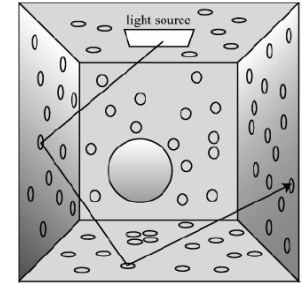


## 2. Rendering s využitím fotonových map

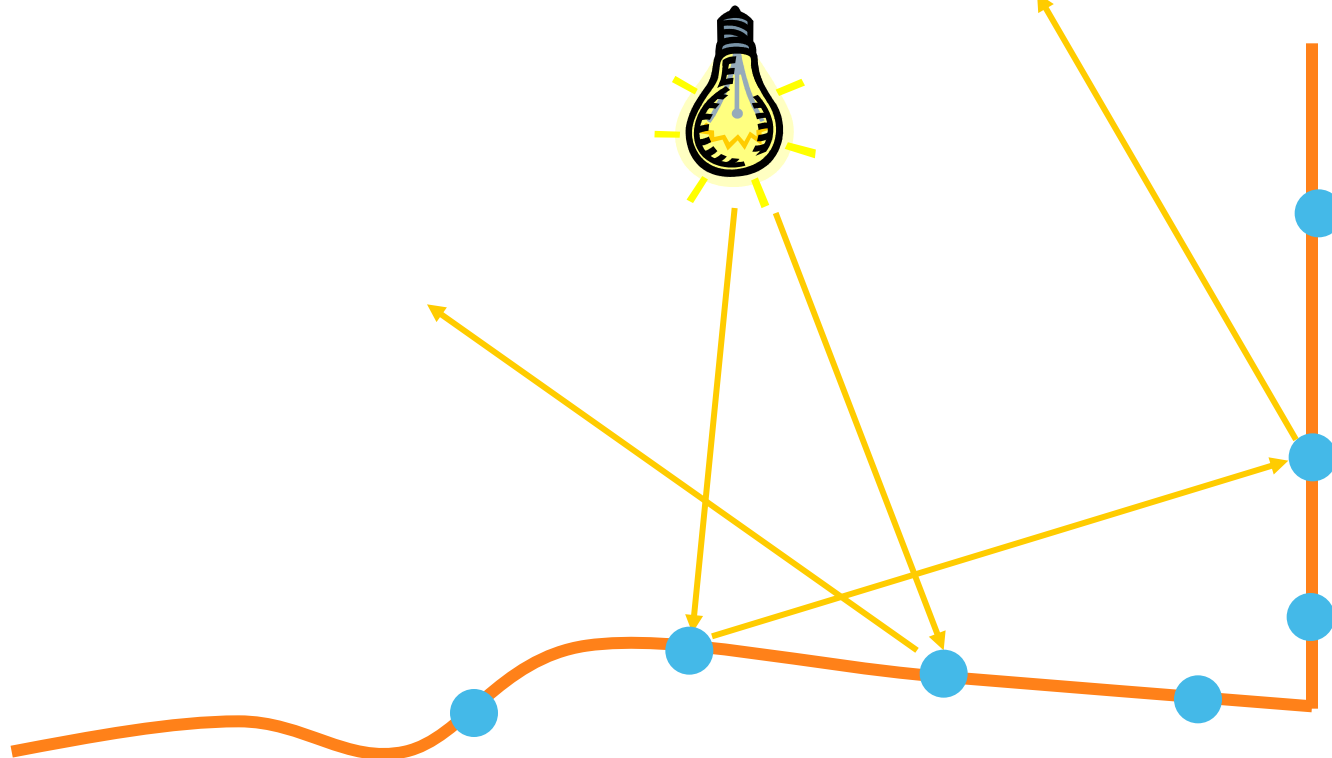
- ❑ Jako path tracing
- ❑ Místo rekurze dotaz do fotonové mapy



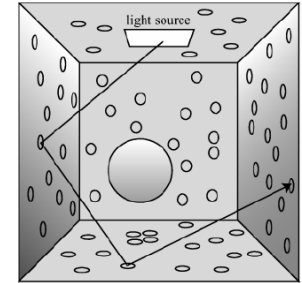
# Fáze 1: Rozmístění fotonů



1. **Emise** fotonů ze světelných zdrojů
2. **Sledování** fotonů
3. Uložení fotonů do „fotonové mapy“ (=seznam fotonů)



# Emise fotonů



## ■ Cíl

- Všechny emitované fotony nesou stejný (podobný) tok (variance odhadů osvětlení z fotonové mapy je pak nižší)

## 1. Emise jednoho fotonu (tj. světelné cesty)

### 1. Vyber světelný zdroj

- Náhodně s pravděpodobností úměrnou jeho celkovému toku

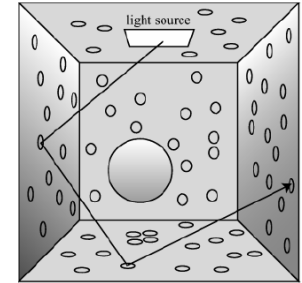
### 2. Vyber počátek emitovaného fotonu

- Pozice světla pro bodové zdroje
- Náhodně (nejčastěji uniformě) pro plošné zdroje světla

### 3. Vyber směr emitovaného fotonu

- Náhodně podle emisní distribuční funkce světla

# Emise fotonů



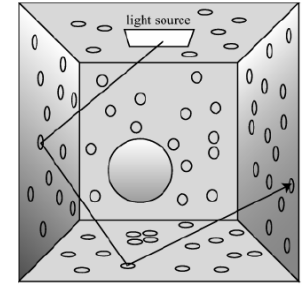
- Tok emitovaného fotonu:

$$\Phi_{p,0} = \frac{L_e(\mathbf{x}_0, \omega_0) |\cos \theta_0|}{P_l p(\mathbf{x}_0, \omega_0)}$$

(diskrétní) p-nost výběru  
světelného zdroje  $l$

hustota p-nosti vzorkování  
pozice  $\mathbf{x}_0$  a směru  $\omega_0$

# Emise fotonů



- Ideální vzorkování

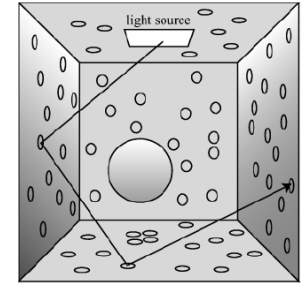
$$p(\mathbf{x}_0, \omega_0) = \frac{L_e(\mathbf{x}_0, \omega_0) |\cos \theta_0|}{\int_{A_{light}} \int_H L_e(\mathbf{x}_0, \omega_0) |\cos \theta_0| d\omega dA} = \frac{L_e(\mathbf{x}_0, \omega_0) |\cos \theta_0|}{\Phi_l}$$

$$P_l = \frac{\Phi_l}{\sum_{i \in lights} \Phi_i} = \frac{\Phi_l}{\Phi_{total}}$$

- Všechny emitované fotony pak nesou stejný tok:  $\Phi_{p,0} = \Phi_{total}$

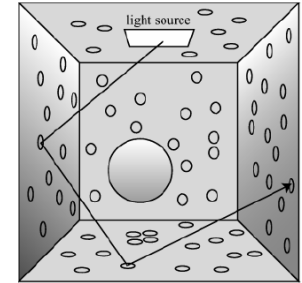


# Sledování fotonů (Photon tracing)



- Podobné jako light tracing
- Interakce foton - plocha
  1. Ulož „foton“ do fotonové mapy
    - foton = (pozice, příchozí směr, tok)
  2. Generuj směr odraženého paprsku
    - BRDF importance sampling
  3. Aktualizuj tok fotonu
    - ...
  4. Ruská ruleta – náhodná absorpce fotonu
    - ...
- Požadavek
  - Zachovat tok fotonu pokud možno konstantní

# Sledování fotonů (Photon tracing)



## 1. Aktualizuj tok fotonu

$$\Phi_{p,i+1}^{tentative} = \Phi_{p,i} \frac{f_r(\mathbf{x}, \omega_o \rightarrow \omega_i) |\cos \theta_o|}{p(\omega_o)}$$

## 2. Ruská ruleta – náhodná absorpce fotonu

$$q_{p,i+1} = \min \left\{ 1, \frac{\text{Luminance}[\Phi_{p,i+1}^{tentative}]}{\text{Luminance}[\Phi_{p,i}]} \right\}$$

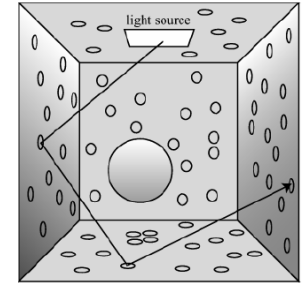
Pravděpodobnost  
přežití

$$\Phi_{p,i+1} = \frac{\Phi_{p,i+1}^{tentative}}{q_{p,i+1}}$$

Výsledný tok fotonu  
při přežití

- Tato strategie zachovává luminanci fotonů

# Sledování fotonů (Photon tracing)

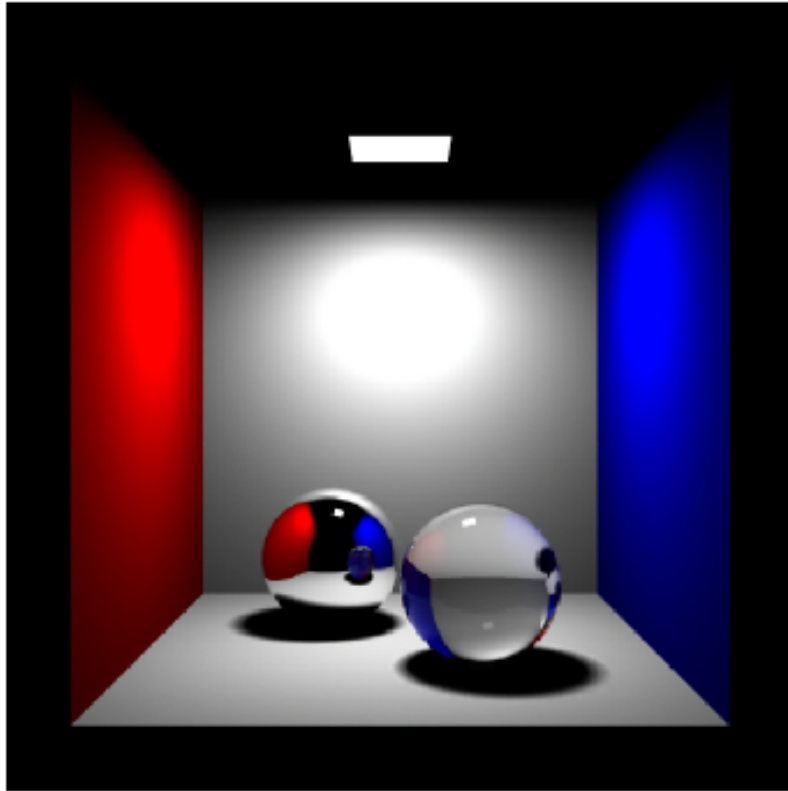


- Pozor na lom světla
  - ❑ při sledování cest od kamery je třeba při lomu změnit radianci podle druhé mocniny změny indexu lomu
  - ❑ avšak fotony nenesou radianci, nýbrž **tok** – žádná změna toku při lomu nenastává

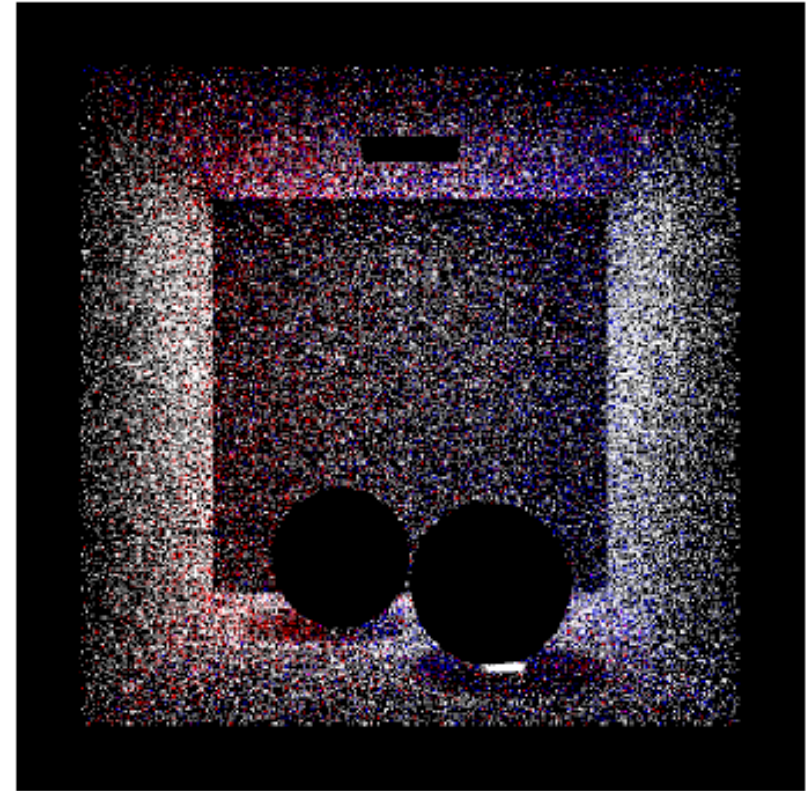
# Fotonová mapa

- Ukládání fotonů do fotonové mapy
  - Při každé interakci fotonu s difúzní (nebo mírně lesklou, ale ne zrcadlovou) plochou (i při absorpci!!!)
- Fotonová mapa
  - Během rozmísťování fotonů pouze lineární seznam fotonů
  - Po rozmístění všech fotonů se postaví *kD*-strom pro rychlejší vyhledávání
- Foton
  - pozice:  $\mathbf{x}_p = (x, y, z)$
  - příchozí směr:  $\omega_p = (\theta, \phi)$  (stačí char [ 2 ])
  - energie (tok):  $\Phi_p = (r, g, b)$  (stačí RGBE: char [ 4 ])
- Počet fotonů – cca  $10^5 - 10^7$  stačí pro většinu scén

# Fotonová mapa



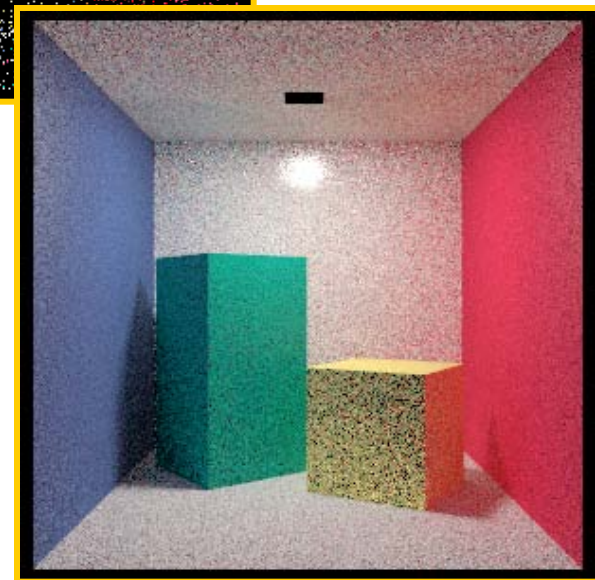
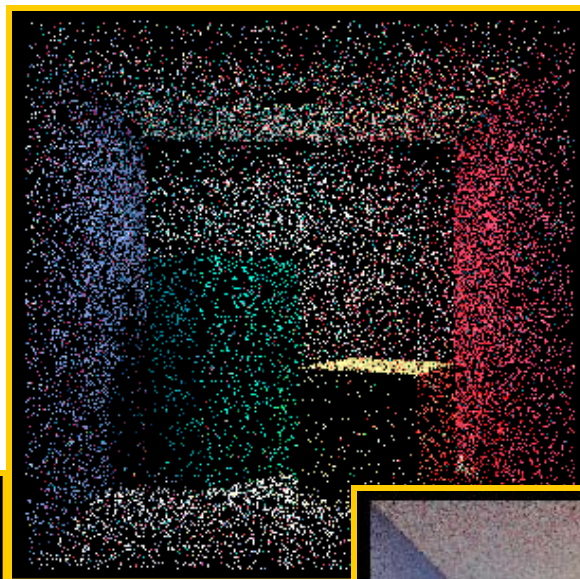
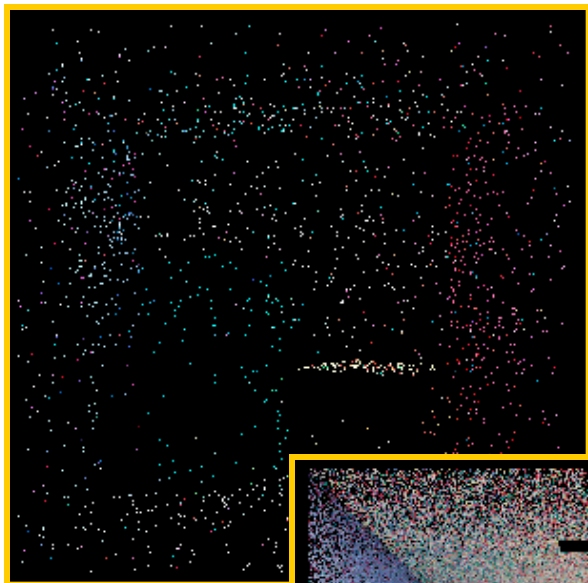
(a)



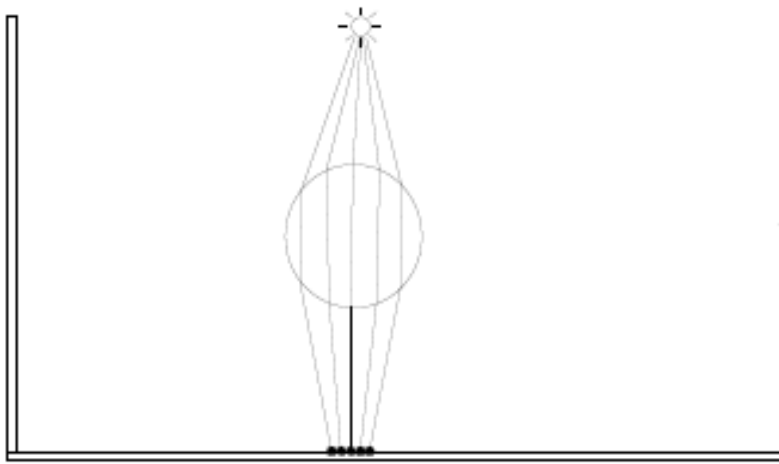
(b)

Figure 2.4: “Cornell box” with glass and chrome spheres: (a) ray traced image (direct illumination and specular reflection and transmission), (b) the photons in the corresponding photon map.

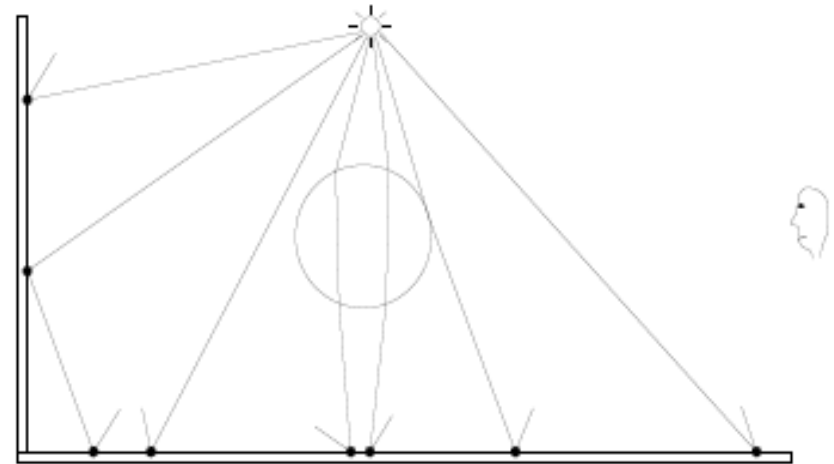
# Fotony reprezentují rovnovážnou radianci ve scéně



# Dvě fotonové mapy



Mapa kaustik



Globální mapa

# Dvě fotonové mapy

## 1. Globální mapa: **L[S|D]\*D**

- Obsahuje i přímé osvětlení

## 2. Mapa kaustik: **LS<sup>+</sup>D**

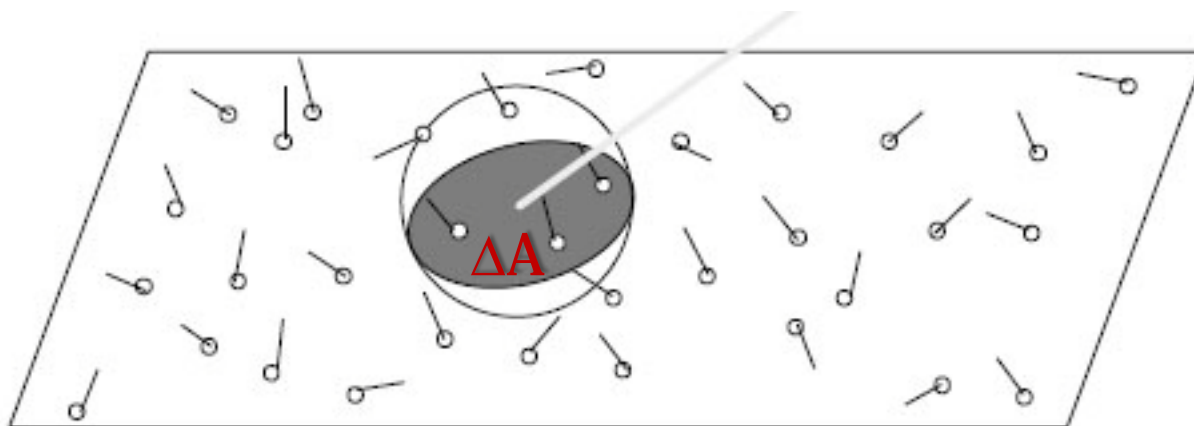
- Obsahuje pouze nepřímé osvětlení
- Je podmnožinou globální mapy
- Různé použití obou map při generování obrázku
  - Je lepší udržovat je zvlášť
- Regulární gramatika světelných cest  
E ... eye, L ... light, D ... diffuse, S ... specular  
G ... glossy (často se zahrnuje do D)



# Příprava fotonových map pro rendering

- Při trasování – přidávání fotonů do lineárního seznamu
- Poté postavení *kD-stromu*
  - Pro rendering potřebuji rychlé hledání *k* nejbližších fotonů

# Odhad radiance z fotonové mapy



$$L_r(\mathbf{x}, \omega_o) \approx \frac{1}{N} \sum_{p=1}^k f_r(\mathbf{x}, \omega_p, \omega_o) \frac{\Phi_p(\mathbf{x}, \omega_p)}{\Delta A}$$
$$\approx \frac{1}{N} \frac{1}{\pi r^2} \sum_{p=1}^k f_r(\mathbf{x}, \omega_p, \omega_o) \Phi_p(\mathbf{x}, \omega_p)$$

obsah kruhové oblasti  $\Delta A$

# Odhad radiance z fotonové mapy

RadianceEstimate(x, wo):

```
Color L = (0,0,0);
```

```
int k = locateNearestPhotons(x, wo, n_max, nearest, r);
```

```
// 'nearest' is an array of k nearest photons to x
```

```
// r is the distance from x to the farthest of them
```

```
if ( k < 5 ) return L;
```

```
for p = 1 to k do
```

```
{
```

```
    if( dot ( nearest[p].wi, N) <= 0 ) continue;
```

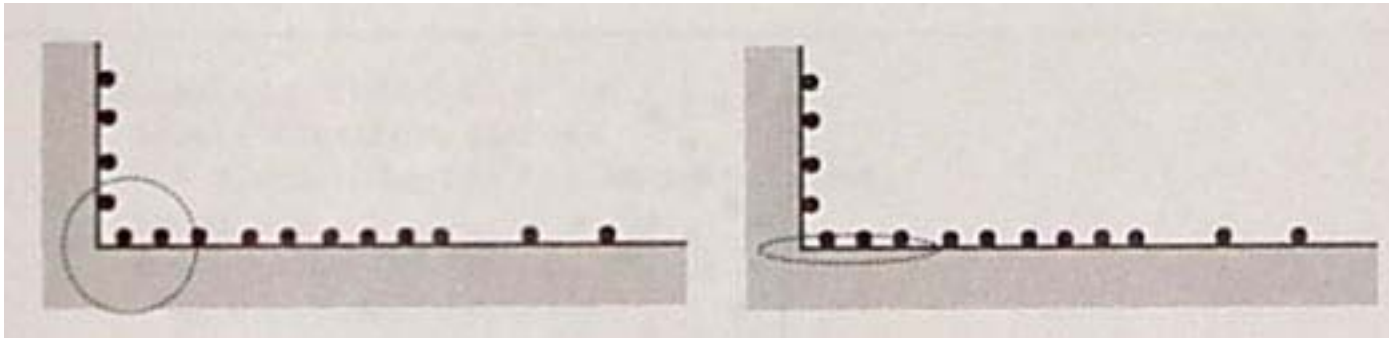
```
    L += fr(x, wo, nearest[p].wi) * nearest[p].flux;
```

```
}
```

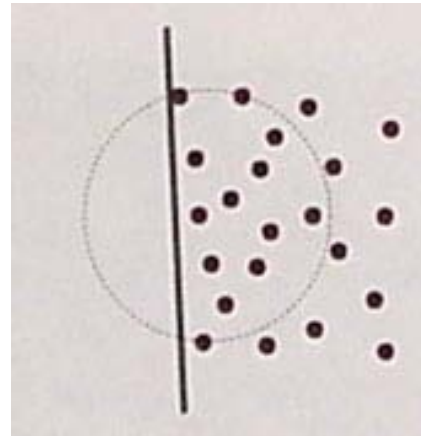
```
return L / (M_PI * r*r);
```

# Odhad radiance – problémy

- Zahrnutí nesprávných fotonů do odhadu



- Nesprávný odhad velikost  $\Delta A$ 
  - stěna, hrana kaustiky



# Rychlé hledání nejbližších fotonů

- Potřebuji pro odhad radiance z fotonové mapy
- Hledání  $k$  nejbližších fotonů je

*k-nearest neighbor search* (k-NN)

# ***k*-D strom – Konstrukce**

- Rekurzivní dělení podél osy s maximálním rozsahem
- Dělení
  - Dělicí rovina prochází přímo mediánem
  - Korespondující uzel stromu obsahuje medián
- Reprezentace stromu v lineárním poli, potomky fotonu na indexu  $i$  jsou na indexech  $2i$  a  $2i+1$

# ***k*-D strom – Hledání nejbližších sousedů**

- Ořezávání průchodu
  - Podle vzdálenosti již nalezeného  $K$ -tého nejbližšího fotonu (hledám-li  $K$  nejbližších)
  - Podle daného poloměru vyhledávání  $R$  (při hledání uvnitř fixního poloměru – „range query“)
- Dosud nalezené fotony se udržují v max-haldě

# Fáze 2: Rendering

- Sledování paprsku z kamery (distributed ray tracing)
  - Rekurze nahrazena odhadem radiance z fotonové mapy
  - Pro ideální zrcadlové plochy se stále používá rekurze jako v klasickém trasování paprsků



# Výpočet osvětlení pro primární paprsek (nebo po zrcadlovém odrazu)

- Odražená radiance

$$L_r(\mathbf{x}, \omega_o) = \int_{\Omega} \underbrace{L_i(\mathbf{x}, \omega_i)}_{\text{příchozí radiance}} \underbrace{f_r(\mathbf{x}, \omega_i \rightarrow \omega_o)}_{\text{BRDF}} \cos \theta_i d\omega_i$$

- Rozdělení příchozí radiance

$$L_i = L_{i,d} + L_{i,c} + L_{i,l}$$

- Rozdělení BRDF

$$f_r = f_{r,D} + f_{r,S}$$

PM ... photon map  
FG ... final gathering

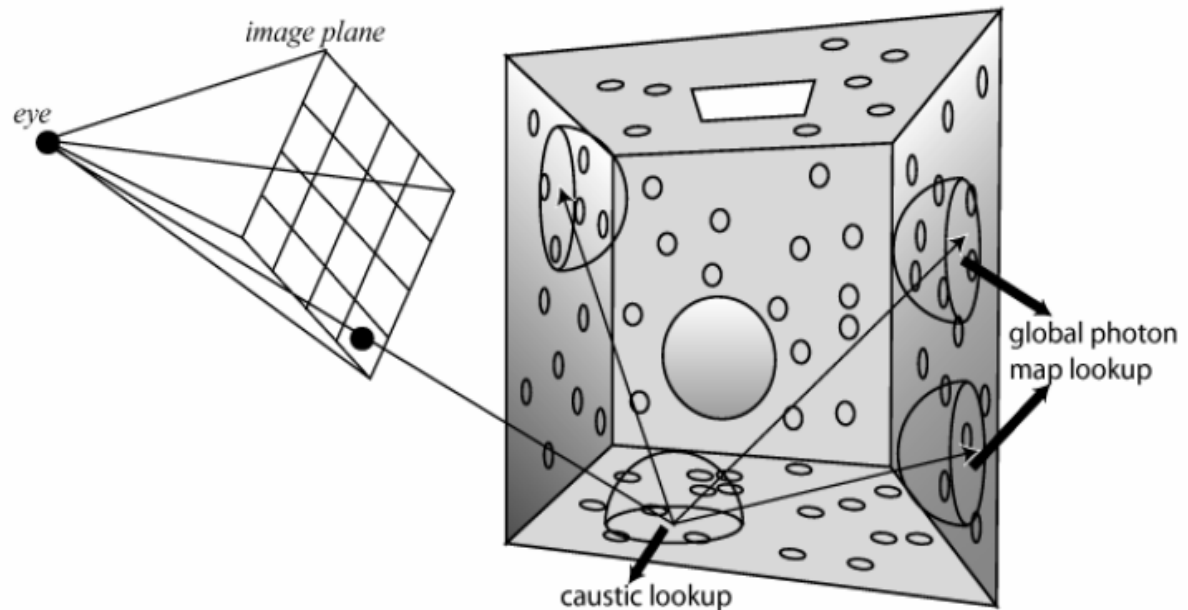
	$f_{r,D}$	$f_{r,S}$
$L_{i,d}$	přímé osvětlení	ideální zrcadlové odrazy/lomy
$L_{i,c}$	kaustiky (PM)	
$L_{i,l}$	difúzní nepřímé (FG + PM)	

# Výpočet osvětlení pro primární paprsek (nebo po zrcadlovém odrazu)

- Bez použití fotonových map
  - Přímé osvětlení
    - Jako obvykle: vzorkování světla + stínové paprsky
  - Ideální zrcadlové odrazy / lomy
    - Jako obvykle: deterministické sekundární paprsky
- S použitím fotonových map
  - ...

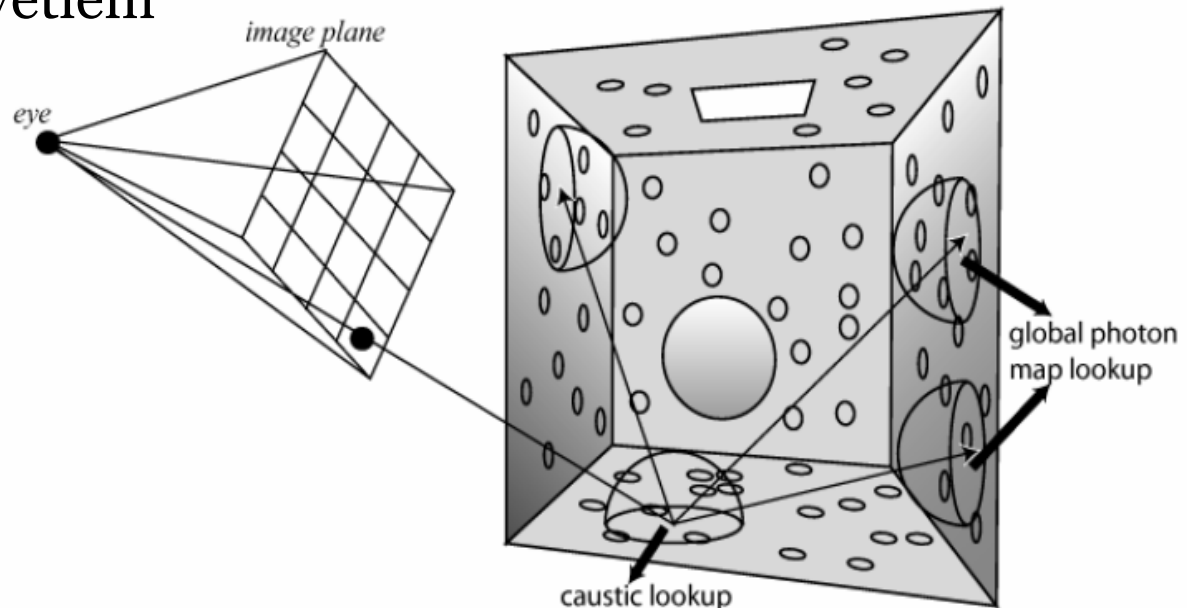
# Výpočet osvětlení pro primární paprsek (nebo po zrcadlovém odrazu)

- S použitím fotonových map
  - Kaustiky
    - Odhad radiance z fotonové mapy kaustik
  - Nepřímé osvětlení na difúzních a mírně lesklých plochách
    - Final gathering ...



# Final gathering (FG)

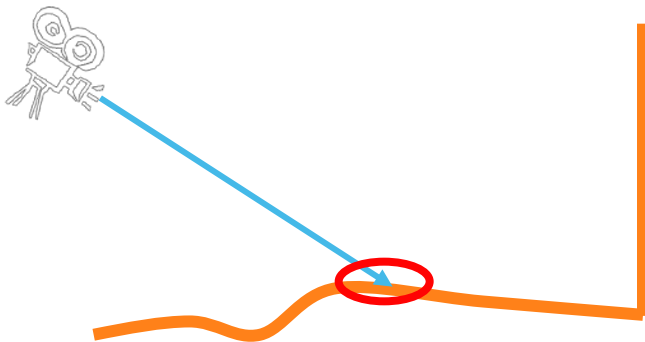
- Nepřímé osvětlení na difúzních a mírně lesklých plochách
- Jedna úroveň rekurze pomocí trasování rozprostřených paprsků (distribution ray tracing)
- Pro průsečíky sekundárních paprsků použij odhad radiance z globální fotonové mapy
  - Není potřeba explicitně počítat přímé osvětlení (je obsaženo v globální mapě )



# Proč potřebujeme final gathering?



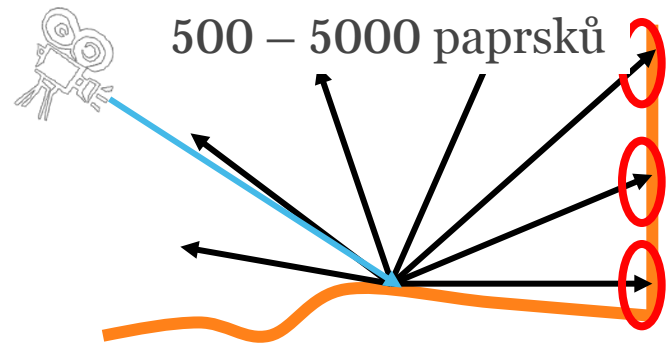
přímé použití



Informace v globální mapě  
příliš nepřesná



**final gathering**



nepřesnost v globální  
mapě se „zprůměruje“

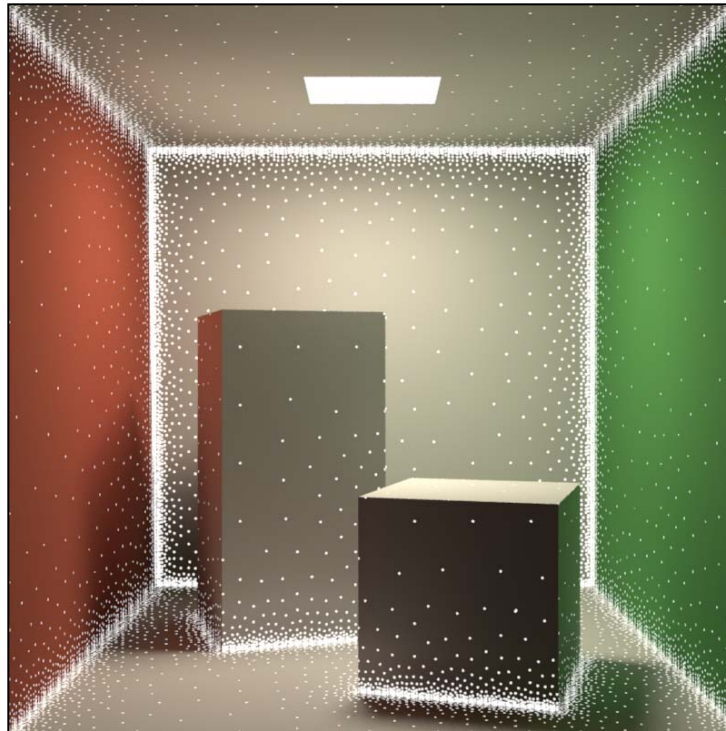
# Proč nepotřebujeme final gathering pro kaustiky?

- Kaustiky = zaostření světla => dostatečná hustota fotonů



# Možnosti urychlení final gatheringu

## 1. Irradiance caching



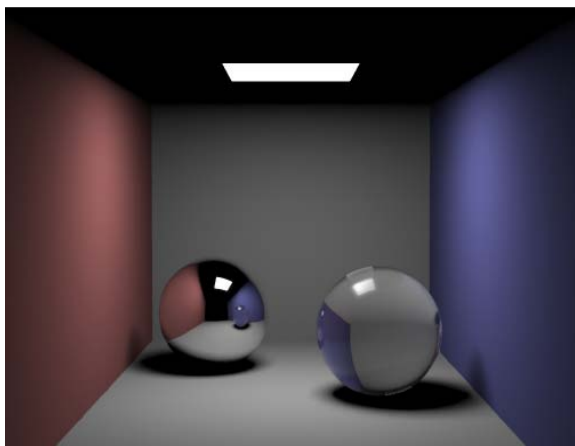
# Možnosti urychlení final gatheringu

## 2. Předvýpočet radiance na pozici fotonů

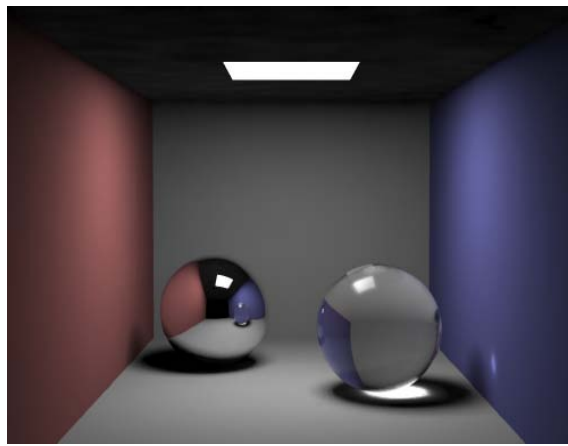
- Po rozmístění všech fotonů (fáze 1) se vybere podmnožina fotonů, na jejich pozicích se provede odhad radiance z fotonové mapy a výsledek se uloží do separátního kd-stromu.
- Při renderingu se pro sekundární paprsky z final gatheringu nemusí provádět k-NN dotaz do fotonové mapy => stačí pouze najít jeden nejbližší záznam s předpočítanou radiancí



# Výsledky

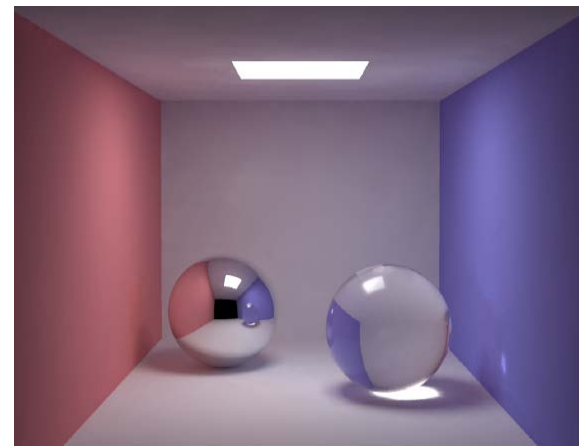


přímé osvětlení (21 s)



kaustiky (45 s)

50 000 fotonů  
v mapě kaustik

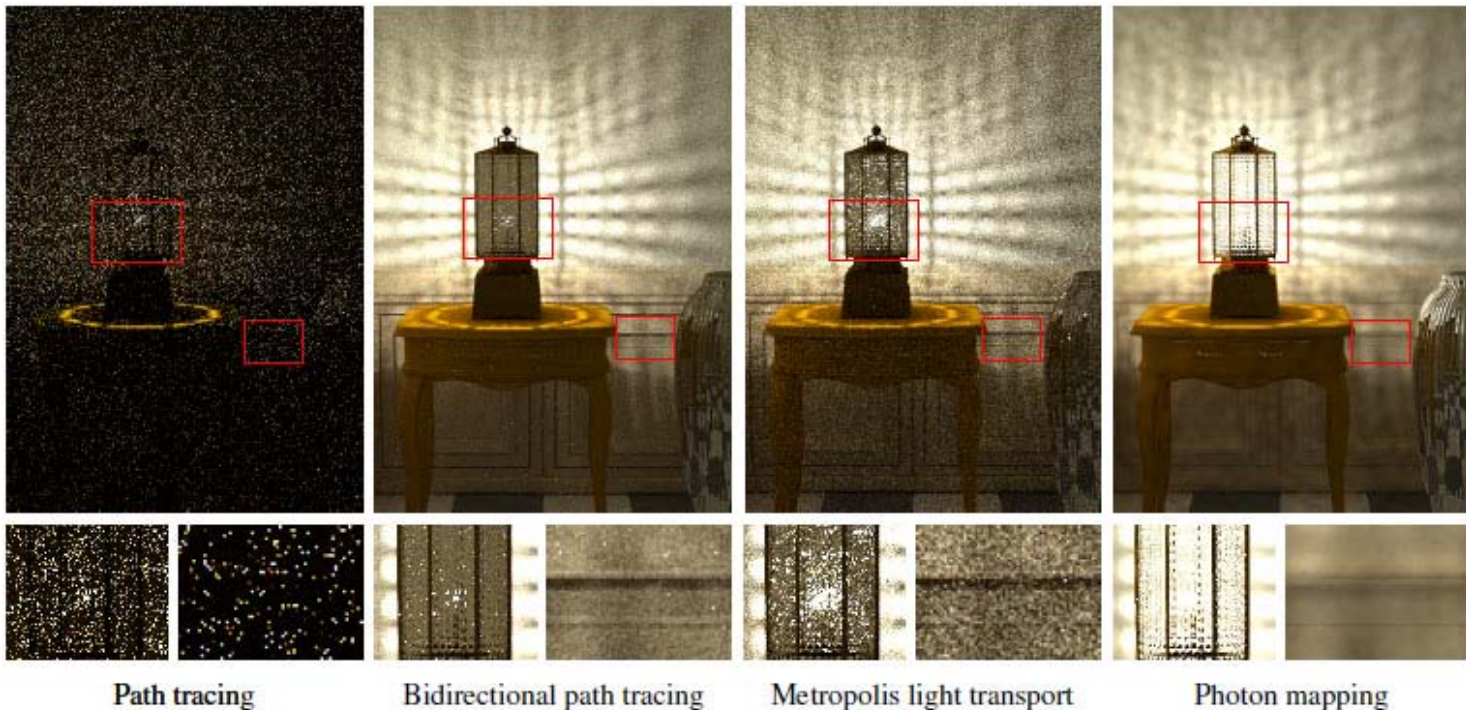


GI (66 s)

200 000 fotonů  
v globální mapě

# V čem vynikají fotonové mapy?

- Kaustiky
- SDS cesty
  - Klasické MC algoritmy nefungují (path tracing, bidirectional path tracing, metropolis light transport)



# SDS cesty – dno bazénu



© H.W.Jensen



© Wojciech Jarosz

# Lze rozšířit na přenos světla v médiu



Henrik Wann Jensen

**... a na rozptyl světla pod povrchem**



# Praktické nedostatky fotonových map

- Algoritmus nefunguje dobře na lesklých plochách



# Praktické nedostatky fotonových map

- Algoritmus nefunguje dobře na lesklých plochách
- Co je špatně?
  - Odhad radiance z fotonové mapy na lesklé ploše trpí vysokou variací
  - Sekundární paprsky z final gatheringu na lesklé ploše často dopadnou do podobného místa ve scéně => korelované výsledky odhadu radiance => vidíme “odraz” fotonové mapy (včetně všech nepřesností, které obsahuje)

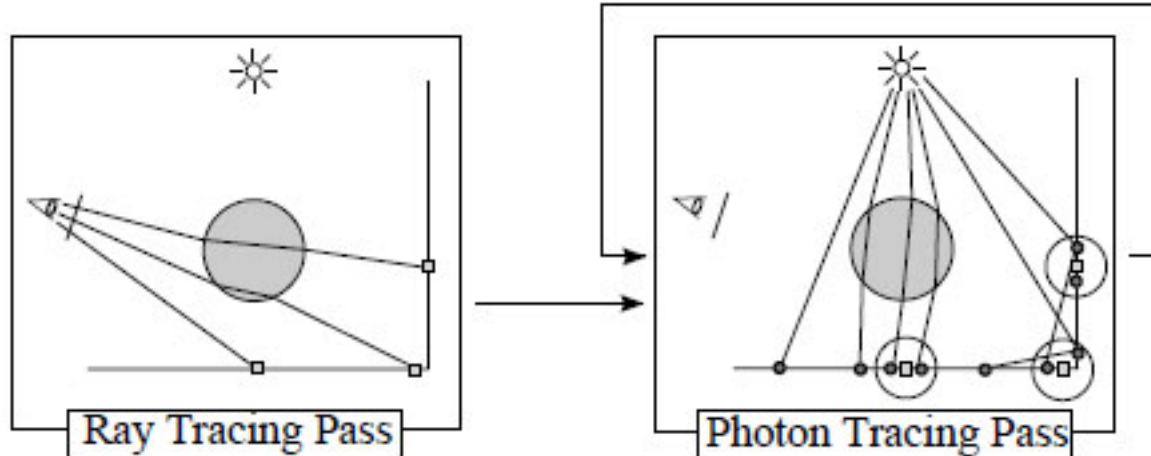
# Teoretické nedostatky fotonových map

- Výsledek **není nestranný**
  - obsahuje systematickou chybu
- Výsledek je **konzistentní**
  - Konverguje pro nekonečný počet fotonů
  - Toho ale prakticky není možné dosáhnout z důvodu omezené velikosti paměti
  - Řešení: **progressive photon mapping**

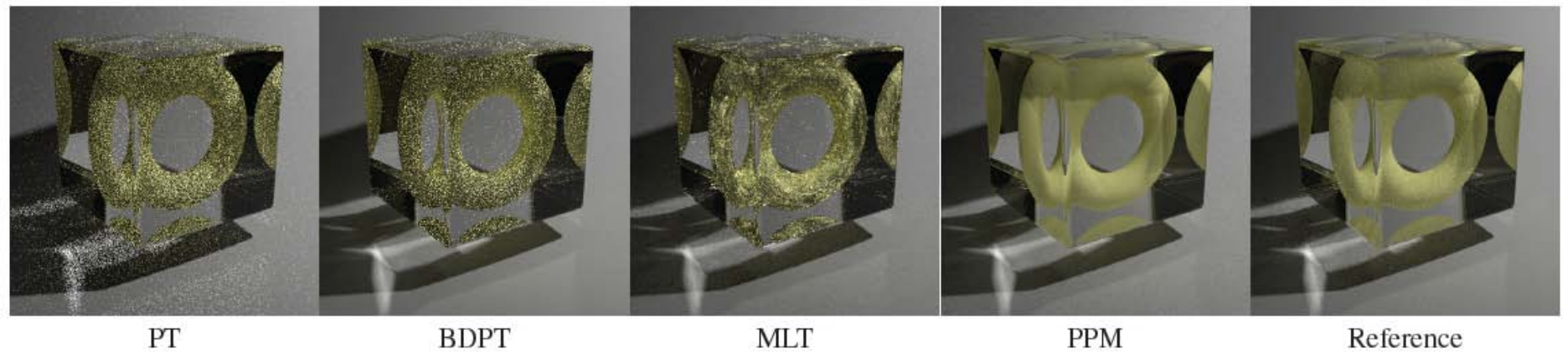


# Progresivní fotonové mapy

- Iterativní procedura
- V každé iteraci se zmenší radius pro hledání nejbližších fotonů, čímž se eliminuje systematická chyba (bias)



# Progresivní fotonové mapy



PT

BDPT

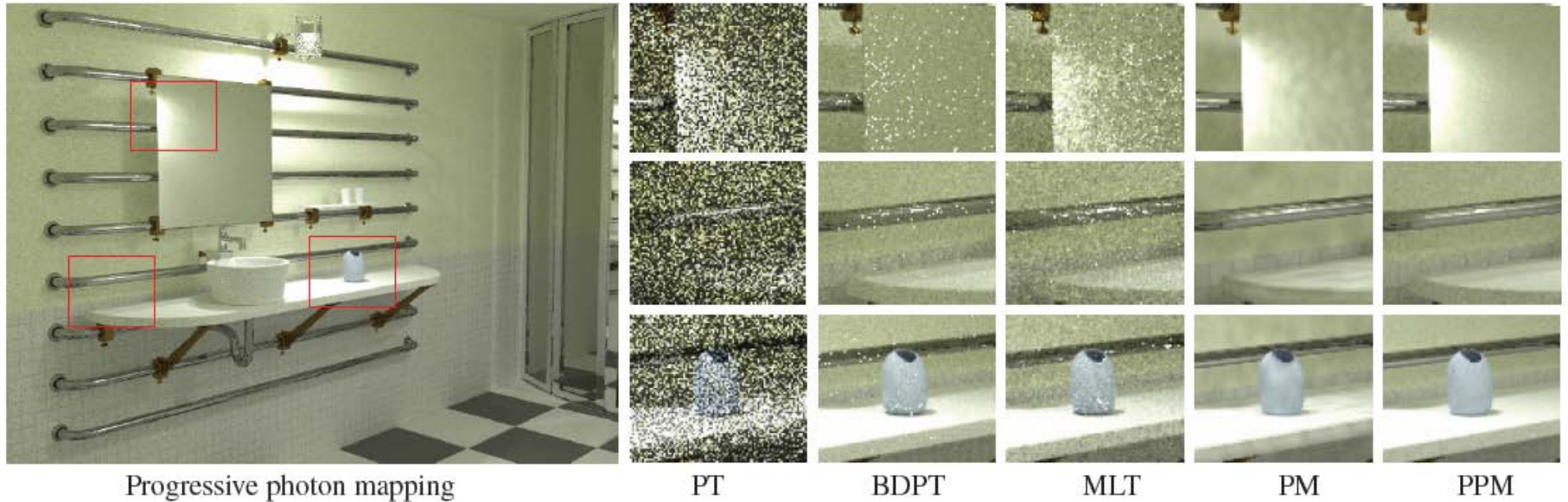
MLT

PPM

Reference

**Figure 7:** *Torus embedded in a glass cube. The reference image on the far right have been rendered using path tracing with 51500 samples per pixel. The Monte Carlo ray tracing methods fail to capture the lighting within the glass cube, while progressive photon mapping provides a smooth result using the same rendering time.*

# Progresivní fotonové mapy



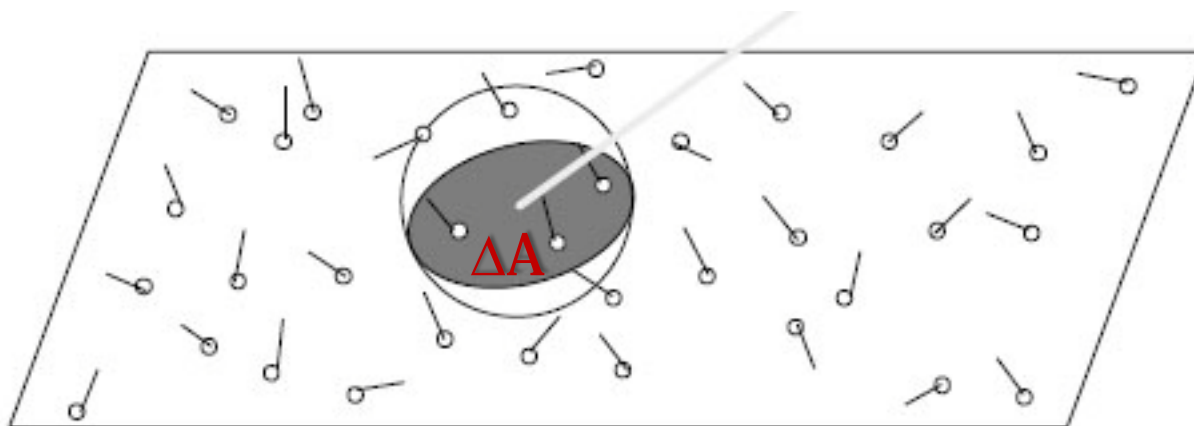
**Figure 8:** *Lighting simulation in a bathroom. The scene is illuminated by a small lighting fixture consisting of a light source embedded in glass. The illumination in the mirror cannot be resolved using Monte Carlo ray tracing. Photon mapping with 20 million photons results in a noisy and blurry image, while progressive photon mapping is able to resolve the details in the mirror and in the illumination without noise.*

---

# **Odvození odhadu radiance z fotonové mapy**

---

# Odhad radiance z fotonové mapy



$$L_r(\mathbf{x}, \omega_o) \approx \frac{1}{N} \sum_{p=1}^k f_r(\mathbf{x}, \omega_p, \omega_o) \frac{\Phi_p(\mathbf{x}, \omega_p)}{\Delta A}$$
$$\approx \frac{1}{N} \frac{1}{\pi r^2} \sum_{p=1}^k f_r(\mathbf{x}, \omega_p, \omega_o) \Phi_p(\mathbf{x}, \omega_p)$$

obsah kruhové oblasti  $\Delta A$

# Odhad radiance z fotonové mapy

- Odvození (intuitivní, ale formálně není ok)

$$\begin{aligned} L_r(\mathbf{x}, \omega_o) &= \int_{\Omega} f_r(\mathbf{x}, \omega_i, \omega_o) L_i(\mathbf{x}, \omega_i) \cos \theta_i d\omega_i \\ &= \int_{\Omega} f_r(\mathbf{x}, \omega_i, \omega_o) \frac{d^2 \Phi_i(\mathbf{x}, \omega_i)}{\cos \theta_i d\omega_i dA_i} \cos \theta_i d\omega_i \\ &= \int_{\Omega} f_r(\mathbf{x}, \omega_i, \omega_o) \frac{d^2 \Phi_i(\mathbf{x}, \omega_i)}{dA_i} \end{aligned}$$

# Odhad radiance z fotonové mapy

- Lze dokázat, že fotony rozmístěné podle popsané procedury představují **vzorky rovnovážné příchozí radiance**, tj. pro libovolné  $W_e$  platí:

$$E \left[ \frac{1}{N} \sum_{p=1}^N \Phi_p W_e(\mathbf{x}, \omega_i) \right] = \int_A \int_H W_e(\mathbf{x}, \omega_i) L_i(\mathbf{x}, \omega_i) \cos \theta_i d\omega_i dA$$

( $N$  je celkový počet fotonů.)

- „Tok“ fotonu zde (korektně) interpretujeme jako hodnotu vzorku příchozí radiance v daném bodě vydělenou PDF vygenerování tohoto vzorku (=fotonu):

$$\Phi_p = \frac{L_i(\mathbf{x}, \omega_i)}{\text{PDF}_p(\mathbf{x}, \omega_i)}$$

# Odhad radiance z fotonové mapy

- „Divný“ způsob zápisu integrálu pro održenou radianci:

$$L_r(\mathbf{x}, \omega_o) = \int \int_{A'H} \delta(\mathbf{x} - \mathbf{x}') f_r(\mathbf{x}', \omega_i, \omega_o) L_i(\mathbf{x}', \omega_i) \cos \theta_i d\omega_i dA'$$

- Tento integrál odhadujeme pomocí MC kvadratury, kde fotony představují použité vzorky
- **Problém:** Pravděpodobnost, že foton dopadne přesně do bodu  $\mathbf{x}$  je nulová.
- **Řešení:**  $\delta$ -distribuci nahradíme normalizovaným jádrem (kernel)  $k$  s konečným nosičem:

$$L_r(\mathbf{x}, \omega_o) \approx \int \int_{A'H} k(\|\mathbf{x} - \mathbf{x}'\|) f_r(\mathbf{x}', \omega_i, \omega_o) L_i(\mathbf{x}', \omega_i) \cos \theta_i d\omega_i dA'$$



# Odhad radiance z fotonové mapy

$$L_r(\mathbf{x}, \omega_o) \approx \int \int_{A'H} k(\|\mathbf{x} - \mathbf{x}'\|) f_r(\mathbf{x}', \omega_i, \omega_o) L_i(\mathbf{x}', \omega_i) \cos \theta_i d\omega_i dA'$$

- Výsledkem je „průměr“ radiance přes nosič jádra  $k$ .
- Náhrada  $\delta$ -distribuce jádrem  $k$  způsobuje systematickou chybu (bias) – charakteristické rozmazání radiance.
- Tento integrál již neobsahuje singularitu a tudíž ho lze řešit pomocí MC, kde fotony představují vzorky.

# Odhad radiance z fotonové mapy

- Chceme spočítat:

$$L_r(\mathbf{x}, \omega_o) \approx \int \int_{A'H} k(\|\mathbf{x} - \mathbf{x}'\|) f_r(\mathbf{x}', \omega_i, \omega_o) L_i(\mathbf{x}', \omega_i) \cos \theta_i d\omega_i dA'$$

- Víme, že pro libovolné  $W_e$  platí:

$$E \left[ \frac{1}{N} \sum_{p=1}^N \Phi_p W_e(\mathbf{x}, \omega_i) \right] = \int \int_{A'H} W_e(\mathbf{x}, \omega_i) L_i(\mathbf{x}, \omega_i) \cos \theta_i d\omega_i$$

- Položíme tedy  $W_e(\mathbf{x}', \omega_o) = k(\|\mathbf{x} - \mathbf{x}'\|) f_r(\mathbf{x}', \omega_i, \omega_o)$

- A dostaneme

$$L_r(x, \omega_i) \approx E \left[ \frac{1}{N} \sum_{p=1}^N \Phi_p k(\|\mathbf{x} - \mathbf{x}'\|) f_r(\mathbf{x}', \omega_i, \omega_o) \right]$$

# Odhad radiance z fotonové mapy

- Konstantní jádro:

$$k(\|\mathbf{x} - \mathbf{x}'\|) = \begin{cases} \frac{1}{\pi r^2} & \text{pro } \|\mathbf{x} - \mathbf{x}'\| \leq r \\ 0 & \text{jinak} \end{cases}$$

kde  $r$  je velikost jádra, kterou nastavíme na vzdálenost ke  $k$ -tému nejbližšímu fotonu.

- Dostáváme

$$L_r(x, \omega_i) \approx \frac{1}{N} \frac{1}{\pi r^2} \sum_{p=1}^k \Phi_p f_r(\mathbf{x}', \omega_i, \omega_o)$$

c.b.d.

# Literatura

- H.W.Jensen, Realistic Image Synthesis using Photon Mapping, AK Peters, 2001
- T. Hachisuka, S. Ogaki and H. W. Jensen, **Progressive Photon Mapping**, ACM Transactions on Graphics (SIGGRAPH Asia 2008).