Noname manuscript No.
(will be inserted by the editor)

A Radiance Cache Method for Highly Glossy Surfaces

Mahmoud Omidvar · Mickaël Ribardière · Samuel Carré · Daniel Méneveaux · Kadi Bouatouch

Abstract Radiance caching methods have proven to be efficient for global illumination. Their goal is to compute precisely illumination values (incident radiance or irradiance) at a reasonable number of points lying on the scene surfaces. These points, called *records*, are stored in a cache used for estimating illumination at other points in the scene. Unfortunately, with records lying on glossy surfaces, the irradiance value alone is not sufficient to evaluate the reflected radiance; each record should also store the incident radiance for all incident directions. Memory storage can be reduced with projection techniques using spherical harmonics or other basis functions. These techniques provide good results for low shininess BRDFs. However, they get impractical for shininess of even moderate value since the number of projection coefficients increase drastically. In this paper, we propose a new radiance caching method that handles highly glossy surfaces while requiring a low memory storage. Each cache record stores a coarse representation of the incident illumination thanks to a new data structure, called *Equivalent Area light Sources* (*EAS*), capable of handling fuzzy mirror surfaces. In addition, our method proposes a new simplification of the interpolation process since it avoids the need for expressing and evaluating complex gradients.

Keywords Rendering · Global Illumination · Radiance caching · Glossy surfaces

Mahmoud Omidvar
CSTB Nantes, Université de Rennes 1, France
E-mail: mahmoud.omidvar@univ-poitiers.fr

Mickaël Ribardière
XLIM-SIC, Université de Poitiers, France
E-mail: mickael.ribardiere@univ-poitiers.fr

Samuel Carré
CSTB Nantes, France
E-mail: samuel.carre@cstb.fr

Daniel Méneveaux
XLIM-SIC, Université de Poitiers, France
E-mail: daniel.meneveaux@univ-poitiers.fr

Kadi Bouatouch
IRISA, Université de Rennes 1, France
E-mail: kadi.bouatouch@irisa.fr

1 Introduction

Producing physically based images requires taking into account multiple light interactions in a 3D environment made up of complex objects and complex materials. This process, often called *global illumination*, is mathematically modeled with the well known rendering equation [7]. It can be performed according to a significant number of techniques [11, 6, 20], improved through years (for an overview of the recent advances in light transport simulation refer to [10]). However, they all produce images with noticeable noise that can be lowered with additional samples (rays, ray paths, photons, etc.) which unfortunately drastically increase the computation time.

One solution for reducing noise while keeping reasonable the computing time is to resort to caching techniques. These latter estimate illumination (incident radiance or irradiance) for few points on the scene surfaces, store the values in a cache (set of records), and use the cache for rendering other points of the scene. This idea was first proposed by Ward *et al.* [24, 23] for the global diffuse component (irradiance) while the indirect glossy and specular components are computed separately, using distributed ray tracing. This method, called *irradiance caching*, consists of two passes. The first one initializes the set of records at which the irradiance is computed, depending on the camera position. The second pass corresponds to the rendering process; cached irradiance values are interpolated in order to estimate irradiance at other points seen from the camera. For records lying on glossy surfaces, the irradiance value is not sufficient to

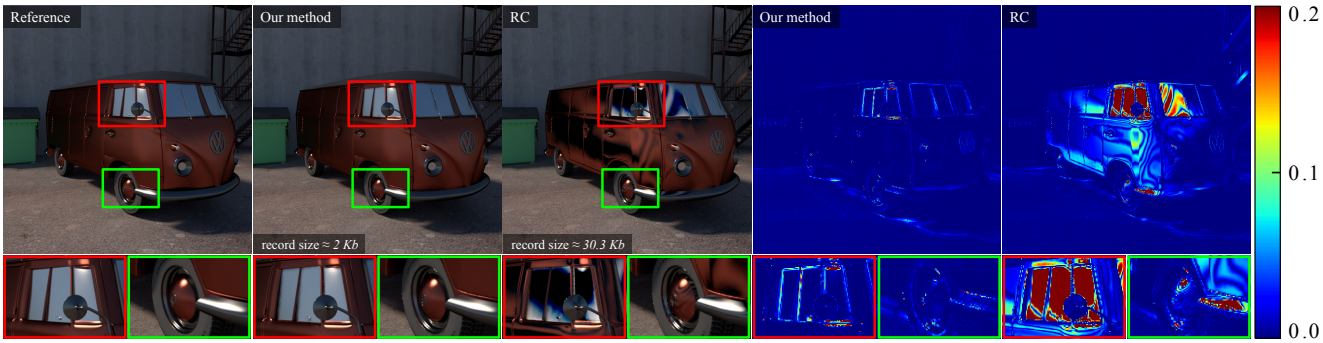


Fig. 1: *Camper van* scene lighted with an environment map: our method renders efficiently sharp glossy materials with less visual artifacts. Each record stores indirect and direct lighting from the environment map. The bodywork and the chrome of the camper van are made up of a material defined by a modified Phong model with a shininess $n = 100$ while the glasses are assigned a shininess $n = 200$. These highly glossy materials cannot be handled precisely by the Radiance Caching (RC) method [9,8].

reconstruct the radiance reflected toward the camera, each record should be associated with the incident radiance for all incident directions. Křivánek *et al.* [9] propose to cache radiance instead of irradiance and store the directional information using a basis function space called hemispherical harmonics [3]. Irradiance and radiance caching methods have been improved in different manners over the last decade [8,2,4,15,16]. However, none of them can handle highly glossy surfaces. Xu et al. [25] have proposed a solution with single bounce inter-reflections. Another caching method, based on Light Vectors, has been proposed [26,18]. In this approach, all the incident directions at a point are replaced by a mean radiance assigned to a single incident direction. Hence, the visible surfaces (to the camera) are supposed to be perfectly diffuse. Unlike the above mentioned methods, our approach is capable of handling a wide range of BRDFs starting from pure diffuse to highly glossy ones in static scenes, as well as multiple reflection bounces.

With cache-based methods, rendering is faster than with brute force methods because costly illumination computations are performed only for a subset of image pixels. However *radiance caching* remains limited to low frequency materials. For instance, with a modified Phong model [13] exponent (shininess) exceeding 50, a huge number of spherical harmonics coefficients are required, hence increasing the computation time as well as the memory cost, and inevitably prone to *ringing artifacts* [9]. An interactive rendering system, based on radiance caching, is also proposed by Scherzer et al. [16]. It handles highly specular materials with complex scenes. The main idea is to replace spherical harmonics projection and evaluation by mip-mapping operations (directly implemented on the graphics processor unit), with an additional memory cost due to mip-maps storage. However, the additional memory makes it impossible to apply this method to scenes that require a high number of records. Moreover, it handles only Phong BRDF models.

This paper proposes a global illumination caching method, based on the *radiance caching* scheme. It is capable of handling shiny objects defined by sharp reflection lobes, while requiring low memory storage. Therefore, each record is associated with a representation of the incident illumination thanks to a new data structure called *Equivalent Area light Sources* (EAS). An EAS corresponds to a group of incident radiances, determined according to geometrical and physical metrics. We show that our approach does not need costly and complex gradients to determine illumination variations around a record position. The main contributions and strengths of our method are:

- an efficient radiance caching approach that handles scenes containing sharp glossy surfaces with a low memory storage;
- a new directional representation of incident radiances called equivalent area light sources;
- a new representation of BRDF based on generalized cosine lobes for handling various incident solid angles;
- a method adapted to existing *irradiance/radiance caching* algorithms with EAS computation and radiance interpolation.

The rest of this paper is organized as follows. Section 2 provides an overview of our approach. Section 3 describes in detail our method. Results are discussed in section 4 before concluding and presenting future work in section 5.

2 Overview

The *irradiance/radiance caching* algorithm exploits spatial coherence using sparse sampling and interpolation of irradiance or incident radiance. Each computed value is saved in a data structure called *record*, and each record R is represented by its position x_R, its normal $\overrightarrow{N_R}$ and a radius r defining its region of influence. The illumination of a given point

x, lying within the influence region of a set of records, is determined by a weighted interpolation of radiance/irradiance stored at these records. The algorithm consists of two steps: the cache filling step which creates all the necessary records and the rendering step. Our approach follows the same principles and is depicted by the global algorithm given by Algorithm 1.

Algorithm 1 Radiance caching algorithm

```
1:  // Cache filling stage
2:  for each pixel j do
3:      x_R ← Shoot-View-Rays(j); // x_R: record position
4:      C ← Gather-Existing-Records(x_R);
5:      if C = {} then
6:          // No record surrounds x_R, create a new record
7:          S ← Stratified-Sampling-Ω(x_R);
8:          data = {};
9:          for each ω_i in S do
10:             L ← Compute-Radiance(ω_i);
11:             data ← data ∪ Store-Contribution(ω_i, L);
12:         end for
13:         EAS ← Determine-EAS(data);
14:         R ← Create-Record(x_R, EAS);
15:         Store-In-Cache(R);
16:     end if
17: end for
18: // Rendering stage
19: for each pixel j do
20:     x_P ← Shoot-View-Rays(j);
21:     C ← Gather-Existing-Records(x_P);
22:     // Interpolate records values computed from EAS
23:     Compute-Pixel-Radiance(C, j);
24: end for
```

The records are generated within a scene from a given viewpoint, using sparse sampling of pixels (Algorithm 1, Line 3).

Let x_R be the position of a new record. The reflected radiance L_r at x_R and in any direction $\overrightarrow{\omega}$ is given by:

$$L_r(x_R, \overrightarrow{\omega}) = \int_{\Omega} L(x_R, \overrightarrow{\omega_i})\ f_r(x_R, \overrightarrow{\omega_i}, \overrightarrow{\omega})\ (\overrightarrow{\omega_i} \cdot \overrightarrow{N_R})\ d\overrightarrow{\omega_i}, \quad (1)$$

where Ω is the hemisphere centered around the position x_R, L is the incoming radiance from the direction $\overrightarrow{\omega_i}$, $\overrightarrow{N_R}$ the normal at the position x_R and f_r the *Bidirectional Reflectance Distribution Function* BRDF (Algorithm 1, Line 10). The integral domain Ω is sampled using a stratified sampling technique (Algorithm 1, Line 7) as in the original irradiance or radiance caching methods. L can be evaluated using any global illumination method such as photon density estimation, path tracing, etc. Equation 1 is notably costly to evaluate, and for caching methods it would be ideal to store all the incident radiances in the record data structure. This is not possible in practice since it would require a high memory storage. To tackle this problem, previous techniques project directional information into a spherical harmonic basis function space [9]. However, only low frequency BRDFs can be considered when using this strategy since highly glossy lobes require a high number of coefficients; this problem arises regardless of the basis functions. Rather, our method does not make use of any basis function, it partitions the set of incident rays into clusters representing the incident radiance; generalized cosine lobes are used as basis functions only for our BRDF representation.

Let $\{x_i\}$ be the set of points corresponding to the intersection between the scene and each sampled incident direction $\overrightarrow{\omega_i}$ (see Figure 2). These points are clustered depending on physical and geometrical criteria as detailed in section 3.2, and each cluster is represented by a surface that illuminates the record at position x_R (Algorithm 1, Line 13 and Figure 2b). This light surface will be called *Equivalent Area Light Source EAS* from now on. A set of EAS is associated with each record (Algorithm 1, Line 14); all the records are stored in an *octree* which represents the global cache.

During the rendering process, the image plane is sampled, providing camera rays. Given an intersection point P between a camera ray and the scene (Algorithm 1, Line 20), the records whose influence zone covers the intersection point P are gathered (line 21) and their contributions are interpolated (line 23), providing the required radiance estimation at P. A record contribution is determined based on its EAS, as explained in Section 3.3.

3 Our approach: details and methodology

Starting from Equation 1, the key idea of our method is to subdivide the hemisphere Ω, which is a solid angle of $2\pi\ sr$, into a set of $\#SA$ solid angles Ω_j such that $\Omega = \sum_{j=1}^{\#SA} \Omega_j$. Such a representation corresponds to the following discretized rendering equation:

$$L_r(x_R, \overrightarrow{\omega}) = \sum_{j=1}^{\#SA} \int_{\Omega_j} L_j(x_R, \overrightarrow{\omega_i})\ f_r(x_R, \overrightarrow{\omega_i}, \overrightarrow{\omega})\ (\overrightarrow{\omega_i} \cdot \overrightarrow{N_R})\ d\overrightarrow{\omega_i}. \quad (2)$$

If the corresponding subdivision is such that the incident radiance L_j within a solid angle Ω_j is approximately constant $\left(\widetilde{L_j}(x_R) \approx L_j(x_R, \overrightarrow{\omega_i}) \forall\ \overrightarrow{\omega_i} \in \Omega_j\right)$, Equation 2 becomes:

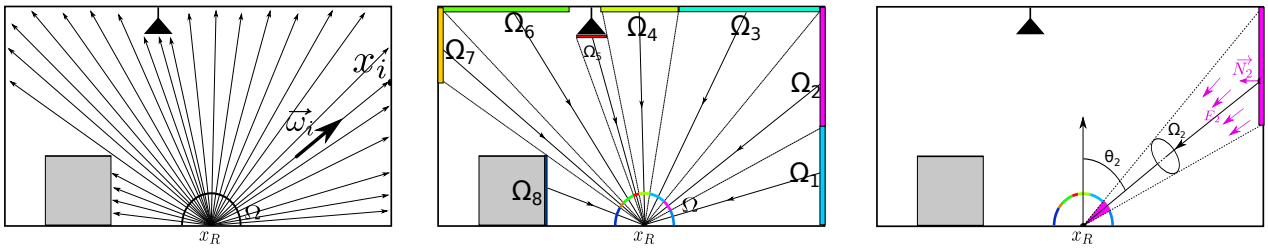
$$L_r(x_R, \overrightarrow{\omega}) \approx \sum_{j=1}^{\#SA} \widetilde{L_j}(x_R) \int_{\Omega_j} f_r(x_R, \overrightarrow{\omega_i}, \overrightarrow{\omega})\ (\overrightarrow{\omega_i} \cdot \overrightarrow{N_R})\ d\overrightarrow{\omega_i}. \quad (3)$$

We also define $L_{r_j}(x_R, \overrightarrow{\omega})$, the radiance reflected from all the contributions within Ω_j:

$$L_{r_j}(x_R, \overrightarrow{\omega}) \approx \widetilde{L_j}(x_R) \int_{\Omega_j} f_r(x_R, \overrightarrow{\omega_i}, \overrightarrow{\omega})\ (\overrightarrow{\omega_i} \cdot \overrightarrow{N_R})\ d\overrightarrow{\omega_i}. \quad (4)$$

With the same assumption (constant incident radiance within a given solid angle Ω_j), the fraction E_j of irradiance in Ω_j is given by:

$$\begin{aligned} E_j(x_R) &= \int_{\Omega_j} L_j(x_R, \overrightarrow{\omega_i})(\overrightarrow{\omega_i} \cdot \overrightarrow{N_R}) d\overrightarrow{\omega_i} \\ &\approx \widetilde{L_j}(x_R) \int_{\Omega_j} (\overrightarrow{\omega_i} \cdot \overrightarrow{N_R}) d\overrightarrow{\omega_i}. \end{aligned} \quad (5)$$



(a) Hemisphere sampling (b) Clustering (c) EAS

Fig. 2: a) Sampling of a hemisphere placed above the record position x_R, each intersection point x_i is associated with an elementary solid angle; b) the samples are clustered depending on their contribution and c) an EAS is defined by its solid angle Ω_2, its mean irradiance E_2 reaching x_R and its elevation angle given by θ_2.

The incoming radiance, constant within a solid angle Ω_j, is directly derived from Equation 5:

$$\widetilde{L_j}(x_R) \approx \frac{E_j(x_R)}{\int_{\Omega_j}(\overrightarrow{\omega_i}\cdot\overrightarrow{N_R})d\overrightarrow{\omega_i}}. \quad (6)$$

Using Equations 6 and 4:

$$L_{r_j}(x_R,\overrightarrow{\omega}) \approx E_j(x_R) \times \left(\frac{\int_{\Omega_j} f_r(x_R,\overrightarrow{\omega_i},\overrightarrow{\omega})\ (\overrightarrow{\omega_i}\cdot\overrightarrow{N_R})\ d\overrightarrow{\omega_i}}{\int_{\Omega_j}(\overrightarrow{\omega_i}\cdot\overrightarrow{N_R})d\overrightarrow{\omega_i}}\right). \quad (7)$$

In Equation 7, the second term of the right part corresponds to the mean BRDF evaluated for Ω_j. It is denoted by $\widetilde{F_r}$ in the following and its computation is detailed in section 3.1.

Each partial irradiance E_j together with its associated solid angle Ω_j represent an EAS. In practice, it corresponds to a cluster of intersection points between rays shot from a record position to collect the radiance (L_j in Equation 1) and the scene. Each intersection point x_i is associated with an elementary solid angle; the solid angle Ω_j assigned to an EAS is equal to the sum of these elementary solid angles (see Figure 2c). The *position* of an EAS is defined as the centroid of these points.

Each record stores its associated EASs and each EAS is a data structure containing the elevation angle θ_j (direction computed using the record position and the EAS position), the solid angle Ω_j and the mean irradiance E_j. Section 3.2 describes how the EASs are constructed.

Finally, using Equation 7 and Equation 3, the reflected radiance can be expressed as the sum of reflected radiances due to all the EASs:

$$L_r(x_R,\overrightarrow{\omega}) \approx \sum_{j=1}^{\#EAS} L_{r_j}(x_R,\overrightarrow{\omega}), \quad (8)$$

where $\#EAS$ is the number of EAS. The EASs are computed as described in the following section.

3.1 Mean BRDF precomputation

In the general case, a solid angle Ω_j of incident radiance is represented by a position on the hemisphere (corresponding to an elevation angle θ_e, and an azimuthal angle ϕ_a), as well as an extent equal to $(\Delta\theta,\Delta\phi)$ which is the range of θ and ϕ around the position of the solid angle. For the sake of simplicity, we describe the method with isotropic materials, but the theory holds for anisotropic materials. In this case, the azimuth value ϕ_a does not need to be accounted for. In Section 4 we provide results obtained with an anisotropic BRDF Ward model [22] and with a Micro-facet based BRDF model.

Following Equation 7, $\widetilde{F_r}$ is defined as the *mean BRDF*, depending on the solid angle Ω_j and the elevation θ_e:

$$\widetilde{F_r}(\theta_e,\Omega_j,\overrightarrow{\omega}) = \frac{\int_{\Omega_j} f_r(x_R,\overrightarrow{\omega_i},\overrightarrow{\omega})\ (\overrightarrow{\omega_i}\cdot\overrightarrow{N_R})\ d\overrightarrow{\omega_i}}{\int_{\Omega_j}(\overrightarrow{\omega_i}\cdot\overrightarrow{N_R})d\overrightarrow{\omega_i}}, \quad (9)$$

where $\overrightarrow{\omega}$ is the reflected direction. Note that Equation 9 does not depend on the scene illumination and for anisotropic materials, the azimuth ϕ_a should be used as a fourth parameter.

Projection on cosine lobe functions

We propose to precompute $\widetilde{F_r}$ for a set of parameter triplets (the incident elevation θ_e, the incident solid angle Ω_j and an outgoing direction $\overrightarrow{\omega}$). The obtained values are projected into a basis function space and the projection coefficients are stored in a tabulated file that describes the material behavior. Each material of a scene must be described by such a file.

The contribution of a given solid angle Ω_j covers several reflected directions $\overrightarrow{\omega}$ (as shown in Figure 3). We propose to estimate values of Equation 9 and represent the contributions using general cosine lobes as basis functions [12, 14]. This choice has been guided by the simplicity and effectiveness of this representation for sharp reflections: one lobe is controlled by only four coefficients C_x, C_y, C_z, and a shininess parameter n. $\widetilde{F_r}$ can thus be approximated by the following sum:

$$\widetilde{F_r}(\theta_e,\Omega_j,\overrightarrow{\omega}) \approx \sum_{k=1}^{\#lobes} \left(C_x^k x_\omega x_{\omega_j} + C_y^k y_\omega y_{\omega_j} + C_z^k z_\omega z_{\omega_j}\right)^{n^k} \quad (10)$$

where x_ω, y_ω and z_ω represent vector coordinates. For a given elevation angle θ_e and a given solid angle Ω_j (see Figure 3), all the contributing values of the mean BRDF $\widetilde{F}_r$ are used to determine the generalized lobe parameters C_x^k, C_y^k, C_z^k and n^k. This fitting process is depicted in Algorithm 2 and is performed using a genetic algorithm [1]. As fitting is not the main topic of the paper, we do not provide any details about our implementation. Note that other fitting techniques could be used.

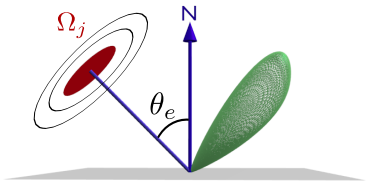


Fig. 3: $\widetilde{F}_r$ representation for a given solid angle Ω_j (in red) with an elevation angle θ_e contributes to different reflected directions and forms a reflection lobe (in green).

Algorithm 2 Mean BRDF fitting

```
1:  for each elevation angle θ_e do
2:      for each solid angle Ω_j(ΔΘ, ΔΦ) do
3:          for each reflected direction ω⃗ do
4:              F̃_r(ω⃗) ← Compute-Mean-BRDF(Ω_j, θ_e, ω⃗); // eq. 9
5:          end for
6:          // Genetic Algorithm
7:          {C_x^k, C_y^k, C_z^k, n^k} ← Fitting(all the F̃_r values);
8:          Store {C_x^k, C_y^k, C_z^k, n^k} in tabulated file Tab;
9:      end for
10: end for
```

The result is a tabulated file *Tab* (Algorithm 2 line 8), indexed with the solid angle and the elevation angle, where each element $Tab[\theta_e][\Omega_j]$ represents the projection coefficients of a mean BRDF for an elevation angle θ_e and a solid angle Ω_j such that:

$$\theta_e = e \times \frac{\pi}{2 \times \#H}, \quad e \in]0..\#H]$$
$$\Omega_j = j \times \frac{2\pi(1-\sin(\theta_i))}{\#AS}, \quad j \in]0..\#AS],$$

where $\#H$ and $\#AS$ are the number of elevation angles and solid angles respectively.

In case of diffuse materials, the fitting process should not be performed because Equation 9 is reduced to a diffuse reflectance term which is the same for all the reflected directions whatever the EAS elevation and solid angles.

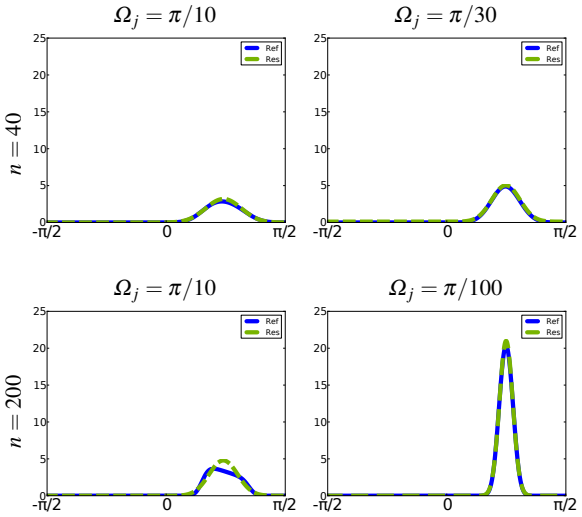


Fig. 4: Fitting results for 2 modified Phong lobes with shininess $n = 40$ (first row) and $n = 200$ (second row) for different solid angle configurations. The plots represent $\widetilde{F}_r$ for an elevation angle $\theta_e = \pi/4$ of the solid angle direction (green plot for our fitting process and blue plot for the reference).

Approximation quality

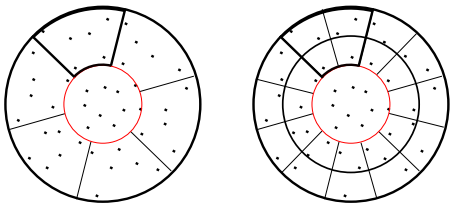
We have applied our mean BRDF representation to a Phong BRDF model with various shininess coefficients n. Figure 4 presents a comparison between our approximation of $\widetilde{F}_r$ for an elevation $\theta_e = \pi/4$ and a reference corresponding to unbiased monte-carlo integration.

This representation leads to a mean error of $\widetilde{F}_r$ always lower than 1% for the images produced in this paper, essentially depending on the solid angle Ω_j (Figure 4). We have illustrated several configurations that produce various errors. With a shininess exponent $n = 40$ and $\Omega_j = \pi/10$, the mean error is equal to 0.56%, while with $\Omega_j = \pi/30$ the mean error is equal to 0.07%. We have also considered a case giving an error of 2.48% with $n = 200$ and $\Omega_j = \pi/10$, but the error can be highly reduced (down to 0.16%) with a reasonable solid angle $\Omega_j = \pi/100$.

3.2 Record construction and EAS

For each record, the corresponding hemisphere of incoming radiance directions is sampled; each sample is described by an incident direction $\overrightarrow{\omega_i}$ (which corresponds to a point on the unit hemisphere) and an incident radiance $L_j(x_R, \overrightarrow{\omega_i})$.

The incident directions (points on the unit hemisphere) are orthogonally projected on the disk hemisphere base. These projected points are recursively clustered to build a tree data structure as shown in Figure 5. A node is either a spherical cap (represented by a disk on the hemisphere base) or a sector. A spherical cap can be recursively subdivided into six



(a) (b)

Fig. 5: Projected representation of the hemisphere and recursive subdivisions. Each point corresponds to a sampled direction. a) first subdivision level with 1 spherical cap in red and 6 sectors; b) each sector is subdivided into 4 new sectors.

sectors and a smaller spherical cap located at its center (Figure 5a), while a sector can be subdivided into four smaller sectors (Figure 5b). Note that any other subdivision scheme could be used but the spherical cap is elegantly handled with our method. From now on, a *sample* corresponds to either an incident direction or its projection on the hemisphere base.

Each leaf of the tree is a sample that represents a potential EAS associated with an elevation θ_i (the elevation angle of the sample), an elementary solid angle $\Omega_i = 2\pi/\#s$, $\#s$ being the number of samples, and an irradiance given by:

$$E_i(x_R) \approx L(x_r, \overrightarrow{\omega_i}) \frac{2\pi}{\#s} (\overrightarrow{\omega_i} \cdot \overrightarrow{N_R}). \tag{11}$$

The tree is traversed bottom-up to assign each internal node N (representing a potential EAS) a solid angle Ω_N, an irradiance E_N, a position Pos_N (used to compute its elevation θ_N) and a normal $\overrightarrow{N_N}$. These data are computed from the children SN of N as follows:

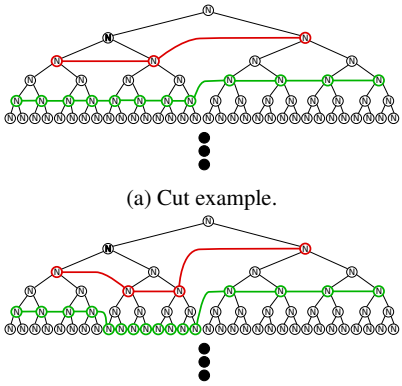
$$\begin{aligned}
E_N(x_R) &= \sum_{i=1}^{\#SN} E_{SN_i}(x_R) \\
\Omega_N &= \sum_{i=1}^{\#SN} \Omega_{SN_i} \\
Pos_N &= \frac{\sum_{i=1}^{\#SN} Pos_{SN_i} \times E_{SN_i}(x_R)}{\sum_{i=1}^{\#SN} E_{SN_i}(x_R)} \\
\overrightarrow{N_N} &= \frac{\sum_{i=1}^{\#SN} \overrightarrow{N_N} \times E_{SN_i}(x_R)}{\sum_{i=1}^{\#SN} E_{SN_i}(x_R)}
\end{aligned} \tag{12}$$

where $\#SN$ is the number of children of node N. In the leaves, each sample (given by an incident direction $\overrightarrow{\omega_i}$) is associated with the normal $\overrightarrow{N_i}$ of the corresponding intersection point x_i. The mean normal $\overrightarrow{N_N}$, assigned to each node N, allows to account for the EAS irradiance variations within the influence zone of the corresponding record as explained in section 3.3.

EAS clustering

The hierarchical representation described above corresponds to a tree built for each record R. On one hand, the set of leaves (smallest potential EASs) are the most accurate estimated incident illumination at the record position x_R; however, it is impossible to store all these small EASs for all the records due to the high memory cost needed. On the other hand, the root of this tree (only one EAS) is a very coarse representation of the incident illumination (in fact it is the total irradiance at the record position). This is why we propose to determine the nodes of the tree which could represent at best the incident illumination. These nodes define the EASs that are actually stored in the record data structure. They are obtained according to cuts in the tree similarly to [21] and with the lowest possible memory cost. In [4], the authors propose to cache incident radiance using lightcuts with a combination of spherical harmonics basis functions and VPLs. Again, the use of spherical harmonics is not adapted to very shiny objects that can be handled with our method. In addition, although our approach has similar aspects, it does not require to project incoming radiance onto basis functions and our tree representation is associated with each record, providing a more accurate incident radiance approximation.

Cuts in the tree are found according to an iterative process, starting from the root and going down the tree until meeting a given global precision criterion τ, defined by the user (all the images in this paper have been produced with $\tau = 0.02$). The initial cut is composed of the first level nodes in the tree since the root node always defines a too coarse directional representation of incident radiance. Ideally, the reflected radiance representation using these nodes as EASs, should be compared to the reflected one due to leaves in order to validate the precision of the current cut. For the sake of efficiency, we have found experimentally that a comparison of a cut with 2 levels below (called *comparison level*) propose a good trade-off (Figure 6a). The cut process consists in computing a *global* error ε_g between the mean reflected radiance at the record position due to the EASs of the current level and the one due to the EASs of the comparison level. If $\varepsilon_g > \tau$ the current cut is not accurate enough and a deeper cut has to be found. Therefore, the node of the current cut leading to the less accurate reflected radiance representation is replaced by its children in the cut (Figure 6b). It is identified by the highest *local* error ε (associated with each node of the current cut) which is evaluated by comparing the accuracy between each node and its descendants two levels below.



(a) Cut example.

(b) Next subdivision...

Fig. 6: A six level fictitious EAS tree; a) given a cut in the tree (in red), a global error ε_g determines the representation accuracy compared with two levels below (in green). If $\varepsilon_g > \tau$, τ being a user defined criterion, a local error ε is estimated for each node of the current cut; b) the node associated with the higher ε value is replaced by its children in the cut.

More precisely, let $\bar{\varepsilon}_g$ be the relative *Root Mean Square Error* which is an *RGB* vector:

$$\bar{\varepsilon}_g = \frac{\sqrt{\frac{\sum_i^{\#D}\left(L_r^{comp}(x_R,\overrightarrow{\omega_{r_i}}) - L_r^{cur}(x_R,\overrightarrow{\omega_{r_i}})\right)^2}{\#D}}}{\overline{L_r^{comp}}}, \tag{13}$$

where L_r^{cur} and L_r^{comp} are radiances reflected in $\#D$ directions. They are computed according to Equation 8, using $\widetilde{F_r}$ and the EASs of the current and the comparison levels respectively. $\#D$ is a user defined parameter representing the number of considered reflected directions. $\overline{L_r^{comp}}$ is the mean reflected radiance due to the EASs of the comparison level (also over the $\#D$ reflection directions $\overrightarrow{\omega_{r_i}}$).

In our implementation $\bar{\varepsilon}_g$ is transformed into the *XYZ* color space and $\varepsilon_g = Y$.

Given a node N of the current level that needs to be refined, we estimate for each color component R,G,B, an error ε between this node and all its children located 2 levels below in the tree. $\bar{\varepsilon}$ is expressed as:

$$\bar{\varepsilon} = \sum_i^{\#D}\left(L_r^{Child}(x_R,\overrightarrow{\omega_{r_i}}) - L_r^{N}(x_R,\overrightarrow{\omega_{r_i}})\right)^2 \tag{14}$$

where L_r^N (computed with Equation 7) and L_r^{Child} (computed using Equation 8) are the reflected radiance due to the EAS of node N and to those of its children respectively. Likewise, ε is the Y component of $\bar{\varepsilon}$.

Quality of radiance approximation

The combined use of EASs and mean BRDF $\widetilde{F_r}$ provides a new representation of the radiance L_r reflected by glossy surfaces. Given a record located at x_R, with a material approximated by $\widetilde{F_r}$, the estimation of L_r is given by:

$$L_r(x_R,\overrightarrow{\omega}) \approx \sum_{i=1}^{\#EAS} E_i(x_R) \times \widetilde{F_r}(\Omega_i,\theta_i,\overrightarrow{\omega}), \tag{15}$$

where E_i corresponds to the irradiance associated with EAS i, $\#EAS$ being the number of EAS of the record.

Since $\widetilde{F_r}$ is represented for fixed solid angles and elevations, intermediate values can be obtained according to a linear interpolation of cosine lobe coefficients, from the closest solid angles and elevation parameters.

As an illustration, Figure 7 provides some results corresponding to a Cornell Box scene containing a glossy ground, represented by a modified Phong model, with several shininess values n. The images represent the radiance reflected at a record location x_R on this ground, each pixel corresponds to the radiance for a reflection direction $\overrightarrow{\omega}$ (θ,ϕ). With a sharp lobe ($n = 100$), only 100 EAS are required to approximate the reflected radiance. The plot in Figure 7 illustrates the influence of the number of EAS on the reflected radiance quality (measured using an RMSE metrics).

Figure 8 compares the reflected radiance obtained with our method to the one proposed by Křivánek *et al.* [9] using hemispherical harmonics. We use the same scene configuration as in Figure 7 and for both methods the same number of incident rays per record. For a fixed image quality (*RMSE*), we also compare memory storage depending on the Phong shininess exponent n. Except for low-frequency BRDFs ($n = 5$), for which memory storage is roughly equivalent, our method requires much less memory. Moreover, the error distribution is smoother with EAS than with spherical harmonics that produce *ringing effects*.

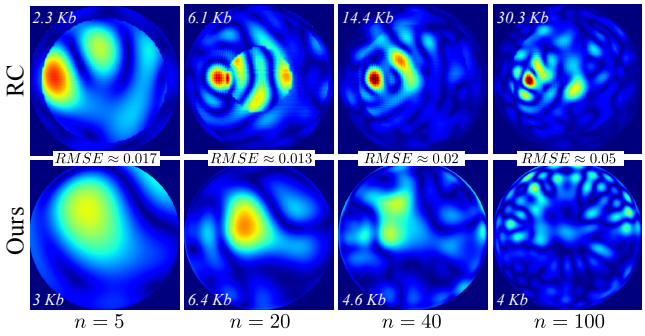


Fig. 8: Quality and memory storage comparisons between our radiance approximation with EAS (bottom line) and hemispherical harmonics used in the original radiance caching [9] method (top line), for various modified Phong BRDF exponents n. False colors illustrate the difference with a Monte-Carlo reference solution (in red for high differences and dark blue for low differences). For $n = 100$ our method requires 7.5 times less memory than RC.

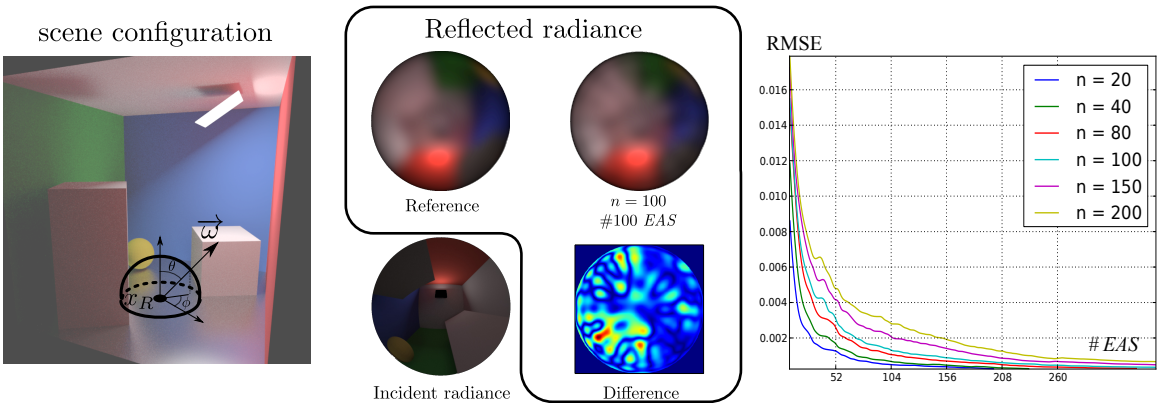


Fig. 7: (left) Record position x_R on a glossy surface with a Phong shininess $n = 100$; (middle) reflected radiance at x_R for all directions ω, and quality obtained with 100 EASs. (right) RMSE plot, according to the number of EAS for different Phong lobe exponents.

3.3 Rendering: using EASs

Our rendering process does not introduce any other change compared to usual approaches (Algorithm 1). Any 3D point visible from the camera, lying in the influence zone of a set of records, can be rendered. This section describes the management of our new representation.

Influence zone of a record

Each record of the cache is defined with its associated EASs and an influence zone of radius r. For a point P within the influence zone of a record located at x_R, we propose a geometric expression of the irradiance variations of all the EASs assigned to this record. Let us consider an EAS with a solid angle Ω, an elevation θ, a normal $\overrightarrow{N}$, and an irradiance E. Let C be the centroid of this EAS as illustrated in Figure 9. The contribution of this EAS to P provides a new irradiance E' associated with a solid angle Ω', in which P sees the EAS:

$$E'(P) = E \times \frac{d^2 \cos\theta' \cos\alpha'}{d'^2 \cos\theta \cos\alpha}, \quad \Omega' = \Omega \times \frac{d^2 \cos\alpha'}{d'^2 \cos\alpha} \tag{16}$$

where d and d' are the distance to C from x_R and P respectively. The angles α, α', θ and θ' are represented in Figure 9. Note that the factor $\frac{d^2}{\cos\theta * \cos\alpha}$ in Equation 16 is not dependent on the position P and could be precomputed and stored as an EAS parameter to speed up computations. Note also that first [23] or second order gradients [15,17] can be used to improve interpolation but at the cost of a non negligible computation and memory overhead.

General case: use of several records

A point P can be covered by the influence zone of several records. The reflected radiance (Equation 4) at this point in a direction $\overrightarrow{\omega}$ is a weighted sum of the contributions of all the records whose influence zone covers P. It is expressed as:

$$L_r(P, \overrightarrow{\omega}) \approx \frac{\sum_j^S \left(\sum_{i=1}^{\#EAS_j} E'_i \times \widetilde{F}_r(\Omega'_i, \theta'_i, \overrightarrow{\omega}) \times W_j \right)}{\sum_{j=1}^S W_j} \tag{17}$$

where S is the number of records whose influence zone covers P, $\#EAS_j$ is the number of EAS assigned to the j^{th} record and W_j is a weight depending on the distance to the j^{th} record and on the difference between the normals at the record position and P. Our implementation makes use of the same weights as in [24].

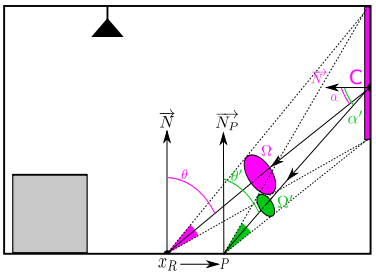


Fig. 9: Geometric variations of an EAS due to a translation from x_R to P: the EAS appearance changes after translation.

4 Results

For validation purposes, the results obtained with our method are compared with reference images generated with unbiased Path Tracing [7] (PT) with $50k$ rays traced through each pixel, and with images computed using Radiance Caching [9] (RC) together with the original library given in [9]. All the methods have been implemented within *Mitsuba renderer* [5], and ran on a 8 core Intel I7 CPU computer with 16 *GB* of RAM. Each record is processed independently of

the others, hence 8 records are processed in parallel. The images resolution are: 1080×1080 for Figures 1, 11, 12, 1920×1080 for Figure 13 and 683×512 for Figure 10.

The caching process and the record creation of *Mitsuba* do not need to be changed, our method only adds new functions for computing EASs and performing interpolation. For all our test scenes, the influence zone radius of a record is computed as the minimum distance to the objects from a record position, as in [19]. The incident radiance at a record position is computed with Mitsuba Path Tracer. A precise irradiance estimation for each EAS is obtained from $4k$ samples generated on a stratified hemisphere. We also used Mitsuba *neighbor clamping* as suggested by Krivanek et al. [8].

To demonstrate the ability of our method to handle complex lighting effects due to glossy surfaces with high shininess, we rendered 5 different scenes with specific lighting features. For each scene, the visual quality (RMSE value computed using a path-tracing reference) and the memory size of a record are evaluated for our method and RC. The differences are shown with false color pictures by means of a geometric distance, in the *rgb* space, between our results and the references. Parameters that are common to the methods have always been set to the same values. We have also compared RC with our method for high shininess exponents of the modified Phong BRDF. We have fixed the Spherical Harmonic order to 29 (note that a shininess exponent higher than 80 is clearly beyond the potential of RC [9]). Statistics for the scenes are shown in Table 1.

In general, our method outperforms the original RC method as it needs much less memory and our incident radiance representation with EAS requires less records without compromising the visual quality while reducing the rendering time. This is true for highly glossy materials ($n = 100$) but also for moderately glossy materials (*Matpreview* scene in Figure 10). In addition, our method is not subject to *ringing artifacts* often visible with spherical harmonics, as shown on the cube of the *Dragon* scene in Figure 11.

For the scenes *Cornell box* (Figure 12), *Campervan* (Figure 1) and *Kitchen* (Figure 13), the behavior and visual rendering quality of our method have been evaluated. With low memory requirements (see Table 1), our method renders results without noticeable visual artifacts, and the RMSE ranges from 0.02 to 0.04; it is particularly interesting for the scene *Campervan* (Figure 1) which corresponds to a difficult geometric configuration involving a high number of records. Increasing the number of EASs up to 300 in the *Cornell box* scene (Figure 12), definitely removes the visual artifacts of glossy inter-reflections.

Discussion and limitations

With very shiny materials (modified Phong model shininess higher than 1000), storing reflection directions does not make sense anymore, importance sampling Monte Carlo would be more efficient. However, our experiments have been successful with a shininess n ranging from 1 to $n = 500$ while requiring a moderate number of EASs. Perfectly diffuse surfaces can be rendered with a few number of EASs (up to 9) using our approach or a classical irradiance cache.

To sum up, our method can be embedded in a rendering framework to handle materials ranging from Lambertian to highly glossy (isotropic and anisotropic materials as well as micro-facet based BRDF models), while perfect specular materials are rendered with a classical path tracing (Figures 14 and 15).

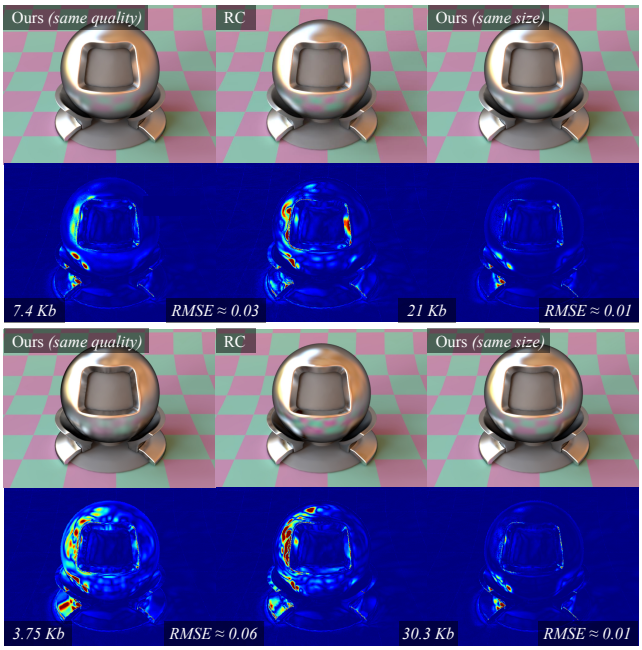


Fig. 10: Comparison (in terms of both visual quality and record memory size) of our approach and RC with *Matpreview* scene, for $n = 40$ (first two rows) and $n = 100$ (last two rows) lighted by an environment map; records store direct and indirect lighting from the environment map for both methods; record sizes are mentioned in kilobytes. Complete statistics are summarized in Table 1 (scene by Jonas Pilo and environment map is courtesy of Bernhard Vogl).

5 Conclusion

This paper described a new radiance caching method capable of efficiently handling scenes containing very glossy surfaces. Our approach relies on an original representation of the lighting environment called EAS (Equivalent Area light Sources), requiring a reasonable memory capacity. Complex incident illumination at a record position can be represented by a small number of EASs, even for highly glossy surfaces, whereas hemispherical or spherical harmonics based

	Matpreview (Fig. 10)				Dragon (Fig. 11)		Campervan (Fig. 1)		Kitchen (Fig. 13)	
	Phong exponent $n = 40$		Phong exponent $n = 100$							
	RC	**Ours***	RC	**Ours***	RC	**Ours***	RC	**Ours**	RC	**Ours**
Lighting	Indirect + Direct				Indirect		Indirect + Direct		Indirect	
# glossy rec.	2740	**2585**	3103	**2348**	20000	**16800**	21654	**24316**	6638	**3841**
record size (*Kb*)	14.4	**7.4**	30.3	**15.6**	14.4	**7.5**	30.3	**1.9**	30.3	**19.5**
SH order / **# EAS**	20	**189**	29	**399**	20	**192**	29	**50**	29	**500**
Total memory (Mb)	39.4	**19.1**	94	**36.6**	288	**126**	658	**168**	194	**79.3**
Rendering time	20 min.	**12 min.**	30 min.	**16 min.**	2.5 h.	**2 h.**	4.5h	**3h**	3.5h	**2h**
RMSE	**0.03**		**0.06**		**0.06**		0.27	**0.1**	0.05	**0.03**

Table 1: Comparison with the original Radiance Caching (RC): our method performs better (lower *RMSE*) with lower memory and faster rendering time. (Ours*) corresponds to a parameter tuning of our method that produces the same *RMSE* as RC. The figures also show the RMSE for both methods with the same record memory size.

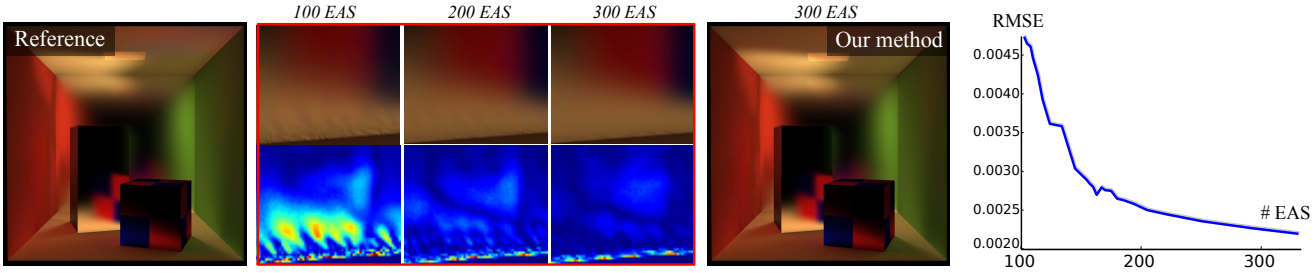


Fig. 12: **Cornell Box** scene (indirect illumination only) with glossy materials on the back wall ($n = 100$), the large cube ($n = 200$) and the small cube ($n = 40$). In the middle, close-up views of the red insert show the result quality when increasing the number of EASs . The global *RMSE* drastically decreases when the number of EASs increases (the rightmost plot).

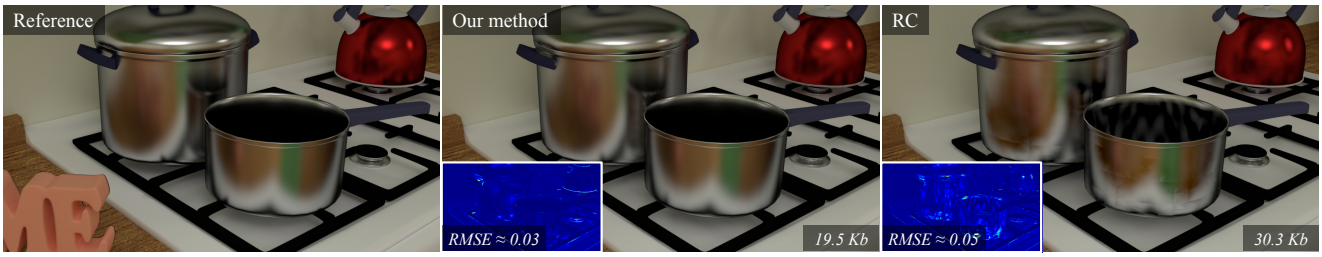


Fig. 13: **Kitchen** scene (indirect illumination only) with sharp glossy materials, $n = 200$ for the kitchenware.

methods need to store hundreds of projection coefficients together with their gradients. Moreover, our approach can be implemented in existing rendering systems such as Mitsuba or PBRT, the only functions that have to be added are those in charge of the computation of the EASs and the interpolation of the radiance at points within the influence zone of records. With our method there is no need to express and evaluate complex gradients for interpolation. However, our approach could be improved or enhanced with additional functionalities. For instance, our fitting process could be experimented with BRDFs represented by other basis functions such as radial functions or wavelets. In addition, in lighting simulation systems, a high number of artificial area light sources makes direct lighting computationally expensive, while radiance caching based methods resort to interpolation to evaluate the radiance at points that are not records. Therefore, in this case, it would be more efficient to store at each record both the direct and indirect components so that the direct radiance component could also be interpolated; this would require to account for visibility gradients.

References

1. Brady, A., Lawrence, J., Peers, P., Weimer, W.: genbrdf: Discovering new analytic brdfs with genetic programming. ACM Trans. Graph. **33**(4), 114:1–114:11 (2014)
2. Gassenbauer, V., Křivánek, J., Bouatouch, K.: Spatial directional radiance caching. Computer Graphics Forum **28**(4), 1189–1198 (2009). Eurographics Symposium on rendering, EGSR '09
3. Gautron, P., Křivánek, J., Pattanaik, S.N., Bouatouch, K.: A novel hemispherical basis for accurate and efficient rendering. In: Rendering Techniques 2004, Eurographics Symposium on Rendering, pp. 321–330. Eurographics Association (2004)

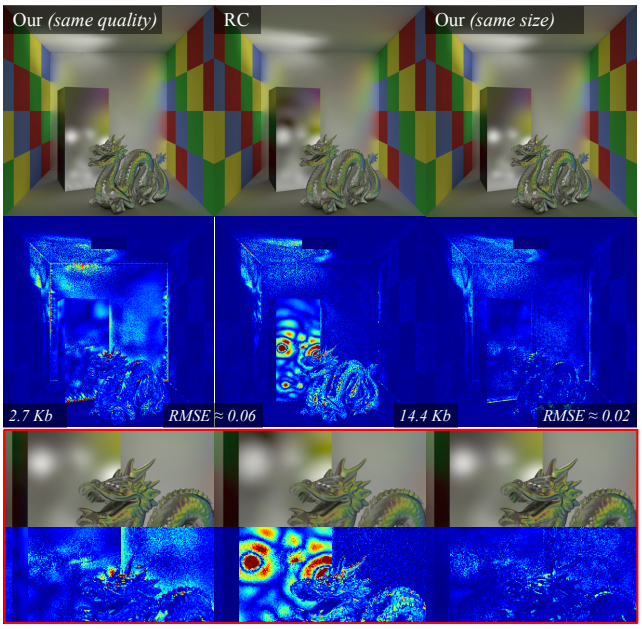


Fig. 11: Dragon scene (indirect only): record memory size is given in kilobytes (statistics are given in Table 1). False color images encode the difference with the reference solution (Phong shininess exponent $n = 20$ for the back wall, $n = 80$ for the dragon and $n = 100$ for the cube).

Fig. 14: Our method can handle different kinds of glossy materials: (1) Buddha: modified Phong BRDF with a shininess n=80, (2) teapot: Ward anisotropic BRDF model with $\alpha_x = 0.1$ and $\alpha_y = 0.5$, (3) dragon: pure specular, rendered with a path tracing technique (4) Lucy: a Micro-facet based BRDF model (with a Beckmann distribution $\alpha = 0.2$), the ground and the back wall are Lambertian.

Fig. 15: Anisotropic sphere with anisotropic Ward BRDF ($\alpha_x = 0.1$ and $\alpha_y = 0.5$).

4. Herzog, R., Myszkowski, K., Seidel, H.P.: Anisotropic radiance-cache splatting for efficiently computing high-quality global illumination with lightcuts. In: Computer Graphics Forum, vol. 28, pp. 259–268. Wiley Online Library (2009)
5. Jakob, W.: Mitsuba renderer (2010). URL http://www.mitsuba-renderer.org
6. Jensen, H.W.: Global illumination using photon maps. In: Proceedings of the eurographics workshop on Rendering techniques '96, pp. 21–30. Springer-Verlag (1996)
7. Kajiya, J.T.: The rendering equation. SIGGRAPH Comput. Graph. **20**(4), 143–150 (1986)
8. Křivánek, J., Bouatouch, K., Pattanaik, S.N., Žára, J.: Making radiance and irradiance caching practical: Adaptive caching and neighbor clamping. In: Rendering Techniques 2006, Eurographics Symposium on Rendering, pp. 127–138. Eurographics Association, Nicosia, Cyprus (2006)
9. Křivánek, J., Gautron, P., Pattanaik, S., Bouatouch, K.: Radiance caching for efficient global illumination computation. Visualization and Computer Graphics, IEEE Transactions on **11**(5), 550 – 561 (2005)
10. Křivánek, J., Georgiev, I., Kaplanyan, A., Canada, J.: Recent advances in light transport simulation: Theory and practice. In: ACM SIGGRAPH 2013 Courses, SIGGRAPH '13. ACM, New York, NY, USA (2013)
11. Lafortune, E.P., Willems, Y.D.: Bi-directional path tracing. In: COMPUGRAPHICS 93, pp. 145–153 (1993)
12. Lafortune, E.P.F., Foo, S.C., Torrance, K.E., Greenberg, D.P.: Non-linear approximation of reflectance functions. In: Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '97, pp. 117–126. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA (1997)
13. Lewis, R.R.: Making shaders more physically plausible. In: In Fourth Eurographics Workshop on Rendering, pp. 47–62 (1994)
14. Meunier, S., Perrot, R., Aveneau, L., Meneveaux, D., Ghazanfarpour, D.: Cosine lobes for interactive direct lighting in dynamic scenes. Computers & Graphics **34**(6), 767 – 778 (2010)
15. Ribardière, M., Carré, S., Bouatouch, K.: Adaptive records for irradiance caching. Computer Graphics Forum **30**(6), 1603–1616 (2011)
16. Scherzer, D., Nguyen, C.H., Ritschel, T., Seidel, H.P.: Pre-convolved Radiance Caching. Computer Graphics Forum (Proc. EGSR 2012) **4**(31) (2012)
17. Schwarzhaupt, J., Jensen, H.W., Jarosz, W.: Practical hessian-based error control for irradiance caching. ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia) **31**(6) (2012)
18. Serpaggi, X., Peroche, B.: An Adaptive Method for Indirect Illumination Using Light Vectors. Computer Graphics Forum **20**, 268–277 (2001). DOI 10.1111/1467-8659.00520
19. Tabellion, E., Lamorlette, A.: An approximate global illumination system for computer generated films. ACM Trans. Graph. **23**(3), 469–476 (2004)
20. Veach, E.: Robust monte carlo methods for light transport simulation. Ph.D. thesis, Stanford, CA, USA (1998)
21. Walter, B., Fernandez, S., Arbree, A., Bala, K., Donikian, M., Greenberg, D.P.: Lightcuts: A scalable approach to illumination. ACM Trans. Graph. **24**(3), 1098–1107 (2005)
22. Ward, G.J.: Measuring and modeling anisotropic reflection. SIGGRAPH Comput. Graph. **26**(2), 265–272 (1992)
23. Ward, G.J., Heckbert, P.S.: Irradiance Gradients. In: 1992 Eurographics Workshop on Rendering, pp. 85–98 (1992)
24. Ward, G.J., Rubinstein, F.M., Clear, R.D.: A ray tracing solution for diffuse interreflection. SIGGRAPH Comput. Graph. **22**(4), 85–92 (1988)
25. Xu, K., Cao, Y.P., Ma, L.Q., Dong, Z., Wang, R., Hu, S.M.: A practical algorithm for rendering interreflections with all-frequency brdfs. ACM Trans. Graph. **33**(1), 10:1–10:16 (2014)
26. Zaninetti, J., Serpaggi, X., Peroche, B.: A vector approach for global illumination in ray tracing. Comput. Graph. Forum **17**(3), 149–158 (1998)