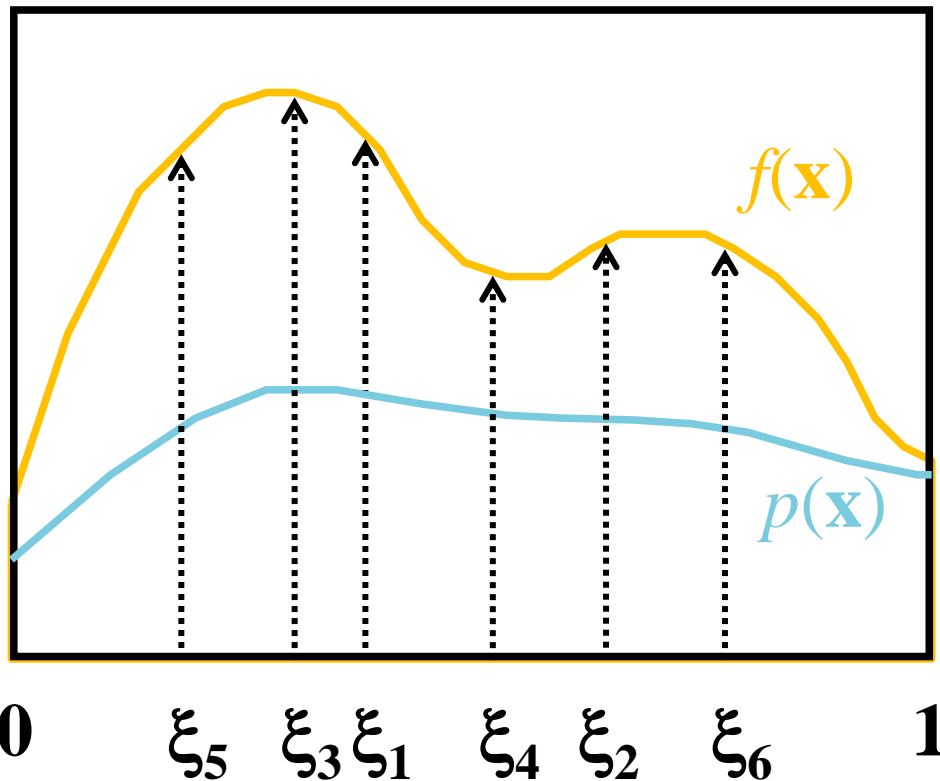

Počítačová grafika III – Monte Carlo integrování II

Jaroslav Křivánek, MFF UK

Jaroslav.Krivanek@mff.cuni.cz

Monte Carlo integrování

- Obecný nástroj k numerickému odhadu určitých integrálů



Integrál:

$$I = \int f(\mathbf{x})d\mathbf{x}$$

Monte Carlo odhad I :

$$\langle I \rangle = \frac{1}{N} \sum_{i=1}^N \frac{f(\xi_i)}{p(\xi_i)}; \quad \xi_i \propto p(\mathbf{x})$$

„V průměru“ to funguje:

$$E[\langle I \rangle] = I$$

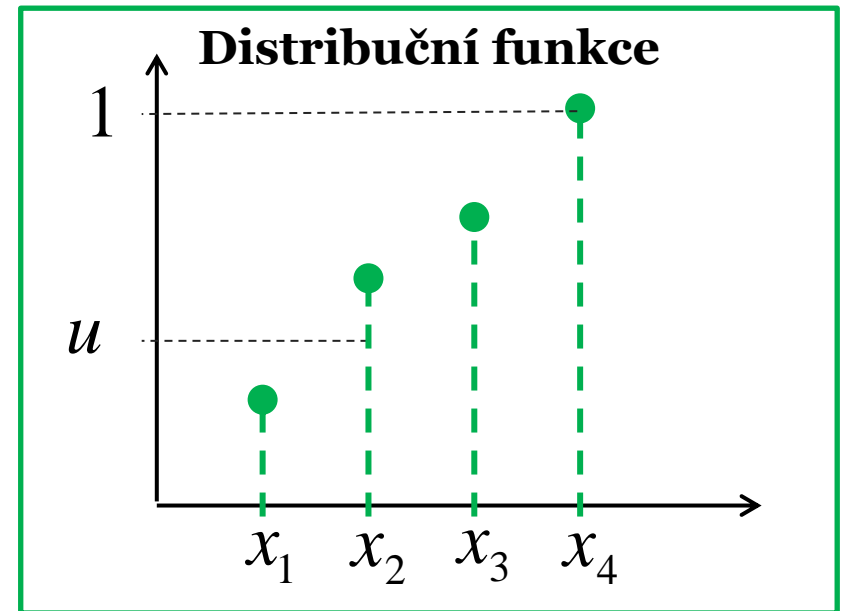
Generování vzorků z distribuce

1D diskrétní náhodná veličina

- Dána p-nostní fce $p(i)$, distribuční fce $P(i)$
- Postup
 1. Vygeneruj u z $R(0,1)$
 2. Vyber x_i pro které

$$P(i-1) < u \leq P(i)$$

(definujeme $P(0) = 0$)

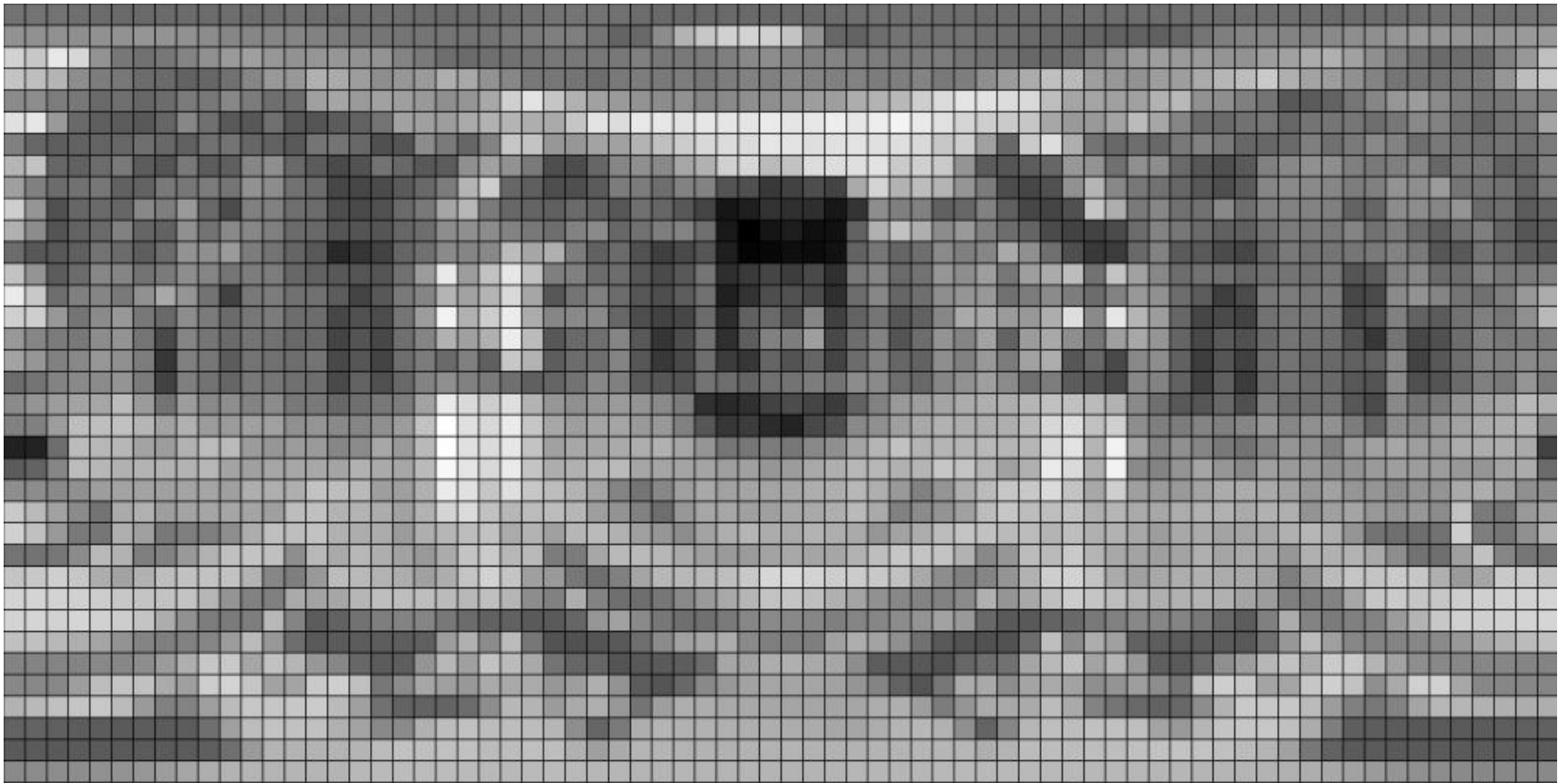


- Nalezení i se provádí půlením intervalu

2D diskrétní náhodná veličina

- Dána p-nostní fce $p_{I,J}(i, j)$
- Možnost 1:
 - Interpretovat jako 1D vektor pravděpodobností
 - Vzorkovat jako 1D distribuci

2D diskrétní náhodná veličina



2D diskretní náhodná veličina

■ Možnost 2 (lepší)

1. „Sloupec“ i_{sel} vybrat podle marginálního rozdělení, popsaného 1D marginální p-nostní fčí

$$p_I(i) = \sum_{j=1}^{n_j} p_{I,J}(i, j)$$

2. „Řádek“ j_{sel} vybrat podle podmíněného rozdělení příslušejícího vybranému „sloupci“ i_{sel}

$$p_{J|I}(j | I = i_{\text{sel}}) = \frac{p_{I,J}(i_{\text{sel}}, j)}{p_I(i_{\text{sel}})}$$

Vzorkování 1D spojité náhodné veličiny

- Transformací rovnoměrné náhodné veličiny
- Zamítací metoda (rejection sampling)

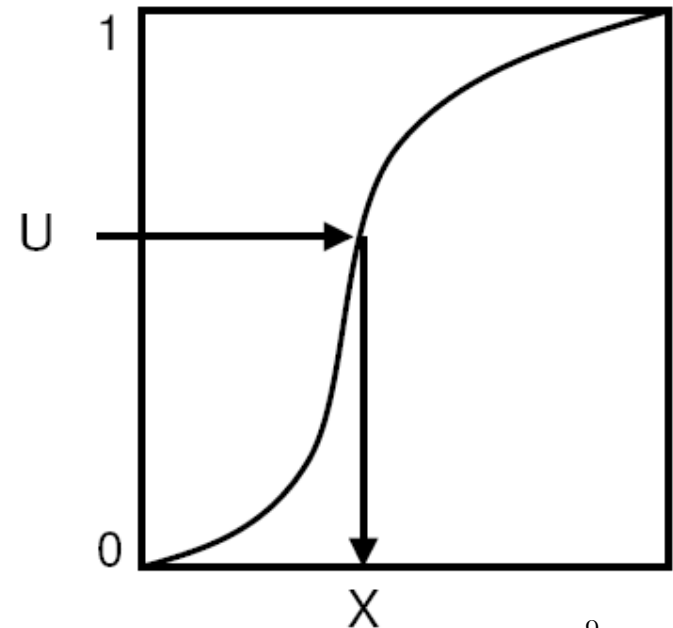
Vzorkování 1D spojité náhodné veličiny transformací

- Je-li U je náhodná veličina s rozdělením $R(0,1)$, pak náhodná veličina X

$$X = P^{-1}(U)$$

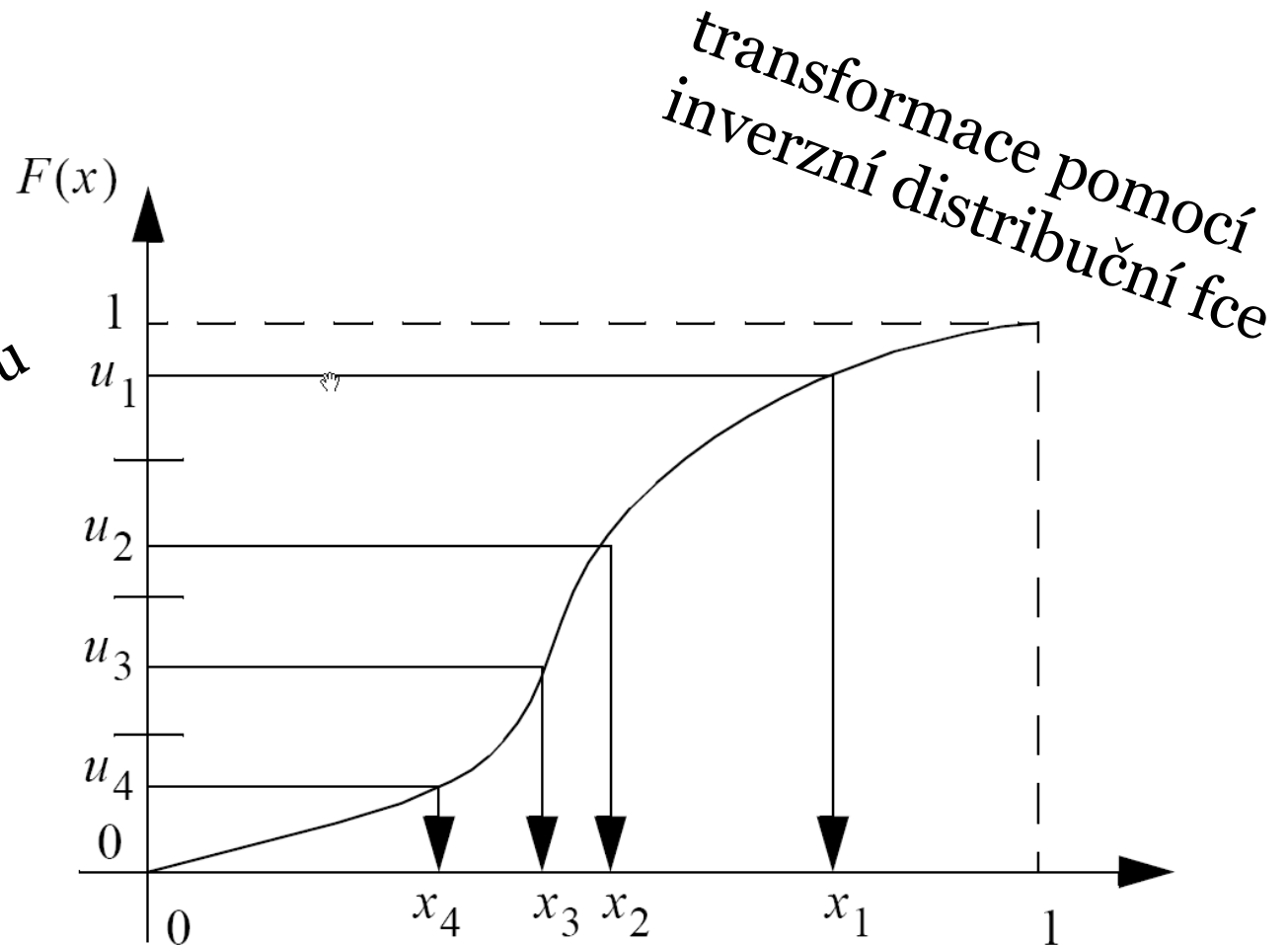
má rozdělení popsané distribuční funkcí P .

- Pro generování vzorků podle hustoty p potřebujeme
 - Spočítat cdf $P(x)$ z pdf $p(x)$
 - Spočítat inverzní funkci $P^{-1}(x)$



Kombinace vzorkování po částech s transformační metodou

stratifikace v prostoru
náhodných čísel



Vzorkování 1D spojité náhodné veličiny zamítací metodou

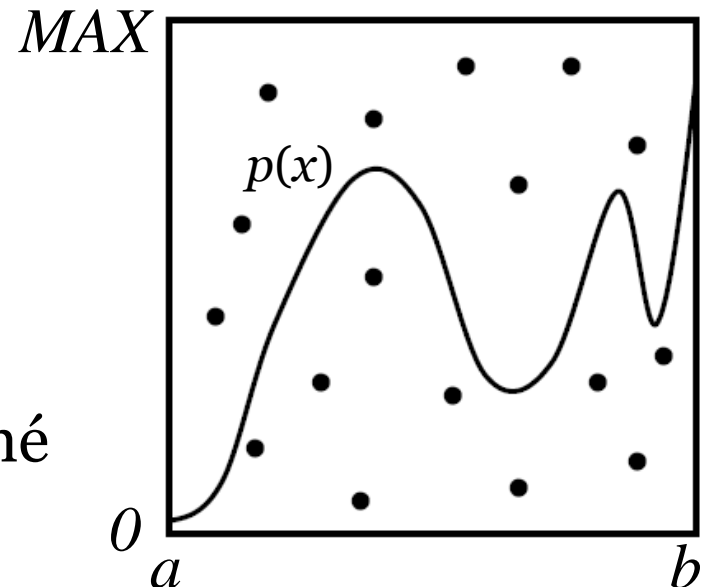
■ Algoritmus

- ❑ Vyber náhodné u_1 z $R(a, b)$
- ❑ Vyber náhodné a u_2 z $R(0, MAX)$
- ❑ Přijmi vzorek, pokud $p(u_1) > u_2$

■ Přijaté vzorky mají rozložení dané hustotou $p(x)$

■ Účinnost = % přijatých vzorků

- ❑ Plocha funkce pod křivkou / plocha obdélníka
- ❑ Transformační metoda vždy efektivnější (ale vyžaduje integrovat hustotu a invertovat distribuční fci)



Vzorkování 2D spojité náhodné veličiny

- Jako pro 2D diskrétní veličinu
- Dána sdružená hustota $p_{X,Y}(x, y) = p_X(x) p_{Y|X}(y | x)$
- Postup
 1. Vyber x_{sel} z **marginální hustoty**

$$p_X(x) = \int p_{X,Y}(x, y) dy$$

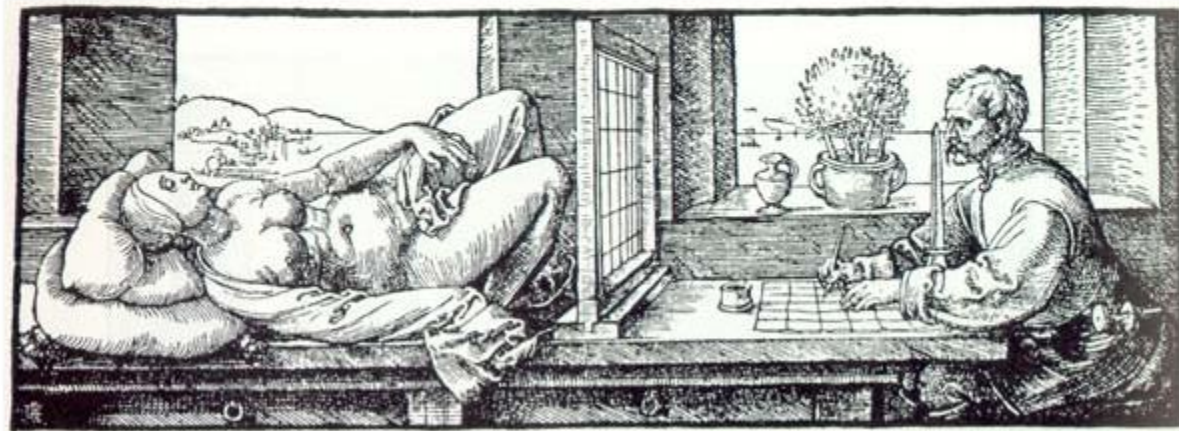
2. Vyber y_{sel} z **podmíněné hustoty**

$$p_{Y|X}(y | X = x_{sel}) = \frac{p_{X,Y}(x_{sel}, y)}{p_X(x_{sel})}$$

Transformační vzorce

- P. Dutré: **Global Illumination Compendium**, <http://people.cs.kuleuven.be/~philip.dutre/GI/>

Global Illumination Compendium The Concise Guide to Global Illumination Algorithms



Albrecht Dürer, *Underweysung der Messung mit dem Zirkel und Richtscheyt* (Nuremberg, 1525), Book 3, figure 67.

Importance sampling Phongovy BRDF

- Paprsek dopadne na plochu s Phongovou BRDF. Jak vygenerovat sekundární paprsek pro vzorkování nepřímého osvětlení?
- Path tracing
 - Pouze 1 sekundární paprsek – je třeba zvolit komponentu BRDF (druh interakce)
 - Postup:
 1. Vyber komponentu BRDF (difúzní odraz / lesklý odraz / lom)
 2. Vzorkuj vybranou komponentu

Fyzikálně věrohodná Phongova BRDF

$$f_r^{\text{Phong}}(\omega_i \rightarrow \omega_o) = \frac{\rho_d}{\pi} + \frac{n+2}{2\pi} \rho_s \cos^n \theta_r$$

- Kde:

$$\cos \theta_r = \omega_o \cdot \omega_r$$

$$\omega_r = 2(\omega_i \cdot \mathbf{n})\mathbf{n} - \omega_i$$

- Zachování energie: $\rho_d + \rho_s \leq 1$

Výběr interakce

```
pd = max(rhoD.r, rhoD.g, rhoD.b);  
ps = max(rhoS.r, rhoS.g, rhoS.b);  
pd /= (pd + ps);    // pravd. výběru difúzní komponenty  
ps /= (pd + ps);    // pravd. výběru lesklé komponenty
```

```
Vec3 dir, float pdf, Col brdfVal;  
if (rand(0,1) <= pd)  
    {dir, pdf, brdfVal} = sampleDiffuse();  
    return {dir, pdf * pd, brdfVal}  
  
else  
    {dir, pdf, brdfVal} = sampleSpecular();  
    return {dir, pdf * ps, brdfVal}
```


Vzorkování difúzního odrazu

- Importance sampling s hustotou $p(\theta) = \cos(\theta) / \pi$
 - θ ...úhel mezi normálou a vygenerovaným sekundárním paprskem
 - Generování směru:

$$\begin{aligned}\varphi &= 2\pi r_1 & x &= \cos(2\pi r_1) \sqrt{1-r_2} \\ \theta &= \arccos(\sqrt{r_2}) & y &= \sin(2\pi r_1) \sqrt{1-r_2} \\ & & z &= \sqrt{r_2}\end{aligned}$$

- r_1, r_2 ... uniformní na $\langle 0,1 \rangle$
- Zdroj: Dutre, Global illumination Compendium (on-line)
- Odvození: Pharr & Huphreys, PBRT

sampleDiffuse()

```
// build a local coordinate frame with N = z-axis
Vec3 U = arbitraryNormal(N); // U is perpendicular to the normal N
Vec3 V = crossProd(N, U);    // orthonormal basis with N and U

// generate direction in the local coordinate frame
float r1 = rand(0,1), r2 = rand(0,1);
float sinTheta = sqrt(1 - r2);
float cosTheta = sqrt(r2);
float phi      = 2.0*PI*r1;
float pdf      = cosTheta/PI;
// convert [theta, phi] to Cartesian coordinates
Vec3 locDir (cos(phi)*sinTheta, sin(phi)*sinTheta, cosTheta);

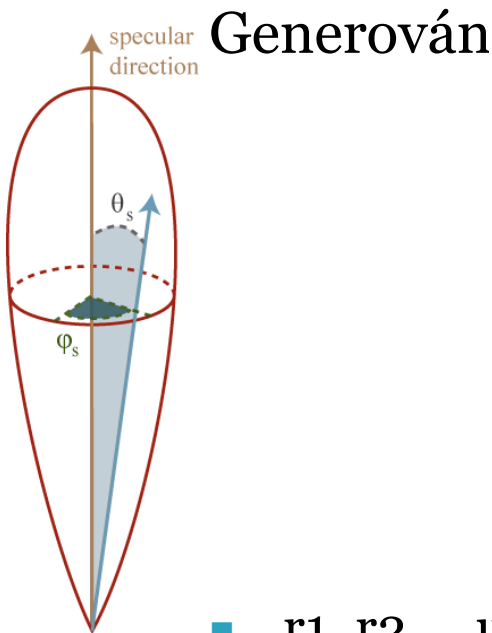
// transform ldir to the global coordinate frame
Vec3 globDir = locDir.x * U + locDir.y * V + locDir.z * N

// evaluate BRDF component
Col brdfVal = rhoD / PI;

return {globDir, pdf, brdfVal}
```

Vzorkování lesklého odrazu

- Importance sampling s hustotou $p(\theta) = (n+1)/(2\pi) \cos^n(\theta)$
 - θ ...úhel mezi ideálně zrcadlově odraženým ω_o a vygenerovaným sekundárním paprskem



$$\varphi = 2\pi r_1$$

$$\theta = \arccos\left(r_2^{\frac{1}{n+1}}\right)$$

$$x = \cos(2\pi r_1) \sqrt{1 - r_2^{\frac{2}{n+1}}}$$

$$y = \sin(2\pi r_1) \sqrt{1 - r_2^{\frac{2}{n+1}}}$$

$$z = r_2^{\frac{1}{n+1}}$$

- r_1, r_2 ... uniformní na $\langle 0,1 \rangle$

sampleSpecular()

```
// build a local coordinate frame with R = z-axis
Vec3 R = 2*dot(N,wi)*N - wi;           // ideal reflected direction
Vec3 U = arbitraryNormal(R);          // U is perpendicular to R
Vec3 V = crossProd(R, U);             // orthonormal basis with R and U

// generate direction in local coordinate frame
{Vec3 locDir, float pdf} = rndHemiCosN (n); // formulas form prev. slide

// transform locDir to global coordinate frame
Vec3 globDir = locDir.x * U + locDir.y * V + locDir.z * R

// return zero BRDF value if the generated direction is
// under the tangent plane
float cosThetaIn = dot(N, globDir);
if(cosThetaIn <= 0) return {globDir, pdf, Col(0)};

// evaluate the BRDF component
Col brdfVal = rhoS * (n+2)/(PI*2) * pow(locDir.z, n); //ldir.z = cosThetaR

return {globDir, pdf, brdfVal}
```

Image-based lighting

Image-based lighting

- Introduced by Paul Debevec (Siggraph 98)
- Routinely used for special effects in films & games

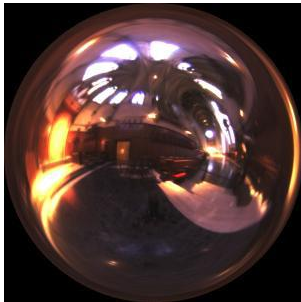
Image-based lighting

- Illuminating CG objects using measurements of real light (=light probes)

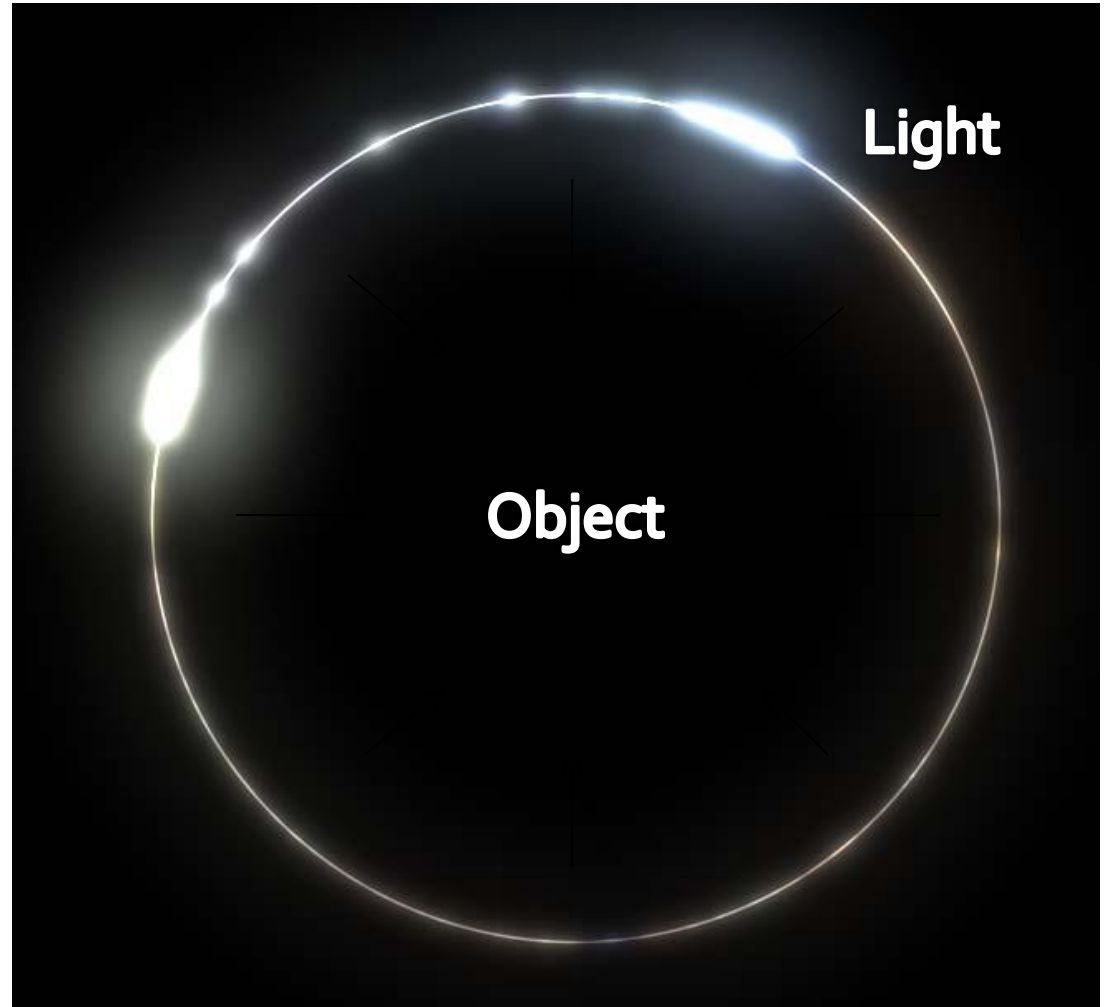
Eucaliptus
grove



Grace
cathedral



Uffizi
gallery



P

Point lighting

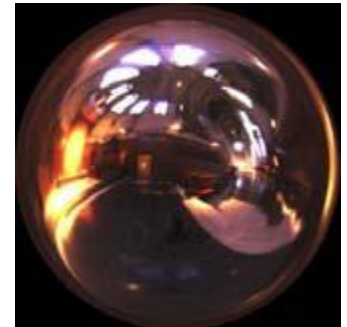
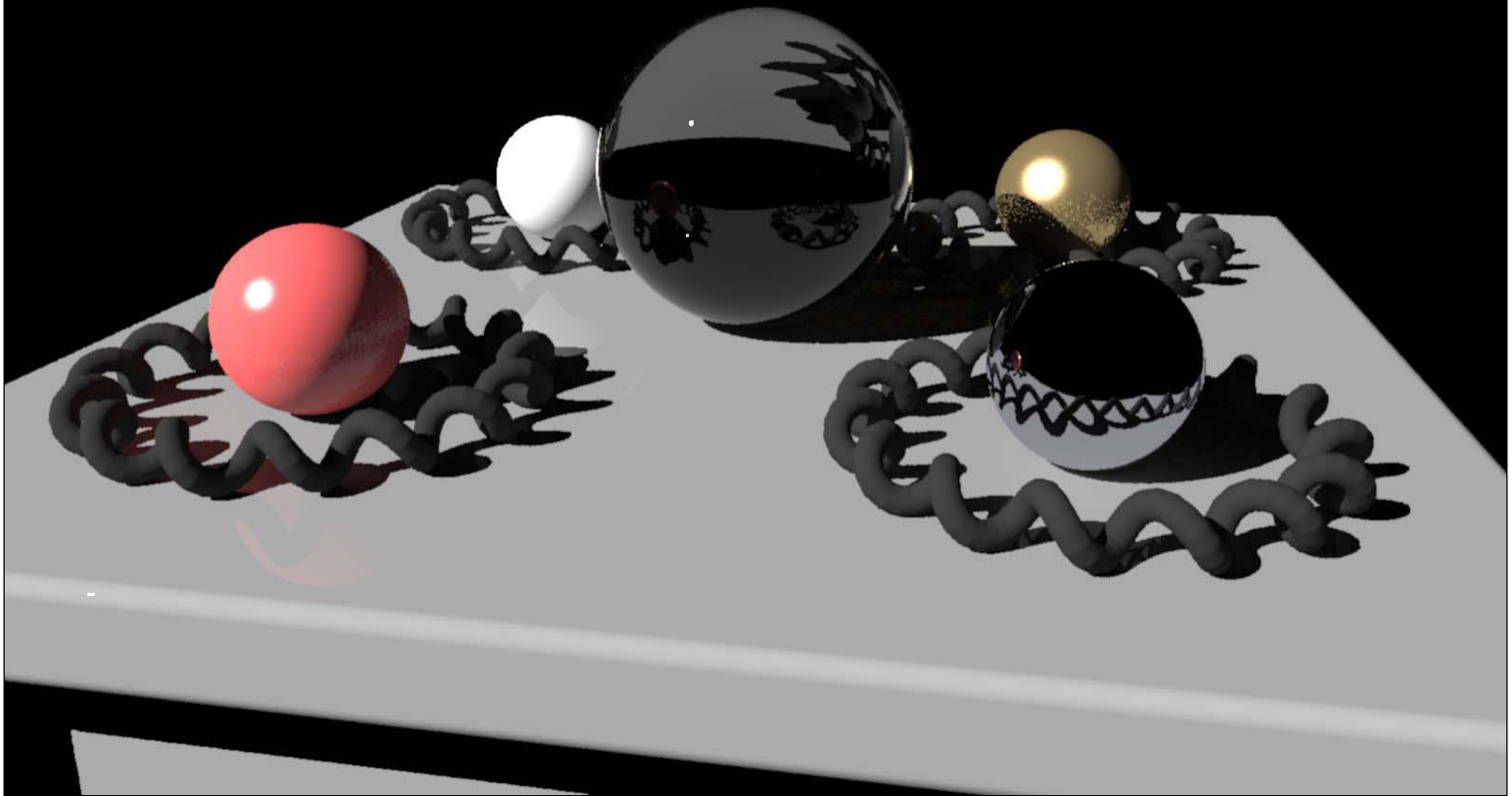


Image-based lighting

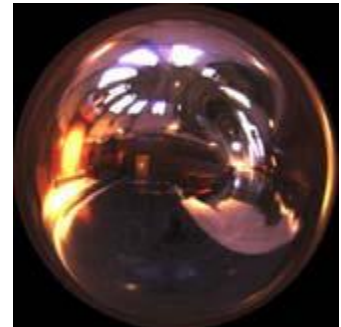
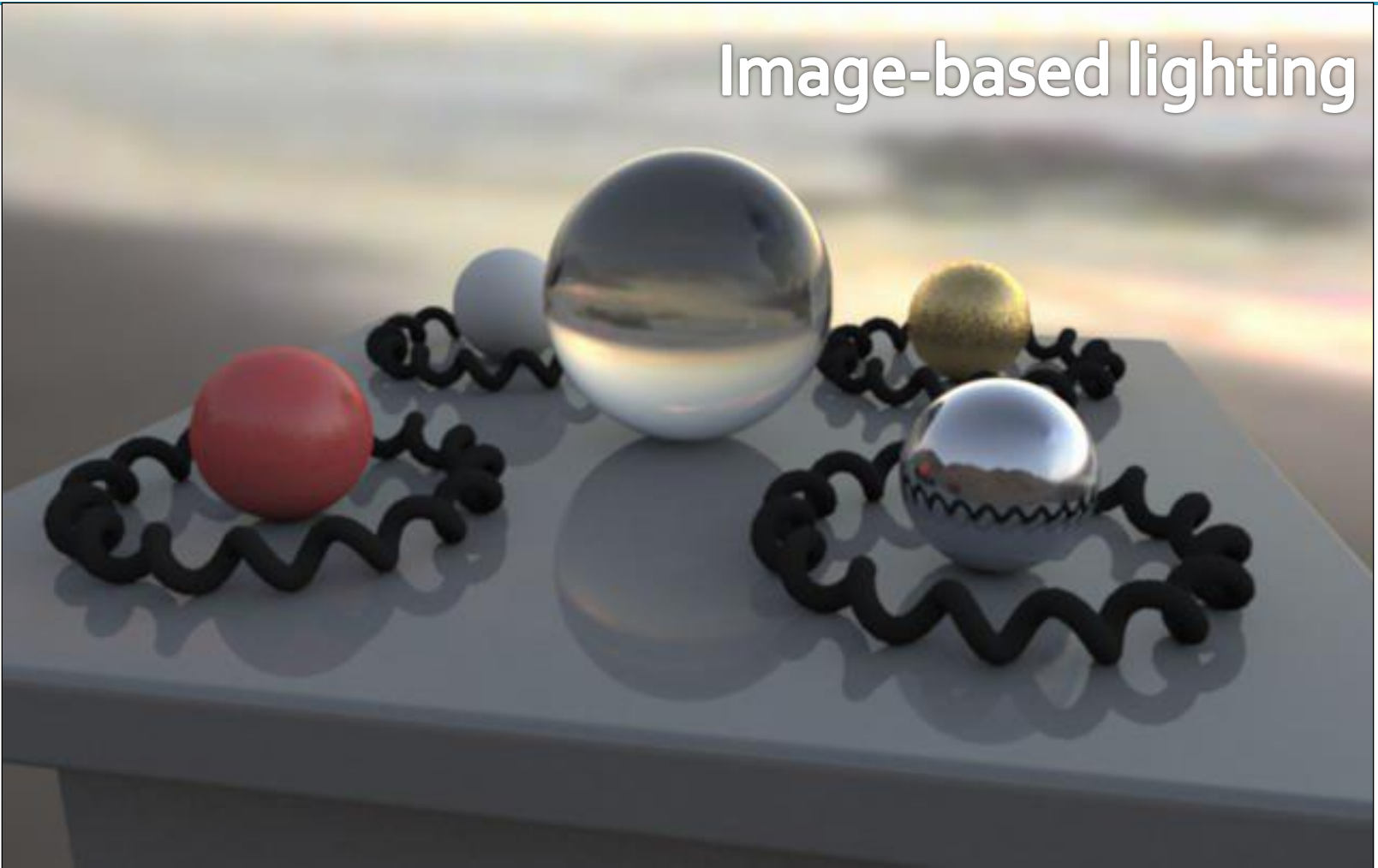


Image-based lighting

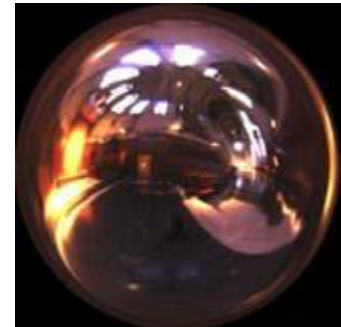
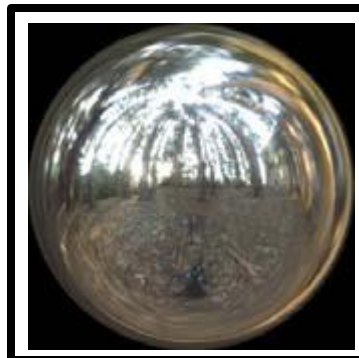


Image-based lighting

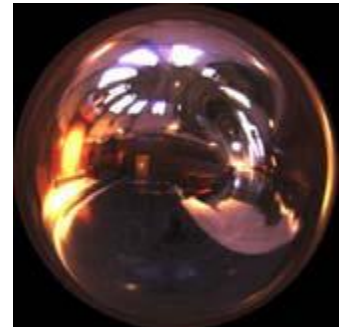
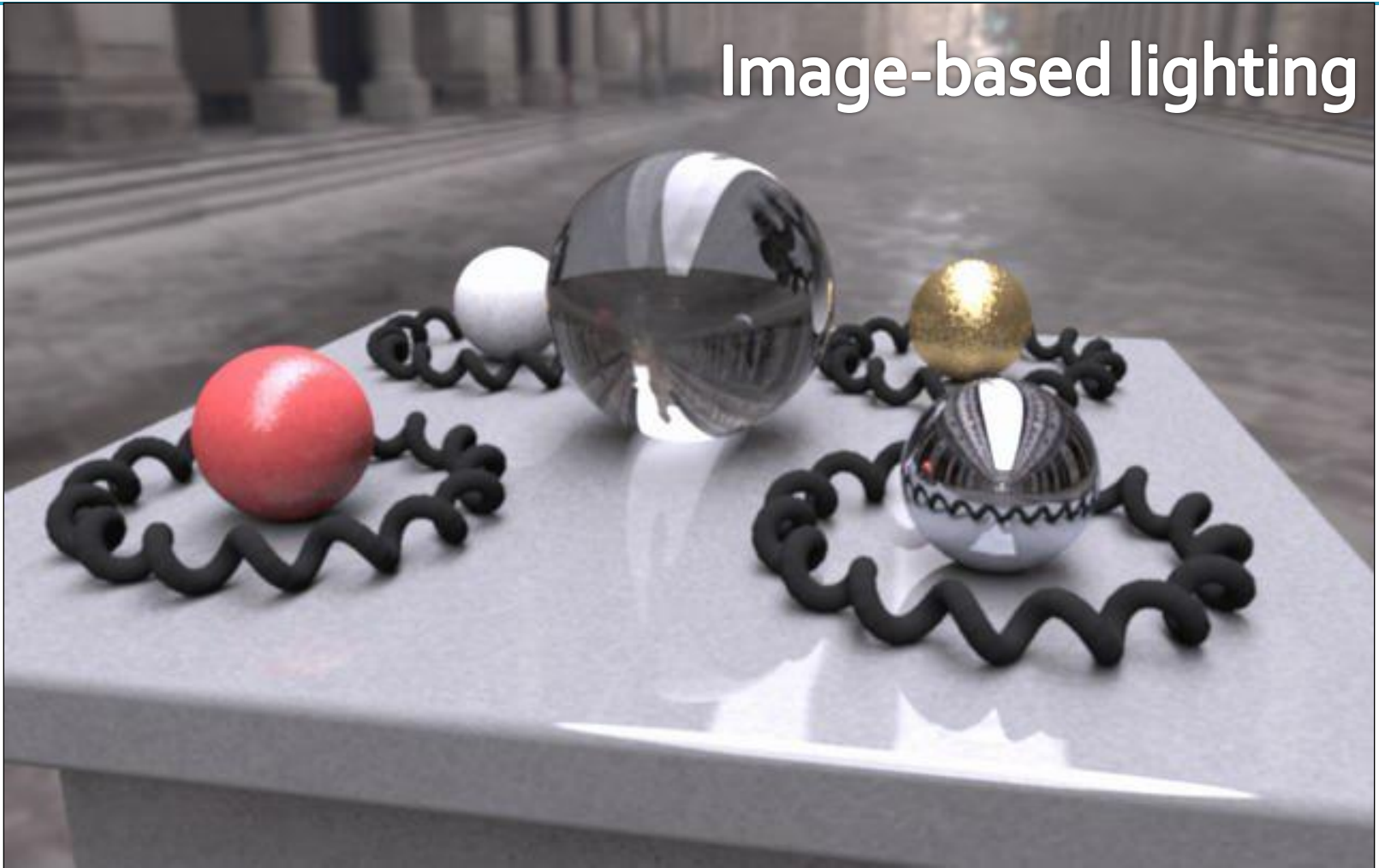
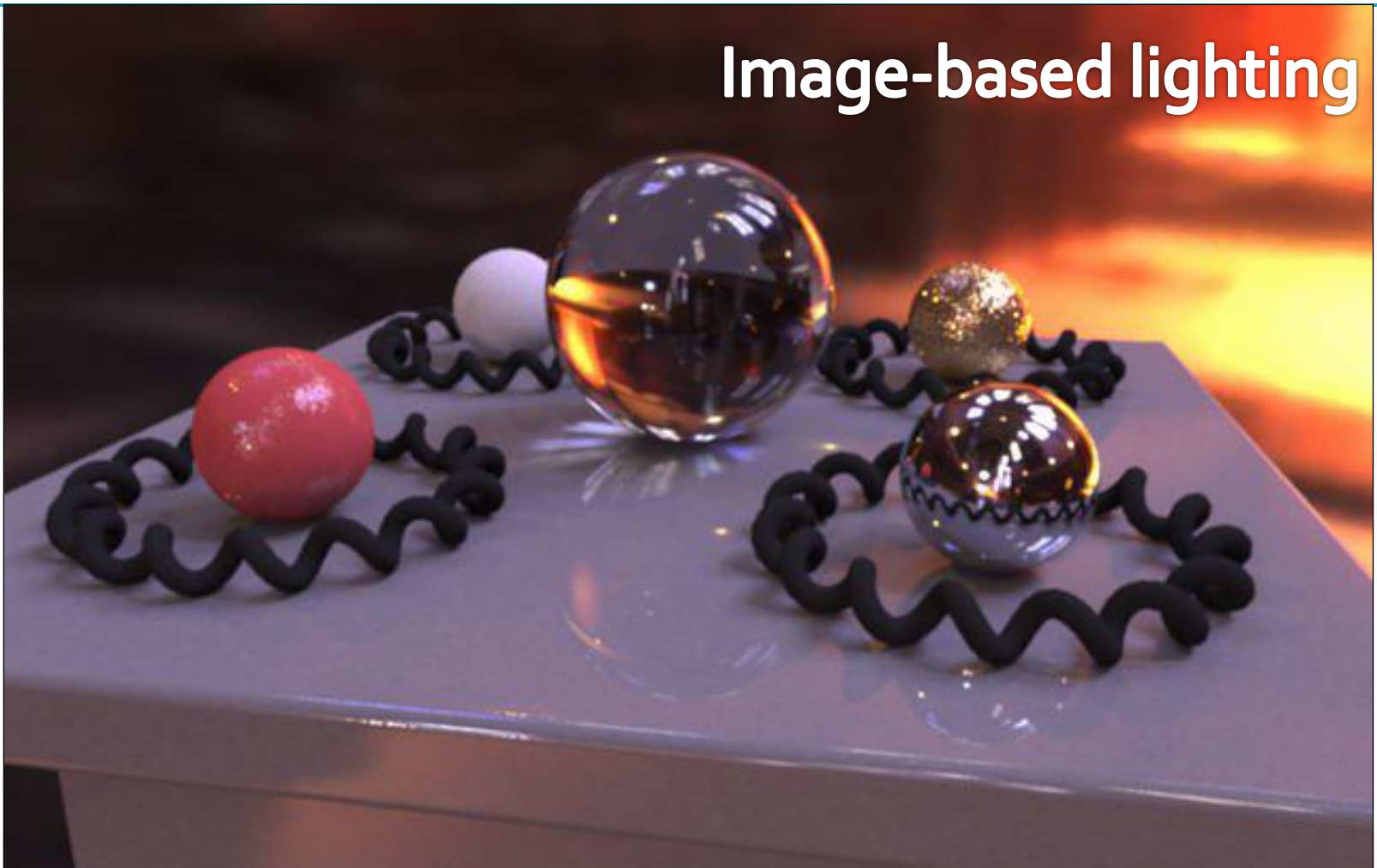


Image-based lighting

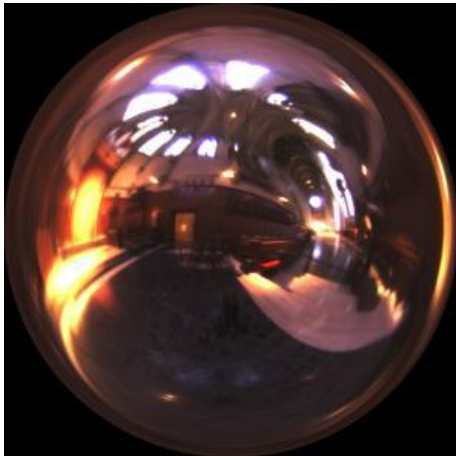


Mapping

Eucalyptus grove



Grace cathedral



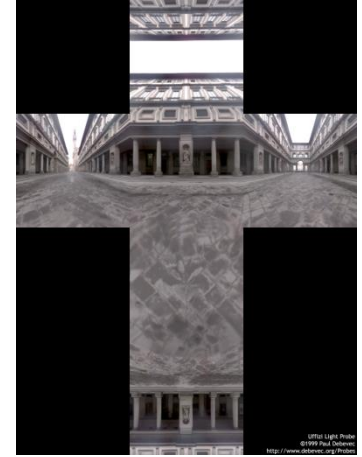
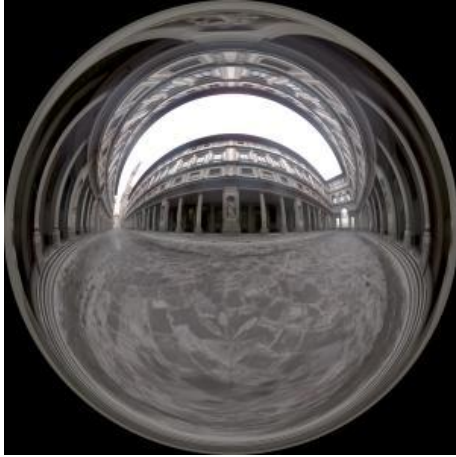
Debevec's spherical

"Latitude – longitude" (spherical coordinates)

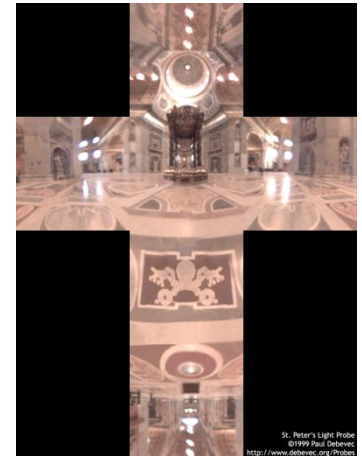
Cube map

Mapping

Uffizi gallery



St. Peter's Cathedral



Debevec's spherical

"Latitude – longitude" (spherical coordinates)

Cube map

Mapping

- Mapping from direction in Cartesian coordinates to image UV.

```
float d = sqrt(dir.x*dir.x + dir.y*dir.y);  
float r = d>0 ? 0.159154943*acos(dir.z)/d : 0.0;  
u = 0.5 + dir.x * r;  
v = 0.5 + dir.y * r;
```



Quote from "<http://ict.debevec.org/~debevec/Probes/>"

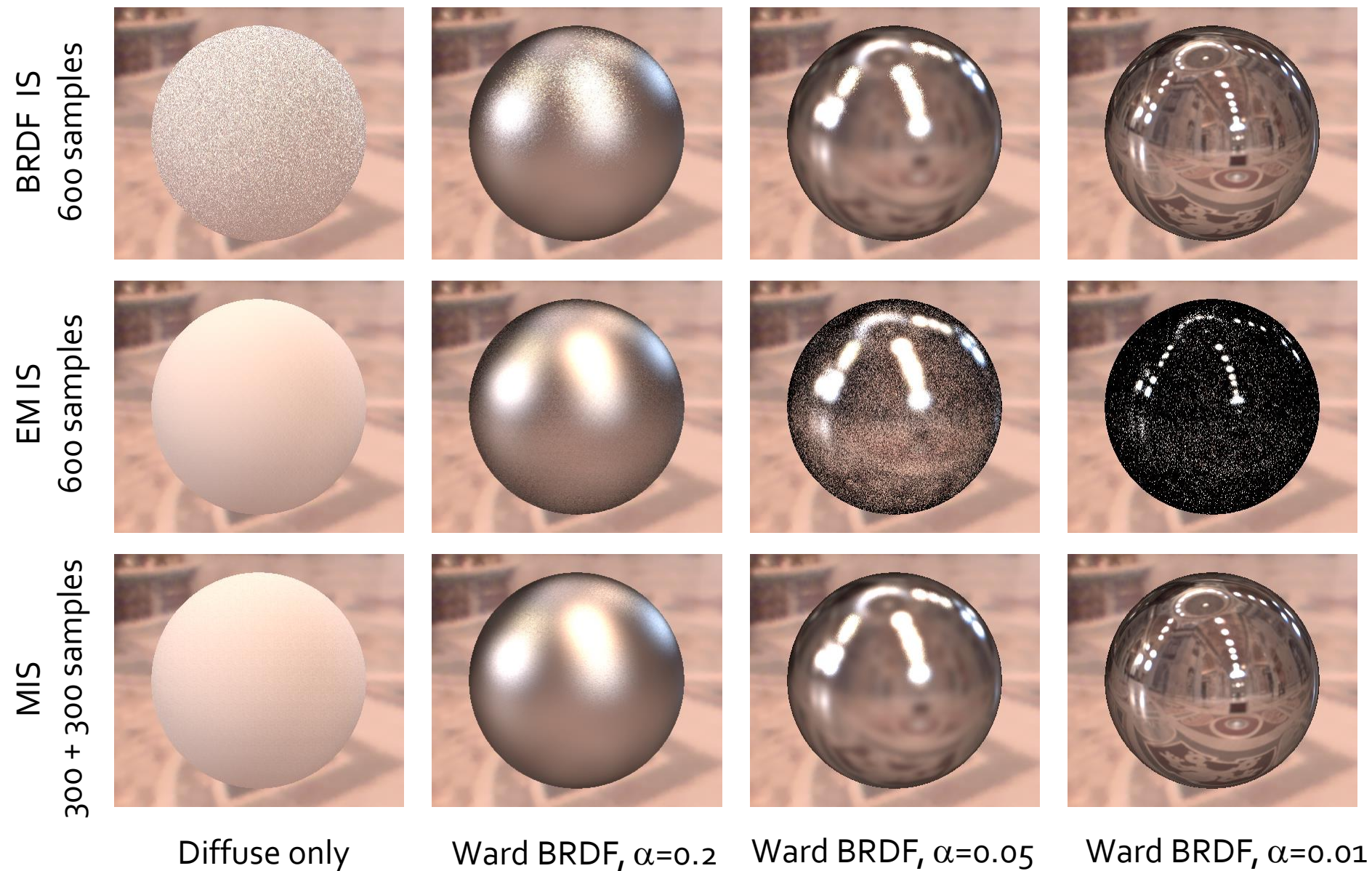
The following light probe images were created by taking two pictures of a mirrored ball at ninety degrees of separation and assembling the two radiance maps into this registered dataset. The coordinate mapping of these images is such that the center of the image is straight forward, the circumference of the image is straight backwards, and the horizontal line through the center linearly maps azimuthal angle to pixel coordinate.

Thus, if we consider the images to be normalized to have coordinates $\mathbf{u}=[-1,1]$, $\mathbf{v}=[-1,1]$, we have $\theta=\text{atan2}(\mathbf{v},\mathbf{u})$, $\phi=\pi\sqrt{\mathbf{u}*\mathbf{u}+\mathbf{v}*\mathbf{v}}$. The unit vector pointing in the corresponding direction is obtained by rotating $(0,0,-1)$ by ϕ degrees around the \mathbf{y} (up) axis and then θ degrees around the $-\mathbf{z}$ (forward) axis. If for a direction vector in the world (D_x, D_y, D_z) , the corresponding (\mathbf{u},\mathbf{v}) coordinate in the light probe image is (D_x*r, D_y*r) where $r=(1/\pi)*\text{acos}(D_z)/\sqrt{D_x^2 + D_y^2}$.*

Sampling strategies

- Technique (pdf) 1:
BRDF importance sampling
 - Generate directions with a pdf proportional to the BRDF
- Technique (pdf) 2:
Environment map importance sampling
 - Generate directions with a pdf proportional to $L(\omega)$ represented by the EM

Sampling strategies



Vzorkování směrů podle mapy prostředí

- Intenzita mapy prostředí definuje hustotu (pdf) na jednotkové kouli
- Pro účely vzorkování ji aproximujeme jako 2D diskrétní distribuci nad pixely mapy
- Pravděpodobnost výběru pixelu je dána součinem
 - Intenzity pixelu
 - Velikostí pixelu na jednotkové kouli (závisí na mapování)
- Detaily
 - Writeup
 - PBRT - <http://pbrt.org/plugins/infinitesample.pdf>