Position-dependent Importance Sampling of Light Field Luminaires

Heqi Lu, Romain Pacanowski, Xavier Granier

Abstract—The possibility to use real world light sources (aka luminaires) for synthesizing images greatly contributes to their physical realism. Among existing models, the ones based on light fields are attractive due to their ability to represent faithfully the near-field and due to their possibility of being directly acquired. In this paper, we introduce a dynamic sampling strategy for complex light field luminaires with the corresponding unbiased estimator. The sampling strategy is adapted, for each 3D scene position and each frame, by restricting the sampling domain dynamically and by balancing the number of samples between the different components of the representation. This is achieved efficiently by simple position-dependent affine transformations and restrictions of Cumulative Distributive Functions that ensure that every generated sample conveys energy and contributes to the final result. Therefore, our approach only requires a low number of samples to achieve almost converged results. We demonstrate the efficiency of our approach on modern hardware by introducing a GPU-based implementation. Combined with a fast shadow algorithm, our solution exhibits interactive frame rates for direct lighting for large measured luminaires.

Index Terms—Models of Light Sources, Light Field, Importance Sampling, Real-time Rendering,

1 Motivation

Over the past twenty years, rendering techniques have significantly improved qualitatively, to obtain realistic and plausible solutions, and quantitatively to simulate physical phenomena. One of the reasons for this improvement is the introduction of increasing realistic reflectance functions from measured isotropic data [1] and, less intensively, anisotropic BRDFs [2] or complex real world light sources. In Computer Graphics, standard light sources are represented by point, directional or uniform area models. Even though point light sources may be extended with goniometric diagrams for computer graphics [3] or for professionals [4], all of these models are still limited in terms of spatial variations of emitters. The point assumption is only valid for regions of a 3D scene where the distance to the luminaire is large compared to its size. Furthermore, real luminaires are hardly approximated by uniformly emitting surfaces. For example, complex emittance function of indoor luminaires or headlights cannot be represented accurately by uniform distributions.

A classic way to improve the physical accuracy is to capture the so-called 4D *near-field* emissivity by sampling the light space around the emitter using either a *ray set* [5], [6], or more densely with a set of images [7]. Goesele et al. [8] have built an acquisition system where the 4D near-field is projected on a predefined basis, leading to a priori control of the model accuracy. Although the authors have shown that their model can be used in an interactive manner using graphics hardware [9], their dedicated approach was still limited to relatively small data sets while introducing large approximations. Despite their accuracy, their realism and the relatively simple acquisition systems required by light field luminaires, the lack of efficient and accurate rendering approaches for them is probably the reason why they are still not widely used compared to the more limited classical light models. In this paper, we demonstrate that a direct importance sampling approach is sufficient to obtain real-time and accurate solution when combined with a GPU implementation.

In order to facilitate the understanding of our contributions, we start by reminding that the near-field emission of a light source can be represented by a light field parametrized by two parallel planes (cf. Figure 1 left). A light field is well approximated by a sum of 2D images C_m weighted by the ray-dependent function $\Psi_m(\mathbf{u} \to \mathbf{s})$:

$$L(\mathbf{u} \to \mathbf{s}) = \sum_m C_m(\mathbf{s})\, \Psi_m(\mathbf{u} \to \mathbf{s}) \tag{1}$$

with $L(\mathbf{u} \to \mathbf{s})$ being the radiance transferred from $\mathbf{u}$ to $\mathbf{s}$. This notation generalizes the lumigraph [10], the lumigraph-inspired canned light sources [11] and the luminaire models of Goesele et al. [8]. More details about the relationship between Ψ_m and C_m are given in Section 3.

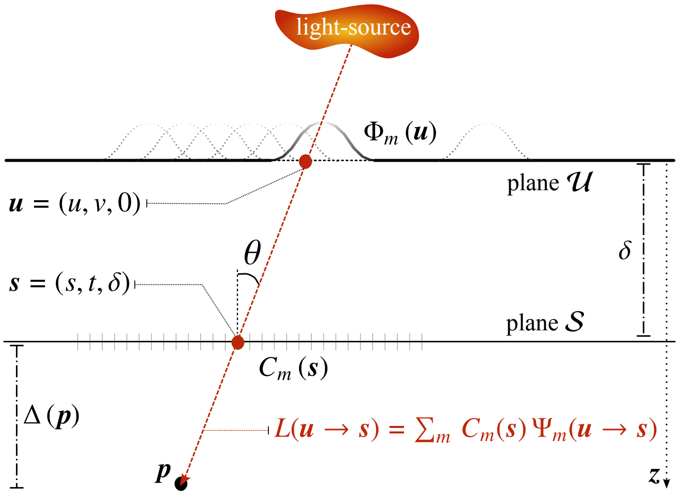
For a light source model based on light field, the irradiance $I(\mathbf{p})$ at a point $\mathbf{p}$ is defined by

$$I(\mathbf{p}) = \int_S L(\mathbf{u} \to \mathbf{p}) \frac{\Delta(\mathbf{p})}{|\mathbf{s} - \mathbf{p}|^3}\, d\mathbf{s}$$

or, in a more compact form, by

$$I(\mathbf{p}) = \Delta(\mathbf{p}) \int_S L(\mathbf{u} \to \mathbf{s}) \frac{1}{|\mathbf{s} - \mathbf{p}|^3}\, d\mathbf{s} \tag{2}$$

- *Heqi Lu is an Inria PhD student at Univ. Bordeaux, with LaBRI (CNRS : UMR5800 - Univ. Bordeaux) - 33400 Talence, France. E-mail: heqi.lu@inria.fr*
- *Romain Pacanowski is a research engineer at CNRS, with LP2N (Institut d'Optique Graduate School - CNRS : UMR5298 - Univ. Bordeaux) and with Inria - 33400 Talence, France. E-mail: romain.pacanowski@institutoptique.fr*
- *Xavier Granier is a professor at Institut d'Optique Graduate School, with LP2N (Institut d'Optique Graduate School - CNRS : UMR5298 - Univ. Bordeaux) and with Inria - 33400 Talence, France. E-mail: xavier.granier@institutoptique.fr*



Geometric Configuration	
$\mathbf{p}$	Position of a shaded point in the scene
δ	Absolute distance between planes $\mathcal{U}$ and $\mathcal{S}$
$\Delta(\mathbf{p})$	Absolute distance between $\mathbf{p}$ and plane $\mathcal{S}$
$\mathbf{u} = (u, v, 0)$	Position on plane $\mathcal{U}$
$\mathbf{s} = (s, t, \delta)$	Position on plane $\mathcal{S}$
$\mathbf{u} \to \mathbf{s}$	Ray passing through $\mathbf{u}$ in the direction of $\mathbf{s}$
θ	Angle between $\mathbf{u} \to \mathbf{s}$ and the normal of $\mathcal{S}$
Light Field Models	
$L(\mathbf{u} \to \mathbf{s})$	Radiance along the ray $\mathbf{u} \to \mathbf{s}$
$C_m(\mathbf{s})$	m^{th} image parameterized on plane $\mathcal{S}$
$\Phi_m(\mathbf{u})$	m^{th} basis function on plane $\mathcal{U}$
$[\mathbf{u}_m^{min}, \mathbf{u}_m^{max}]$	Axis-aligned bounding box support of Φ_m
$[\mathbf{s}_m^{min}(\mathbf{p}), \mathbf{s}_m^{max}(\mathbf{p})]$	Axis-aligned bounding box on plane $\mathcal{S}$ position-dependent projection of $[\mathbf{u}_m^{min}, \mathbf{u}_m^{max}]$
$\Psi_m(\mathbf{u} \to \mathbf{s})$	m^{th} reconstruction function (based on Φ_m)
Sampling	
$\text{pdf}_m^{\star}(\mathbf{s}\|\mathbf{p})$	Optimal position-dependent PDF
$\text{pdf}_m(\mathbf{s}\|\mathbf{p})$	Position-dependent PDF
$\text{cdf}_m(s\|\mathbf{p})$	Corresponding position-dependent CDFs
$\text{cdf}_m(t\|(s, \mathbf{p}))$	
$\mathbf{s}_k, \mathbf{u}_k$	k^{th} sample on $\mathcal{S}$ and $\mathcal{U}$

Fig. 1. Geometric configuration and notation table. (**Up**) Two-plane parametrization (similar to the lumigraph [10]). The 4D space of rays emitted from the light source is parametrized by a position **u** on a plane $\mathcal{U}$ and a position s on a plane $\mathcal{S}$. By definition, **u**, **s** and **p** are aligned. (**Right**). Mathematical notations used in this paper.

where $\Delta(\mathbf{p})$ is the distance between $\mathbf{p}$ and the plane $\mathcal{S}$. A **key** observation with any light field model, is that $L(\mathbf{s} \to \mathbf{p})$ is equal to $L(\mathbf{u} \to \mathbf{s})$ when assuming that no visibility events nor participating media are present. Therefore, combining Equation 1 and Equation 2 results in a new formulation for the irradiance:

$$I(\mathbf{p}) = \sum_m I_m(\mathbf{p}) \tag{3}$$

$$I_m(\mathbf{p}) = \Delta(\mathbf{p}) \int_{\mathcal{S}} C_m(\mathbf{s})\,\Psi_m(\mathbf{u} \to \mathbf{s})\,\frac{1}{|\mathbf{s} - \mathbf{p}|^3}\,d\mathbf{s}\,. \tag{4}$$

The reader should keep in mind that **u** depends on **p** and **s**. In fact, **u** is the intersection between the line $\overline{\mathbf{ps}}$ and plane $\mathcal{U}$. As detailed in the next section, methods which are both efficient and accurate have never been introduced to sample $I(\mathbf{p})$ for light field sources on a view-dependent and adaptive manner. To our knowledge, our new technique is also the *first* to achieve interactive frame rates while providing high-quality and accurate results.

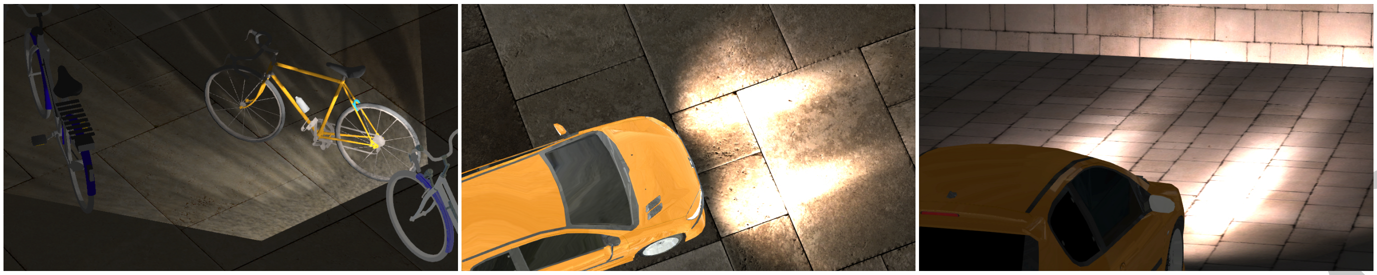
In this paper, we focus on sampling $I(\mathbf{p})$ efficiently and without introducing any bias by dynamically constructing restricted Cumulative Distribution Functions (CDFs) for each scene position **p**. More specifically, we introduce the following contributions:

- **Position-dependent Restriction of CDF (Section 3)**. We demonstrate that we can *dynamically* apply a *position-dependent* affine transformation on a CDF to *restrict* the sampling domain and consequently reduce the number of light samples without decreasing the result quality.
- **Simple Balancing** Strategy (Section 4). Additionally, we introduce an efficient balancing strategy that prevents generating light samples that convey only a small amount of energy. In other words, for each 3D scene position, our sampling strategy distributes light samples dynamically among the different light field images according to their intensity.
- **GPU implementation (Section 6)**. We demonstrate a GPU implementation of our CDF restriction and balancing strategy that reaches interactive frame rates (cf. Figure 2). Furthermore, we combine direct lighting with shadow effects by introducing a new shadow map-based algorithm that approximates the visibility.

2 Previous Work

Importance sampling is a large research area in Computer Graphics. In this paper, we focus only on importance sampling to compute direct lighting from real world luminaires. Despite the recent progress in global illumination, direct lighting is still a very important step for any computation since it is always the first one and since, in most cases, it still greatly contributes to the final quality. This is even more true for interactive global illumination techniques as detailed in the state-of-the-art of Ritschel et al. [12]. Since we are focusing our work only on light source importance sampling, we do not review techniques that apply to BRDF or visibility, or the product of both. Our approach is complementary to these solutions and we discuss this point in Section 8.

One possible solution to integrate complex real world luminaires is to use Photon Mapping [13] as demonstrated by Goesele et al. [8]. Despite recent improvements in interactive Photon Mapping [14] a final gathering [15] pass is still required to capture accurately all the details of direct illumination. Recently, Progressive Photon Mapping [16] has greatly improved the quality control by progressively and adaptively reducing the search neighborhood in order to balance between noise and bias reduction. However, to reach high-quality images, direct lighting requires a large number of passes and



12fps - 200spp 9fps - 200spp 7fps - 200spp

Fig. 2. Our new light importance sampling technique estimates, at interactive frame rate, direct lighting with only 200 samples per pixel (spp) that are distributed among the different images of the light field luminaire. (**Left**) The bike headlight light field contains 9×7 images (300×300 pixels). The visibility is computed with 80 shadow maps (256×256 pixels). (**Center and Right**) The car headlights are represented by the same light field composed of 11×9 images (256×256 pixels).

photons. Since we focus on direct lighting, it seems more efficient to sample directly the incident field for each scene position.

For this purpose, one possible solution is to approximate complex light sources by a set of fixed point lights (e.g., the technique of Agarwal et al. [17] for environment maps). A position-dependent selection of light sources can be done by using importance resampling [18] at the additional cost to evaluate a function for each precomputed light sample.

One way to select quickly a large set of light sources, according to the 3D scene position **p**, is to organize them hierarchically. With *lightcuts* [19], the direct and indirect light sources are organized in a binary hierarchy. At rendering time, a cut is performed in the hierarchy and the corresponding nodes are used for computation. Lightcuts have been used for light-based importance sampling [20] but with the original limitation to constant or cosine-distribution luminaires. This technique has been extended to spherical light-sources [21], but they may not be directly used on light field luminaires. Our proposed approach does not require any conversion of the original data set into a set of lightcuts-compatible light sources.

Structured data such as light fields can be organized hierarchically by projecting them onto a wavelet basis. The technique introduced by Clarberg et al [22] and later improved [23] uses Haar basis for both the lighting space (2D environment maps) and the BRDF space. The product is evaluated on the fly and then used to guide the sample distribution. The memory and computation costs limit their approach to low resolution approximation of BRDF and light sources. This limitation has been later reduced by Cline et al. [24] thanks to a hierarchical splitting of the environment map guided by BRDF peaks. However, all of these techniques have been developed for far-field 2D lighting where incoming lighting is independent of the 3D scene position. Therefore, they are not directly applicable to 4D light field luminaires because near-field effects lead to a different incoming lighting for each 3D scene position.

To our knowledge, only two techniques deal with complex light field luminaires. The first one is the work of Granier et al. [9], which has poor performance on small models with quite large approximations. In the second one [6], importance sampling is done according to the direct map (i.e., a set of particle emitted from the luminaire). Bias may be introduced when a low number of particles is used if the importance sampling function is not conservatively reconstructed. A too conservative approach may generate samples with negligible contribution that correspond to rays with low or null energy. As pointed by Cline et al. [24], the coarser the approximation, the greater the risk to generate these useless samples. In our approach, we stay as close as possible to the original data, without introducing any approximation: this ensures that we render almost all details that were originally measured by the acquisition process. Furthermore, our importance sampling closely mimics the behavior of the luminaire without introducing any bias: it quickly converges to the desired results with a low number of samples.

3 Position-dependent Affine-transformed CDF

The key idea of our approach is to define a sampling strategy that depends on the scene position **p**. This is achieved by using a position-dependent Probability Density Function (PDF) denoted by $\mathrm{pdf}_m(\mathbf{s}|\mathbf{p})$. With such PDF, irradiance from the light source due to the image $I_m(\mathbf{p})$ (cf. Equation 4) is estimated by generating K_m random samples $\mathbf{s}_k$:

$$I_m(\mathbf{p}) \simeq \Delta(\mathbf{p}) \frac{1}{K_m} \sum_{k=1}^{K_m} \frac{C_m(\mathbf{s}_k)\,\Psi_m(\mathbf{u}_k \to \mathbf{s}_k)}{|\mathbf{s}_k - \mathbf{p}|^3} \frac{1}{\mathrm{pdf}_m(\mathbf{s}_k|\mathbf{p})} \qquad (5)$$

where $\mathbf{u}_k$ is the intersection of the line $\overline{\mathbf{s}_k\mathbf{p}}$ with the plane $\mathcal{U}$ and $\Psi_m(\mathbf{u} \to \mathbf{s})$ is a reconstruction function (cf. Figure 1). In the original canned light source model [11], $\Psi_m(\mathbf{u} \to \mathbf{s}) = \Phi_m(\mathbf{u})$ where Φ_m is a piecewise bilinear interpolation function. In this paper we use the Goesele et al. [8] model $\Psi_m(\mathbf{u} \to \mathbf{s}) = \Phi_m(\mathbf{u})\,\delta^2 \cos^{-4}\theta$ where Φ_m a piecewise biquadratic function. The derivation from the original model to the one from Goesele is presented in the supplemental material.

The optimal PDF $\mathrm{pdf}_m^\star$ is proportional to

$$\mathrm{pdf}_m^\star(\mathbf{s}_k|\mathbf{p}) \propto C_m(\mathbf{s}_k)\,\Psi_m(\mathbf{u}_k \to \mathbf{s}_k)|\mathbf{s}_k - \mathbf{p}|^{-3}$$

since it leads to a null variance of the estimator when evaluating $I_m(\mathbf{p})$. Since a generic analytical and invertible form of the integral $\mathrm{pdf}_m^\star(\mathbf{s}_k|\mathbf{p})$ does not exist, a direct use for importance sampling is impossible.

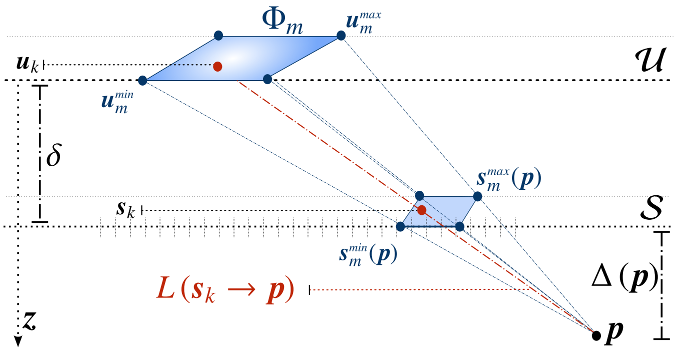


Fig. 3. Projected support of Φ_m on $\mathcal{S}$ for a given position $\mathbf{p}$. Φ_m is strictly positive over a 2D axis-aligned box bounded by $\mathbf{u}_m^{min}$ and $\mathbf{u}_m^{max}$. The projection of this axis-aligned bounding box on $\mathcal{S}$ is still an axis-aligned box bounded by $\mathbf{s}_m^{min}(\mathbf{p})$ (resp. $\mathbf{s}_m^{max}(\mathbf{p})$), which is the intersection of the plane $\mathcal{S}$ with the line $\overline{\mathbf{p}\,\mathbf{u}_m^{min}}$ (resp. $\overline{\mathbf{p}\,\mathbf{u}_m^{max}}$).

Consequently, we have to find a pdf_m that mimics closely $\mathrm{pdf}_m^\star$ while achieving a low variance. For this purpose, **two properties are required**: (i) the generated samples do not introduce any bias in the estimator that is, pdf_m has to ensure that it generates random samples at any position where $\mathrm{pdf}_m^\star(\mathbf{s}_k|\mathbf{p}) \neq 0$, and (ii) each sample must convey some energy that is, $|\mathbf{s}_k - \mathbf{p}|^{-3}\, C_m(\mathbf{s}_k)\, \Psi_m(\mathbf{u}_k \rightarrow \mathbf{s}_k) \neq 0$. (i) and (ii) are achieved by guaranteeing that:

$$\mathrm{pdf}_m(\mathbf{s}_k|\mathbf{p}) \neq 0 \Leftrightarrow \mathrm{pdf}_m^\star(\mathbf{s}_k|\mathbf{p}) \neq 0\,.$$

Since $|\mathbf{s}_k - \mathbf{p}|^{-3} > 0$, and since for existing luminaire models [11][8], with the definitions introduced in Figure 1, $\Psi_m(\mathbf{u}_k \rightarrow \mathbf{s}_k) \neq 0 \Leftrightarrow \Phi_m(\mathbf{u}_k) \neq 0$, it implies

$$\mathrm{pdf}_m(\mathbf{s}_k|\mathbf{p}) \neq 0 \Leftrightarrow C_m(\mathbf{s}_k) \neq 0 \ \text{ and } \ \Phi_m(\mathbf{u}_k) \neq 0\,.$$

The special case when $\mathbf{p}$ is on $\mathcal{S}$ is explained in Section 3.3.

3.1 Precomputed CDFs

The condition $C_m(\mathbf{s}_k) \neq 0$ is fulfilled by computing samples according to the images C_m. This corresponds to the following CDFs for s (i.e., the 1D $\mathrm{cdf}_m(s)$) and for t knowing s (i.e., the 2D $\mathrm{cdf}_m(t|s)$):

$$\begin{aligned}\mathrm{cdf}_m(s) &= \frac{\int_{-\infty}^{s}\int_{-\infty}^{+\infty} C_m(\sigma,\tau)\mathrm{d}\sigma\,\mathrm{d}\tau}{\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} C_m(\sigma,\tau)\mathrm{d}\sigma\,\mathrm{d}\tau}\\ \mathrm{cdf}_m(t|s) &= \frac{\int_{-\infty}^{t} C_m(s,\tau)\mathrm{d}\tau}{\int_{-\infty}^{+\infty} C_m(s,\tau)\mathrm{d}\tau}\end{aligned} \tag{6}$$

where $\mathbf{s} = (s, t, \delta)$ (cf. Figure 1). Assuming C_m is a simple image that is, a piecewise constant and positive function, $\mathrm{cdf}_m(s)$ is a 1D piecewise linear function and $\mathrm{cdf}_m(t|s)$ is a 2D piecewise function, linear in t and constant in s. Therefore, they can be **exactly represented as precomputed 1D and 2D tables**, relying on hardware linear interpolation. Such a derivation may be easily extended to higher order reconstructions for images like piecewise bilinear ones [11].

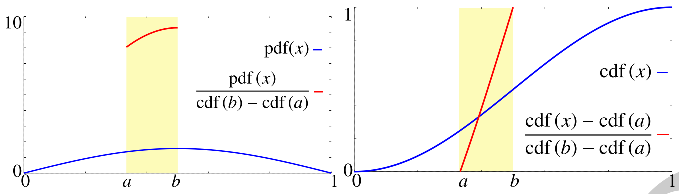


Fig. 4. The original PDF (**left**, in blue) and its corresponding CDF (**right**, in blue) are defined on the interval $[0, 1]$. Restricting the sampling to the interval $[a, b]$ is done according to a new pdf (**left**, in red) that is a rescaled version of the original one. The corresponding new cdf (**right**, in red) is obtained by an affine transformation.

3.2 Restricted CDFs

We also do not want to generate samples for which $\Phi_m(\mathbf{u}_k) = 0$ because they do not convey any energy. Since the Φ_m are defined as the product of two compactly-supported 1D functions [11], [8], the validity domain of samples (i.e., $\Phi_m(\mathbf{u}_k) \neq 0$) is an axis-aligned bounding box defined by $\mathbf{u}_m^{min} < \mathbf{u}_k < \mathbf{u}_m^{max}$. According to their definition, $\mathbf{u}_k$, $\mathbf{s}_k$ and $\mathbf{p}$ are aligned (cf. Figure 3) leading to the position-dependent condition on samples:

$$\mathbf{s}_m^{min}(\mathbf{p}) < \mathbf{s}_k < \mathbf{s}_m^{max}(\mathbf{p})$$

$$\text{with} \quad \begin{pmatrix}\mathbf{s}_m^{min}(\mathbf{p})\\ \mathbf{s}_m^{max}(\mathbf{p})\end{pmatrix} = \frac{\delta}{\delta + \Delta(\mathbf{p})}\begin{pmatrix}\mathbf{p} - \mathbf{u}_m^{min}\\ \mathbf{p} - \mathbf{u}_m^{max}\end{pmatrix} \tag{7}$$

where $\mathbf{s}_m^{min}(\mathbf{p})$ (resp. $\mathbf{s}_m^{max}(\mathbf{p})$) is the intersection of line $\overline{\mathbf{p}\,\mathbf{u}_m^{min}}$ (resp. $\overline{\mathbf{p}\,\mathbf{u}_m^{max}}$) with $\mathcal{S}$. $\mathbf{s}_m^{min}$ and $\mathbf{s}_m^{max}$ represent the axis-aligned bounding box corners of the restricted sampling domain.

Our main idea is to restrict the sample generation to this domain. This is achieved by a simple position-dependent affine transformation of precomputed CDFs.

To illustrate the core idea, we consider the 1D case illustrated in Figure 4. Given a known PDF, denoted pdf, defined on $[0, 1]$ and its corresponding CDF, denoted cdf, it is very easy to restrict the sampling to a sub-interval $[a, b]$. The new sampling strategy corresponds to a new conditional PDF defined on $[a, b]$ which is a rescaled version of the original PDF:

$$\mathrm{pdf}(x|x \in [a, b]) = \frac{\mathrm{pdf}(x)}{\mathrm{cdf}(b) - \mathrm{cdf}(a)}\,.$$

The corresponding conditional CDF, also defined on $[a, b]$, is obtained by a simple affine transformation:

$$\mathrm{cdf}(x|x \in [a, b]) = \frac{\mathrm{cdf}(x) - \mathrm{cdf}(a)}{\mathrm{cdf}(b) - \mathrm{cdf}(a)}\,.$$

This means that for a given 1D CDF, we can exactly compute its restriction to a sub-interval from its definition domain.

As for higher dimensions [25], the same process may be applied to each conditional 1D CDF. The only condition is that the restricted domain where pdf is an axis-aligned bounding box. Fortunately, it applies to our case: the restricted domain is bounded by $\mathbf{s}_m^{min}(\mathbf{p})$ and $\mathbf{s}_m^{min}(\mathbf{p})$. On this domain, the CDFs are transformed to:

$$\begin{aligned}\mathrm{cdf}_m\left(s|\mathbf{p}\right) &= \frac{\mathrm{cdf}_m\left(s\right)-\mathrm{cdf}_m\left(s_m^{min}\left(\mathbf{p}\right)\right)}{\mathrm{cdf}_m\left(s_m^{max}\left(\mathbf{p}\right)\right)-\mathrm{cdf}_m\left(s_m^{min}\left(\mathbf{p}\right)\right)}\,,\\ \mathrm{cdf}_m\left(t|\left(s,\mathbf{p}\right)\right) &= \frac{\mathrm{cdf}_m\left(t|s\right)-\mathrm{cdf}_m\left(t_m^{min}\left(\mathbf{p}\right)\middle|\,s\right)}{\mathrm{cdf}_m\left(t_m^{max}\left(\mathbf{p}\right)|\,s\right)-\mathrm{cdf}_m\left(t_m^{min}\left(\mathbf{p}\right)|\,s\right)}\,.\end{aligned} \tag{8}$$

Importance sampling based on these CDFs generates samples that have the two properties (i) $C_m(\mathbf{s}_k) \neq 0$ is preserved and (ii) $\Phi_m(\mathbf{u}_k) \neq 0$ due to the restriction. We thus ensure the two conditions to have an **unbiased and efficient** sampling strategy except on the plane $\mathcal{S}$, which is a special case detailed in the next section.

3.3 Special case: p on image plane $\mathcal{S}$

We have pointed out that the strategy is not defined for $\mathbf{p} \in \mathcal{S}$. Note that due to the lighting model, either the choice of plane $\mathcal{S}$ or $\mathcal{U}$ will lead to such a discontinuity. However, the same intensity as in Equation 3 may be computed by integrating on the plane $\mathcal{U}$:

$$I\left(\mathbf{p}\right) = \left(\Delta(\mathbf{p})+\delta\right)\int_{\mathcal{U}} L\left(\mathbf{u}\to\mathbf{p}\right)\,\frac{1}{\left|\mathbf{u}-\mathbf{p}\right|^3}\mathrm{d}\mathbf{u}\,.$$

On the image plane $\mathcal{S}$ where $\Delta(\mathbf{p}) = 0$ we thus have

$$I\left(\mathbf{s}\right) = \delta\int_{\mathcal{U}} L\left(\mathbf{u}\to\mathbf{s}\right)\,\frac{1}{\left|\mathbf{u}-\mathbf{s}\right|^3}\mathrm{d}\mathbf{u}\,.$$

By using the same approach for obtaining Equation 4, we obtain

$$I_m\left(\mathbf{s}\right) = \delta\, C_m\left(\mathbf{s}\right)\int_{\mathcal{U}} \Psi_m\left(\mathbf{u}\to\mathbf{s}\right)\,\frac{1}{\left|\mathbf{u}-\mathbf{s}\right|^3}\mathrm{d}\mathbf{u}\,. \tag{9}$$

With the same approach described in the introduction of the current Section 3, we can demonstrate that a uniform sampling $\mathbf{u}_k$ on the support of Φ_m leads to an efficient and unbiased estimator.

4 Balancing the Number of Samples

The PDF corresponding to the sampling strategy introduced in the previous section is obtained by derivation of the restricted CDFs: on the axis-aligned box bounded by $\mathbf{s}_m^{min}(\mathbf{p})$ and $\mathbf{s}_m^{max}(\mathbf{p})$, pdf_m is defined by:

$$\mathrm{pdf}_m\left(\mathbf{s}_k|\mathbf{p}\right) = \frac{C_m\left(\mathbf{s}\right)}{A_m\left(\mathbf{p}\right)} \quad \text{with} \quad A_m\left(\mathbf{p}\right) = \int_{\mathbf{s}_m^{min}(\mathbf{p})}^{\mathbf{s}_m^{max}(\mathbf{p})} C_m\left(\mathbf{s}\right)\mathrm{d}\mathbf{s}\,. \tag{10}$$

As demonstrated in the previous sections, this PDF has the required properties to closely mimic the behavior of the optimal one $\mathrm{pdf}_m^\star$. The resulting estimator defined in Equation 5 simplifies to

$$I_m\left(\mathbf{p}\right) = \frac{A_m\left(\mathbf{p}\right)\,\Delta\left(\mathbf{p}\right)}{K_m}\sum_{k=1}^{K_m}\Psi_m\left(\mathbf{u}_k\to\mathbf{s}_k\right)\,\frac{1}{\left|\mathbf{s}_k-\mathbf{p}\right|^3}\,. \tag{11}$$

Estimating each I_m using the same number of samples is straightforward and can be easily parallelized. However, if $I_m(\mathbf{p}) = 0$, K_m samples that have negligible contribution will still be evaluated leading to unwanted processing. With our sampling strategy $I_m(\mathbf{p}) = 0$ if and only if $A_m(\mathbf{p}) = 0$. We thus use $A_m(\mathbf{p})$ to balance the number of samples among the different images. By introducing

$$K_m\left(\mathbf{p}\right) = \left\lfloor \frac{A_m\left(\mathbf{p}\right)}{A\left(\mathbf{p}\right)}K \right\rceil \qquad \text{with} \qquad A\left(\mathbf{p}\right) = \sum_m A_m\left(\mathbf{p}\right) \tag{12}$$

where $\lfloor x \rceil$ denotes the closest integer approximation of x and K is a global control on the number of samples, we estimate $I_m(\mathbf{p})$ as follows:

$$I_m\left(\mathbf{p}\right) = \frac{A_m\left(\mathbf{p}\right)\,\Delta\left(\mathbf{p}\right)}{K_m\left(\mathbf{p}\right)}\sum_{k=1}^{K_m(\mathbf{p})}\Psi_m\left(\mathbf{u}_k\to\mathbf{s}_k\right)\,\frac{1}{\left|\mathbf{s}_k-\mathbf{p}\right|^3}. \tag{13}$$

Finally $I(\mathbf{p})$ is estimated by accumulating the computed values of $I_m(\mathbf{p})$. Readers may note that the balancing strategy is still dependent on the 3D scene position $\mathbf{p}$ due to the use of $A_m(\mathbf{p})$. The entire sampling strategy is therefore completely adapted to each position dynamically. It is also worth noticing that when $A_m(\mathbf{p}) = 0$ for all m no samples will be generated. This corresponds to regions of the scene that may not be reached by rays emitted from the luminaire (without taking into account visibility). Finally, for the special case where $\mathbf{p} = \mathbf{s}$, the number of samples is trivially balanced according to $C_m(\mathbf{s})$.

5 Generic Shading Estimator

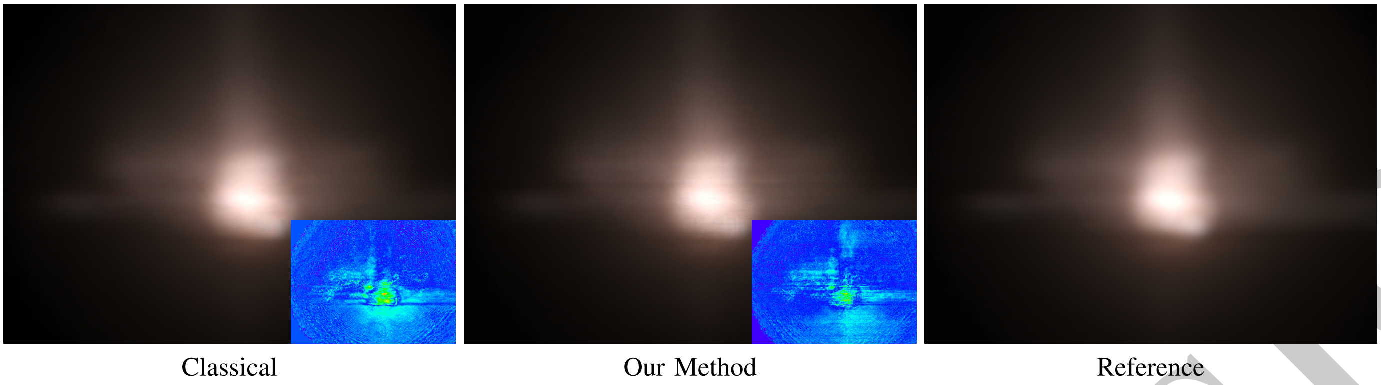
Until now, we have assumed that $C_m(\mathbf{s})$ is a scalar positive value. However, we can easily extend our approach to colored images where $\mathbf{C}_m(\mathbf{s})$ is a three-component vector. For this purpose, we set $C_m(\mathbf{s})$ to be the luminance of $\mathbf{C}_m(\mathbf{s})$ and we store an additional texture per image containing $\mathbf{c}_m(\mathbf{s}) = \mathbf{C}_m(\mathbf{s})/C_m(\mathbf{s})$. During the shading estimation, the intensity conveyed by a sample $\mathbf{s}_k$ is scaled by $\mathbf{c}_m(\mathbf{s}_k)$. Compared to the solution described in the paper, for a light field of M images, this approach requires M additional 2D RGB textures to represent the light source color.

Finally, combining BRDF ρ_k and visibility v_k evaluated for each sample $\mathbf{s}_k$ with luminaire color $\mathbf{c}_m(\mathbf{s}_k)$ leads to the following **final estimator** for reflected radiance:

$$I_m\left(\mathbf{p}\right) = \frac{A_m\left(\mathbf{p}\right)\,\Delta\left(\mathbf{p}\right)}{K_m\left(\mathbf{p}\right)}\sum_{k=1}^{K_m(\mathbf{p})}\rho_k\,v_k\,\mathbf{c}_m\left(\mathbf{s}_k\right)\,\frac{\Psi_m\left(\mathbf{u}_k\to\mathbf{s}_k\right)}{\left|\mathbf{s}_k-\mathbf{p}\right|^3} \tag{14}$$

where $K_m(\mathbf{p})$ is given in Equation 12. This estimator does not introduce any spatial bias as long as different random sequences are used between neighboring pixels (or their corresponding position $\mathbf{p}$). Furthermore, as we said in Section 3, our sampling technique does not introduce any bias.

The generation of the $\mathbf{s}_k$ samples relies on the inversion of the CDF functions. Instead of using a brute force binary search, we take advantage of the fact that our CDFs are piecewise linear and monotonic functions and we use a modified version of the secant method. We modify it such that the recursion stops when the search interval corresponds to a linear part of the function (i.e., between two neighboring pixels). Due to the CDFs properties, our modified secant method will always converge faster to the exact solution.



Classical Our Method Reference

Fig. 5. Comparison of (**Left**) our new binary search with (**Right**) the classical binary search for the same search depth. For a search depth of three, our new binary search gives a qualitatively better results (0.89 vs. 0.93 Lab error). Both results are computed in 83ms using 200 samples per pixel (spp). The mean Lab images and errors are computed against the reference image.

The faster convergence is illustrated in Figure 5 where for a given search depth, our adaptation of the algorithm is closer to the converged solution than the classical binary search. The presented Lab errors are relative errors computed on the tone-mapped images. For all results, we use a global exponential tone-mapping operator where the radiance for each pixel is rescaled by the average radiance of a frame. For the visualization of realistic light source and from a perception point of view, we apply only a constant gamma for each scene to preserve the details as well as the high energy regions of the light source.

6 Dedicated GPU Implementation

In this section we explain the GPU specifics of our sampling strategy when used for direct lighting estimation. Our implementation uses (but is not limited to) OpenGL and OpenCL. For each C_m image we precompute its associated CDFs (cf. Equation 8). These CDFs are stored as textures of the same resolution. $\mathrm{cdf}(s)$ (resp. $\mathrm{cdf}(t|s)$) is stored as a 1D (resp. 2D) floating textures of the same resolution. Finally, we also transform each C_m into a summed area table sat_m to speed up the computation of $A_m(\mathbf{p})$.

6.1 Per frame rendering

Our rendering pipeline is based on deferred shading. For each frame, the sampling and rendering process is divided into four steps:

Step 0 - OpenGL A first G-Buffer pass is done where we construct and store the pixels' positions and normals into two floating-point textures. The shadow maps are then computed and stored for later visibility approximation (cf. Section 6.2).

Step 1 - OpenCL We perform one pass per image C_m to compute $A_m(\mathbf{p})$ and we accumulate them in a dedicated floating-point buffer to estimate $A(\mathbf{p})$. More precisely, for each pixel, we compute in parallel the boundaries $\mathbf{s}_m^{min}(\mathbf{p})$ and $\mathbf{s}_m^{max}(\mathbf{p})$ (Equation 7) and then use sat_m to evaluate efficiently $A_m(\mathbf{p})$ (Equation 10).

Step 2 - OpenCL We perform one pass per image C_m and compute per pixel shading. For each pixel in parallel, we recompute $A_m(\mathbf{p})$ as in step 1 and use the previously computed $A(\mathbf{p})$ to determine the number of samples $K_m(\mathbf{p})$. We then generate the random samples $\mathbf{s}_k$ according to the $\mathrm{cdf}_m(s|\mathbf{p})$ and $\mathrm{cdf}_m(t|s,\mathbf{p})$. For each generated sample, we accumulate its lighting contribution to the pixel by multiplying it to the BRDF and visibility terms. The complete estimator is detailed in the upcoming Equation 14.

Step 3 - OpenGL We perform a simple dynamic tone mapping using the exponential operator [26].

6.2 Efficient Shadow Approximation

Visibility has to be evaluated for each light sample $\mathbf{s}_k$ but it would be too slow for interactive rendering because they are hundreds of dynamic light sources per pixel. Therefore, we introduce a new and fast shadow algorithm that approximates the visibility.

Our approximation is based on the properties of the reconstruction basis functions Φ_m. There are two types of basis functions for our test data: original piecewise quadratic functions [8] (cf. Figure 6-left) and the quadratic B-Splines (cf. Figure 6-right). The first one is not a partition of unity and thus introduces some oscillations when acquiring a constant signal. The second one is by construction a partition of unity and therefore preferred. Note that bases are defined by the acquisition setup and that the same ones have to be used for the reconstruction. In the figure, 1D support of these bases overlaps n cells that are shared with neighboring basis functions. For the two-dimension case, each basis support overlaps n^2 cells. For each cell, we select a reference light position from which the shadow map is generated. For a light field composed of $M = W \times H$ images, we generate $(W+n-1)\times(H+n-1)$ shadow maps. Each shadow map is shared between n^2 neighboring basis functions.

To compute the reference light positions, we first select a position $\mathbf{s}_m = (s_m, t_m)$ on each C_m where $\mathrm{cdf}_m(s_m)$ and $\mathrm{cdf}_m(t_m|s_m)$ are both equals to 0.5. This roughly corresponds to the center of the high-intensity region of C_m. The reference

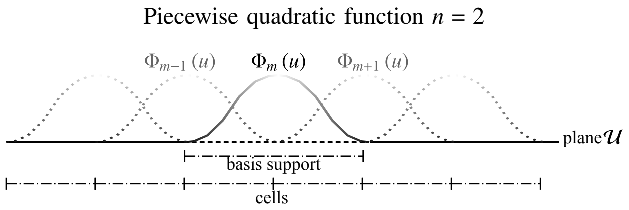


Fig. 6. Definition of cells for 1D piecewise quadratic functions Φ_m. **(Left)** For piecewise quadratic function (used in [8]) the basis support overlaps two cells whereas **(Right)** with quadratic B-Spline the basis support overlaps three cells. Each basis support overlaps n cells that are shared with neighboring basis. For M basis functions, the supporting plane is divided in $M + n - 1$ tiles.

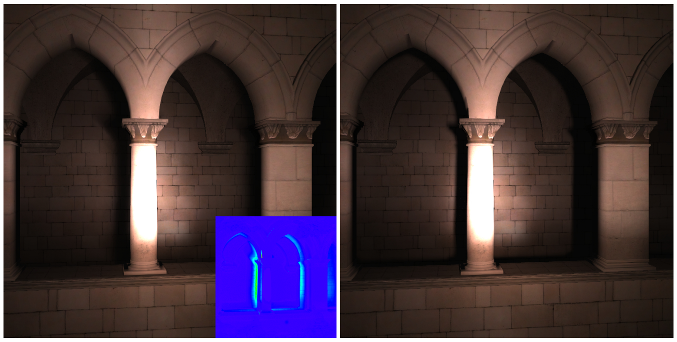


Fig. 7. Comparison of our (**Left**) approximated visibility to (**Right**) a reference solution computed with a ray-tracer. The light source used the car1 data and our algorithm is implemented with 49 shadow maps (192×192). As confirmed by the mean Lab error (1.2) the visual difference between the two images is low.

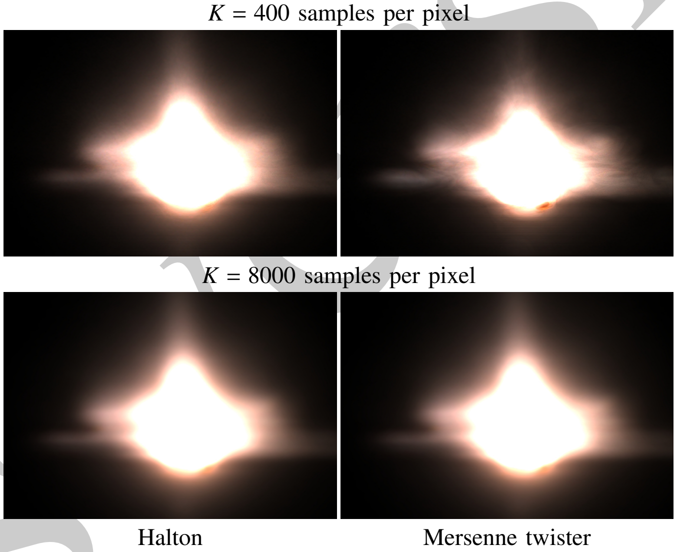


Fig. 8. Influence of the random generator according to the number of samples per pixel on the car1 data (cf. Table 1). **(Top Row)** For a low number of samples, Halton sequence leads more rapidly to better results. **(Bottom Row)** When using a large number of samples, the Mersenne twister [27] is used as reference due to its recognized quality and long period.

light position for a cell is computed as the average of the $\mathbf{s}_m$ on the corresponding images C_m whose associated basis functions overlap on the cell. As shown in Figure 7, our visibility algorithm introduces approximations compared to the ray-traced reference solution. However, these approximations are visually coherent.

Remember that our rendering pipeline executes one pass for each C_m. For each pass, instead of computing one visibility test per light sample $\mathbf{s}_k$, we compute one average visibility $v_m(\mathbf{p})$. The later is computed as the average of the shadow tests on $\mathbf{p}$ against the n^2 shadow maps from cells overlapping the basis function Φ_m.

6.3 Random Sequence Optimizations

For efficiency purpose when implementing on GPU, we use the same random sequence for each pixel. This does not introduce any per-pixel bias but only a spatial bias between pixels and thus reduces the spatial noise. Furthermore, it also improves the cache access coherency of the GPU implementation. To adjust the trade-off between speed and spatial bias, interleaved sampling [28] may be introduced. However, as shown in Figure 11 and Figure 13, the current simple strategy already gives very good results.

Since we need to distribute different number of samples among the different light field images, we choose the Halton sequence because all prefixes of the sequence are well distributed over the domain. As shown in Figure 8, due to their lower discrepancy property (cf. [29]), when used with a small number of samples, Halton sequence gives better result than the one generated with Mersenne Twister.

Performance improvements are also obtained by limiting the number of samples per pixel and per image $K_m(\mathbf{p})$ to a maximum value $K_{\max}$. As shown in Figure 9, this strategy reduces the total number of samples without introducing any bias and with a low impact on the final quality.

7 Results

All presented results are rendered at 1024 × 768 resolution, using a GTX 580 with 1.5GB on a workstation with an Intel Core i7 920 with 6 GB. The companion video shows the interactive frame rates, ranging from 7 to 15fps, of our GPU implementation. The size of the different 3D models are 200K

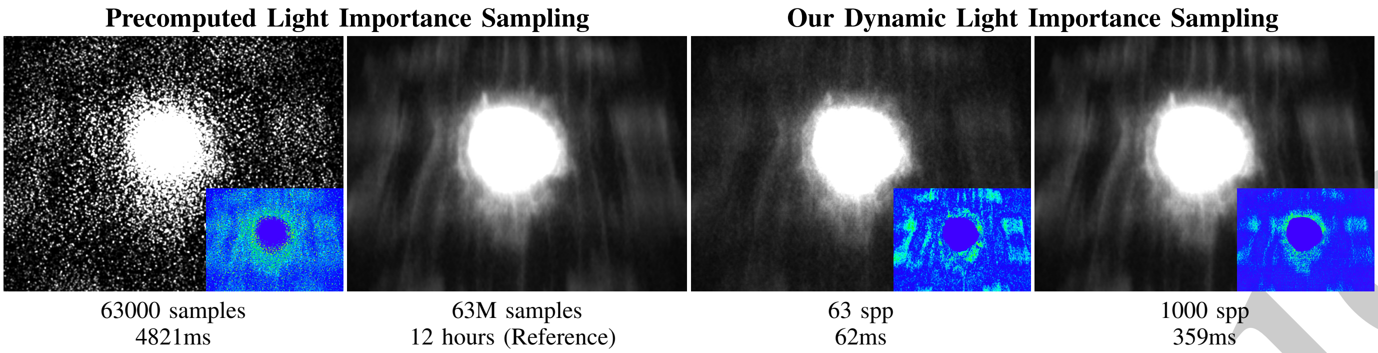

Precomputed Light Importance Sampling
Our Dynamic Light Importance Sampling
63000 samples
4821ms
63M samples
12 hours (Reference)
63 spp
62ms
1000 spp
359ms

Fig. 11. Convergence speed comparisons of precomputed light importance sampling versus our approach with per-pixel light importance sampling for the `bike` headlight data. With less samples per pixel (63 vs. 63000), our dynamic approach achieves a drastically higher quality with 1000 times less samples per pixel and is almost two orders of magnitude faster (62ms vs. 4821ms) than the precomputed approach. Lab error insets are computed against the reference image.

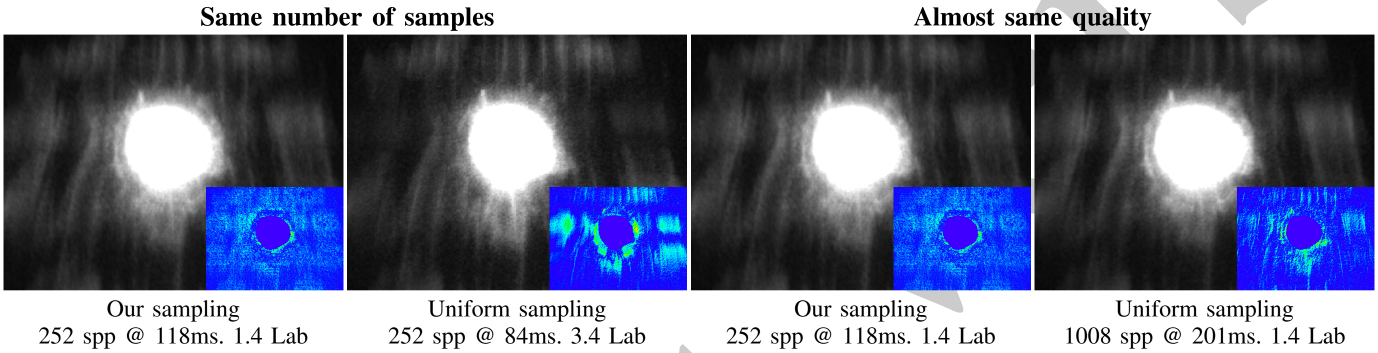


Fig. 12. Qualitative comparisons of uniform sampling versus our light importance sampling technique for the `bike` headlight data. (**Left Side**) For the same number of samples our technique is slower than the uniform approach but its mean Lab error is 2.4 times lower (1.4 vs. 3.4 Lab). (**Right Side**) For almost the same quality (1.4 Lab error), our approach requires four times less samples (252 vs. 1008). Lab error insets are computed against the reference image shown in Figure 11.

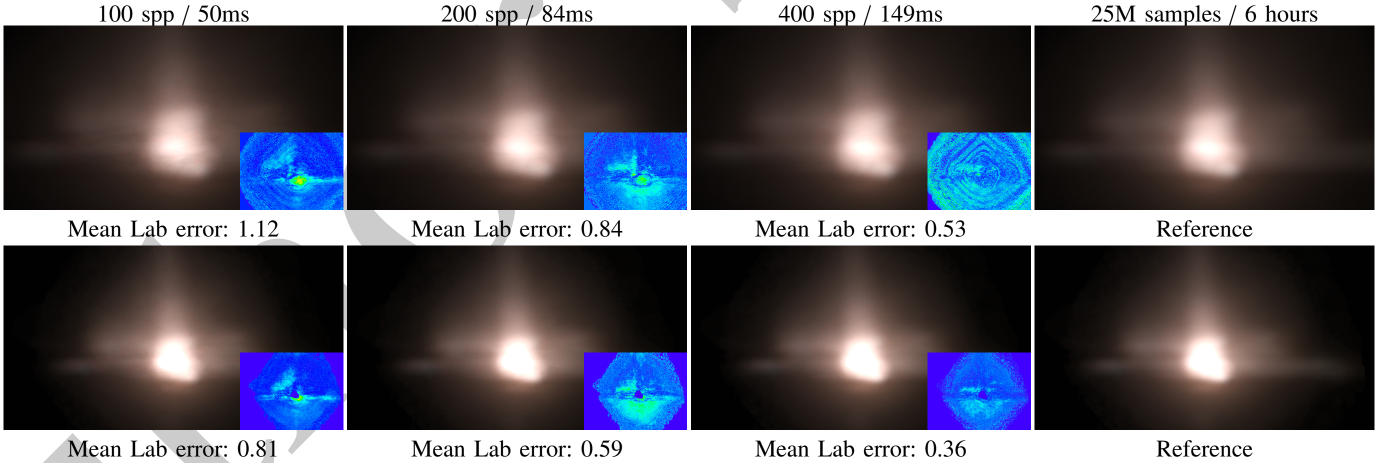


Fig. 13. Convergence of our sampling technique when increasing the number of samples per pixel (spp) for two different viewpoints: **Top** close-up view; **Bottom** farther view of the light source. The Lab errors are computed against the right-most image. Our technique achieves a very good visual quality with only 200 spp.

polygons for the bike, 7000 polygons for the car and 70K for Sponza (cf. Figures 7).

The precomputation time (CDFs, sat and reference light positions for shadow maps) for all light field data listed in Table 1 is quite low: 513ms for `bike`, 129ms for `car1` and 527ms for `car2`. Regarding the GPU memory footprint, our technique requires storing the G-Buffers (24MB) and different textures representing the light field and its CDF. More precisely, for a Light Field of M images the number of floating-point textures stored on the GPU memory is: M 2D textures for $\mathrm{cdf}_m(t|s)$, M 1D textures for $\mathrm{cdf}_m(s)$ and M

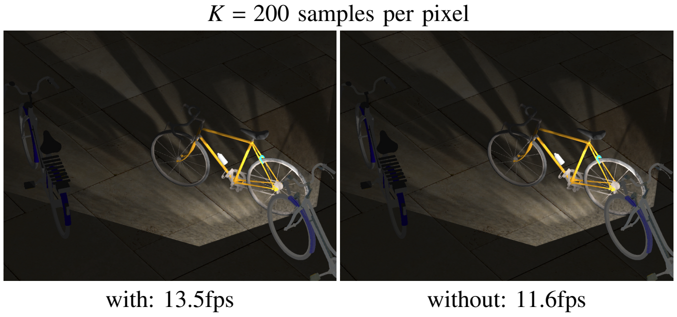


Fig. 9. Influence of the maximum number of samples. To increase the rendering efficiency, the number of samples per pixel and per images $K_m(\mathbf{p}_i)$ may be bounded to a maximum value K_{max}. In this example with the bike data (cf. Table 1), setting $K_{max} = 7$ does not introduce large visual differences since the maximum Lab error is only 23 (0.63 mean error) between the two images.

name	description	Nb. - res. of images	basis
bike	bike	9×7 - 300×300	quad. 15mm
car1	car	5×5 - 256×256	spline 49.5mm
car2	car	11×9 - 256×256	spline 49.5mm

TABLE 1
Light field data and their associated types for the Φ_m functions used in this paper. The bike data are from Goesele et al. [8] and use their dedicated quadratic functions whereas car1 and car2 are new data that use quadratic B-Splines. The size in millimeter indicates the length of the quadrangular basis support.

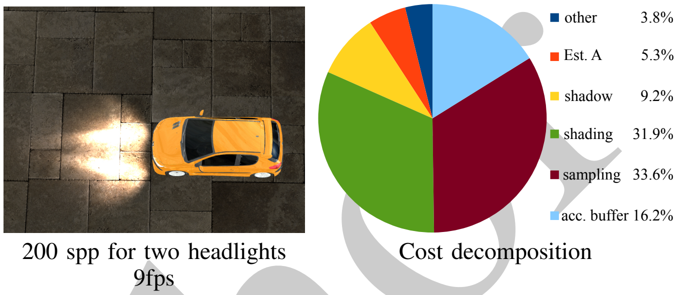


Fig. 10. Repartition of the rendering time for two headlights using car2 data. "Est. A" stands for the computation of each $A_m(\mathbf{p})$ accumulated to get $A(\mathbf{p})$ in step 1 of section 6.1. "sampling" corresponds to the sampling cost of step 2. "other" includes buffer swapping, memory and context sharing between OpenCL and OpenGL. "acc. buffer" stands for the cost of accumulating the contribution of each image C_m.

2D textures for sat. For the car2 data the whole light field and colored $\mathbf{C}_m(\mathbf{s})$ textures (cf. Section 5) add up to 123MB.

To illustrate the advantage of our sampling approach we compare the image quality between **precomputed light importance sampling** and our dynamic sampling strategies both applied on the image plane. Since Precomputed Light Importance sampling is a classical approach when one needs to sample many light sources with different energy (e.g., [30]), it is an appropriate reference solution to compare our approach with.

As shown in Figure 11, with less samples per pixel (63 vs. 63000), our dynamic approach achieves a drastically higher quality with 1000 times less samples per pixel and is almost two orders of magnitude faster (e.g., 62ms vs. 4821ms) than the precomputed approach. Moreover, as shown on the right image of Figure 11, our technique quickly converges toward the reference solution (63M samples, max search depth 15) with only 1000 samples and is several order of magnitude faster. This is explained by the fact that, with the precomputed strategy, samples are generated from global CDF functions (cf. Equation 6) and every pixel is shaded with all precomputed samples. On the contrary, with our approach, for each pixel, samples are dynamically generated on a restricted area and distributed among the light field images.

One may use almost any kind of CDF in a restricted area. Comparisons between our restricted CDF and a uniform one is shown in Figure 12. For the uniform solution, we generate the same number of samples from each texture image and sample the restricted area uniformly on the image plane without taking into account the CDFs. This approach is equivalent to sample uniformly the support of each basis. By contrast, with our importance sampling approach, for each pixel, samples are selected using the restricted CDFs, and therefore, the number of samples varies for each basis support. As illustrated in the left part of Figure 12, when comparing against the reference solution (shown in Figure 11), for the same number of samples, our technique is slower than the uniform approach but its Lab error is 2.4 times lower (1.4 vs. 3.4 Lab), thus showing that our technique converges faster. This is also illustrated in the right part of the same Figure where, for almost the same quality (1.4 Lab error), our approach requires four times less samples (252 vs. 1008). Consequently, our dynamic light importance sampling is more suitable and efficient when combined with complex BRDF since it requires less shading samples. Finally, the convergence speed of our technique is illustrated in Figure 13 with the car1 data. It is shown that our technique requires only a low number of samples per pixel to converge toward a high-quality result.

It is important to notice that the different frame rates reported in Figures 2, 10, and 14 for the car scene are due to the dynamic nature of our sampling strategy. For pixels belonging to a part of the scene that will not receive any energy, our dynamic sampling strategy prevents generating samples that would be useless. Thus, the performance also depends on the percentage of lit pixels. Figure 10 details the cost of each rendering step of Section 6 to display one frame for the two-headlight car2 scene. For 200 samples per pixel, the cost of our sampling techniques (i.e., about 39% with both step 1 and sampling) is almost the same as the shading time (i.e., 41% with shadow and shading). Although the computation of each $A_m(\mathbf{p})$ is duplicated in step 2, its cost remains negligible compared to the shading and sampling cost. This cost is only dependent on the size of the luminaire model, and its relative impact will decrease with the increasing number of samples. However, the shading and sampling costs are dependent on the number of samples and will thus stay similar. It is worth noticing that 16% of the time is spent accumulating the contribution of each light field image C_m of the luminaire model.

Finally, since we achieve interactive frame rate with real world light sources, our approach (cf. Figure 14) allows to rotate and translate light field data interactively. It would be also possible to change the parameter δ from Goesele et al. [8] models. However, it is not a user parameter but an acquisition parameter: its modification would correspond to a new luminaire that may not be a feasible physical system.

8 Discussion and Future Work

Compared to previous techniques, the main strength of our new approach is its ability to deal with 4D light field luminaires and to use acquired data directly. We do not convert the original data into an approximated alternative representation and ensure that our technique is unbiased per pixel.

Our approach is based on the fact that the two parameterizing planes are parallel. However, it can be extended to non-rectangular representation as long as we can compute a bounding box of the projected support. Unfortunately, without a two-plane parallel configuration, the projection of the axis-aligned box of the basis functions support does not remain an axis-aligned bounding box. The sampling efficiency is directly related to the ratio of the support area to the bounding box area. Hence, for future acquisition setups, it might be worth to maintain it or to combine it with multiple two-plane configurations.

Although we demonstrate interactive frame rates, we think that there are potential improvements that could be investigated to reach real-time performances. First, as shown in the previous section, 16% of the rendering cost is due to the accumulation of the C_m images, hence reducing the number of images, or their size, would be a valid approach to improve the performances. Another possibility is to use the screen space coherency as recently demonstrated for visibility (e.g., [31]).

One can also expect that better balancing strategy could also lead to some improvements. We have experimented a solution that multiplies the accumulated energy $A_m(\mathbf{p})$ for one image I_m with the solid angle of the basis support viewed from the current position: this is an accurate estimation of the contribution of one image to the final shading. Unfortunately, it increases more the performance penalty than the quality in our test scenes. For instance, we have obtained 8% computation overhead with only 5% improvement in quality for the scene shown in Figure 13. However, such an approach may be worth pursuing for more complex shading scenarios.

Our GPU-dedicated and approximated visibility algorithm based on shadow maps is currently sufficient for interactive purposes but could be improved for real-time applications. With the current solution, we have introduced a uniform selection of shadow map positions. Better strategies might certainly lead to speed and quality improvements. One possibility would be to adapt the imperfect shadow maps (ISMs) [32] where a coarse approximation of the scene geometry is used to improve the rendering speed. However, as pointed by the authors, ISMs reduce the quality of direct shadows since the high-frequency nature of direct lighting requires accurate visibility. We have shown in Figure 7 that the best shadow quality is reached when evaluating the visibility for each light sample. Unfortunately, this is currently limited to offline processing.

Finally, our light source sampling approach could be extended by taking into account the influence of BRDF and visibility terms. We envision two strategies to achieve this goal. The first one would be to incorporate our approach in a Multiple Importance Sampling framework (e.g., [33]). The second direction would be to determine how to incorporate the visibility or BRDF influence into the importance function (e.g., [24], [23]), by adjusting the number of samples per image or by defining a more accurate restriction on the sampling domain. In the case of complex specular paths, Markov Chains Monte Carlo approaches (e.g., [34]) could also be used. It is possible to use our method in Bidirectional Importance Sampling (e.g., [35], [20]) as well. One possibility is to use a two-step resampling method [35], as long as our method replaces the classical light sampling. This is straightforward, since light sampling and BRDF sampling are usually performed separately. Another possibility is to construct a per-pixel light tree and perform lightcuts [20] on it. To construct the light tree, one needs a set of point light sources that can be directly obtained with our sampling method.

9 Conclusion

In this paper, we have introduced a position-dependent importance sampling technique that can be used on light field luminaires. This technique is based on a simple position-dependent affine transformation of CDFs that dynamically restricts the sampling domain. Furthermore, our technique distributes the number of samples between the different light field images according to their intensity. Combined together, these two solutions ensure that every generated sample contributes more evenly to the final results. Our new method is unbiased per pixel and requires only a small number of light samples to reach high-quality results. Finally, we have demonstrated a GPU implementation with a new and appropriate shadow algorithm that achieves interactive results.

References

[1] W. Matusik, H. Pfister, M. Brand, and L. McMillan, "A data-driven reflectance model," *ACM Trans. Graph.*, vol. 22, no. 3, 2003.

[2] A. Ngan, F. Durand, and W. Matusik, "Experimental analysis of BRDF models," in *Proc. Eurographics Symposium on Rendering '05*, 2005, pp. 117–226.

[3] C. P. Verbeck and D. P. Greenberg, "A comprehensive light source description for computer graphics," *IEEE Computer Graphics & Applications*, vol. 4, no. 7, pp. 66–75, 1984.

[4] IESNA Committee, *Standard File Format for Electronic Transfer of Data: LM-63*, ser. Lighting Measurements Series. Illuminating Engineering Society of North America, 2001.

[5] I. Ashdown and R. Rykowski, "Making Near-Field Photometry Practical," in *1997 IESNA Conference Proceedings*. Illuminating Engineering Society of North America, 1997, pp. 368–389.

[6] A. Mas, I. Martin, and G. Patow, "Compression and Importance Sampling of Near-Field Light Sources," *Computer Graphics Forum*, vol. 27, no. 8, pp. 2013–2027, 2008.

[7] I. Ashdown, "Near-Field Photometry: Measuring and Modeling Complex 3-D Light Sources," in *SIGGRAPH 95 Course Notes - Realistic Input for Realistic Images*. ACM Press, 1995, pp. 1–15.

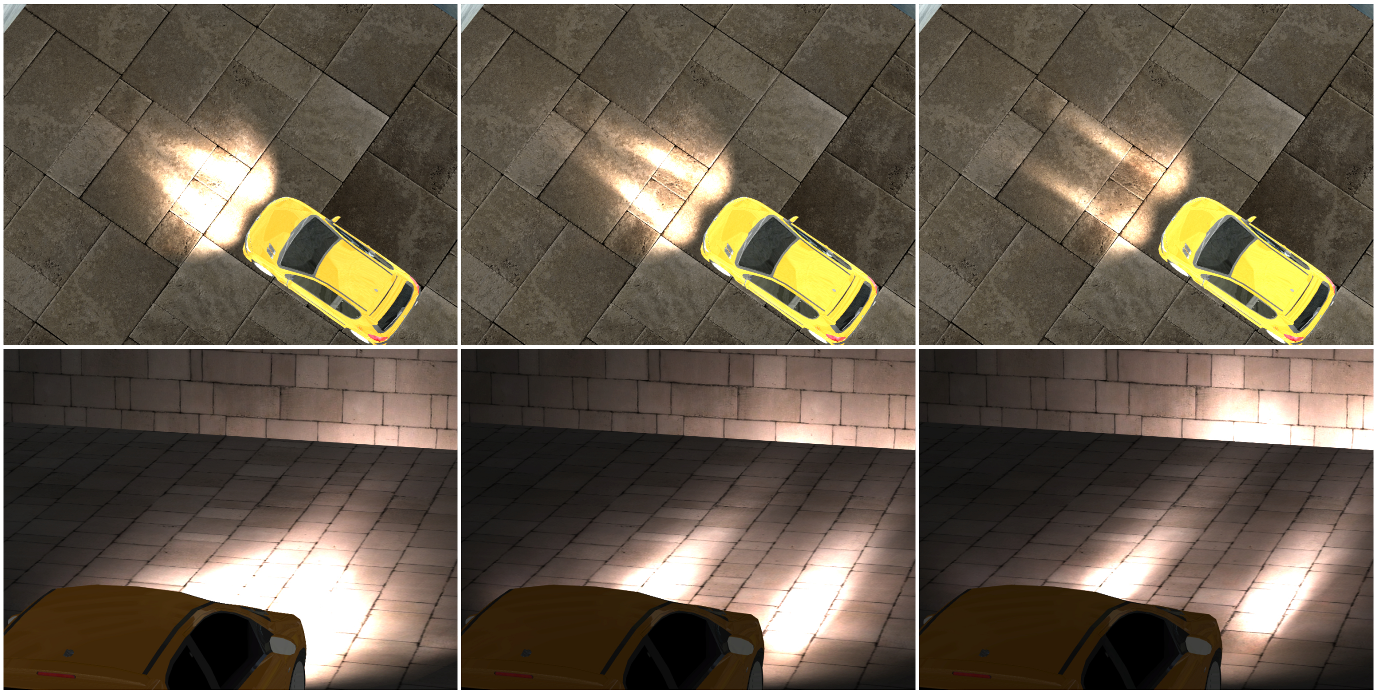


Fig. 14. Light editing for `car2` headlight data. Our technique allows to modify interactively (8fps in average) the position and orientation of captured light field.

[8] M. Goesele, X. Granier, W. Heidrich, and H.-P. Seidel, “Accurate light source acquisition and rendering,” *ACM Trans. Graph.*, vol. 22, no. 3, pp. 621–630, 2003.

[9] X. Granier, M. Goesele, W. Heidrich, and H.-P. Seidel, “Interactive Visualization of Complex Real-World Light Sources,” in *Proc. Pacific Conference on Computer Graphics and Applications*. IEEE Computer Society, 2003, pp. 59–66.

[10] S. J. Gortler, R. Grzeszczuk, R. Szelinski, and M. F. Cohen, “The Lumigraph,” in *Proc. ACM SIGGRAPH*. ACM Press/Addison-Wesley Publishing Co., 1996, pp. 43–54.

[11] W. Heidrich, J. Kautz, P. Slusallek, and H.-P. Seidel, “Canned Lightsources,” in *Rendering Techniques ’98*. Eurographics, 1998, pp. 293–300.

[12] T. Ritschel, C. Dachsbacher, T. Grosch, and J. Kautz, “The State of the Art in Interactive Global Illumination,” *Computer Graphics Forum*, vol. 31, no. 1, pp. 160–188, 2012.

[13] H. W. Jensen, *Realistic Image Synthesis Using Photon Mapping*. A. K. Peters, 2001.

[14] C. Yao, B. Wang, B. Chan, J. Yong, and J.-C. Paul, “Multi-Image Based Photon Tracing for Interactive Global Illumination of Dynamic Scenes,” *Computer Graphics Forum*, vol. 29, no. 4, pp. 1315–1324, 2010.

[15] R. Wang, R. Wang, K. Zhou, M. Pan, and H. Bao, “An efficient GPU-based approach for interactive global illumination,” *ACM Trans. Graph.*, vol. 28, no. 3, pp. 91:1–91:8, 2009.

[16] C. Knaus and M. Zwicker, “Progressive photon mapping: A probabilistic approach,” *ACM Trans. Graph.*, vol. 30, no. 3, pp. 25:1–25:13, 2011.

[17] S. Agarwal, R. Ramamoorthi, S. Belongie, and H. W. Jensen, “Structured importance sampling of environment maps,” *ACM Trans. Graph.*, vol. 22, no. 3, pp. 605–612, 2003.

[18] J. F. Talbot, D. Cline, and P. K. Egbert, “Importance resampling for global illumination,” in *Proc. Eurographics Symposium on Rendering*, 2005.

[19] B. Walter, S. Fernandez, A. Arbree, K. Bala, M. Donikian, and D. P. Greenberg, “Lightcuts: a scalable approach to illumination,” *ACM Trans. Graph.*, vol. 24, no. 3, pp. 1098–1107, 2005.

[20] R. Wang and O. Akerlund, “Bidirectional Importance Sampling for Unstructured Direct Illumination,” *Computer Graphics Forum*, vol. 28, no. 2, pp. 269–278, 2009.

[21] M. Hašan, J. Křivánek, B. Walter, and K. Bala, “Virtual spherical lights for many-light rendering of glossy scenes,” *ACM Trans. Graph.*, vol. 28, no. 5, pp. 143:1–143:6, 2009.

[22] P. Clarberg, W. Jarosz, T. Akenine-Möller, and H. W. Jensen, “Wavelet importance sampling: efficiently evaluating products of complex functions,” *ACM Trans. Graph.*, vol. 24, no. 3, pp. 1166–1175, 2005.

[23] P. Clarberg and T. Akenine-Möller, “Practical Product Importance Sampling for Direct Illumination,” *Computer Graphics Forum*, vol. 27, no. 2, 2008.

[24] D. Cline, P. K. Egbert, J. Talbot, and D. L. Cardon, “Two Stage Importance Sampling for Direct Lighting,” in *Proc. Eurographics Symposium on Rendering*, 2006, pp. 103–113.

[25] G. S. Fishman, *Monte Carlo: Concepts, algorithms, and applications*, ser. Springer Series in Operations Research. Springer-Verlag, 1996.

[26] E. Reinhard, W. Heidrich, P. Debevec, S. Pattanaik, G. Ward, and K. Myszkowski, *High Dynamic Range Imaging: Acquisition, Display and Image-Based Lighting*. Morgan Kaufmann Publishers, 2010, 2nd edition.

[27] M. Matsumoto and T. Nishimura, “Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator,” *ACM Trans. Model. Comput. Simul.*, vol. 8, no. 1, pp. 3–30, 1998.

[28] A. Keller and W. Heidrich, “Interleaved sampling,” in *Proc. Eurographics Workshop on Rendering*, 25–27 2001.

[29] H. Niederreiter, *Random number generation and quasi-Monte Carlo methods*. Society for Industrial and Applied Mathematics, 1992.

[30] H. Lu, R. Pacanowski, and X. Granier, “Real-Time Importance Sampling of Dynamic Environment Maps,” in *Eurographics Annual Conference (Short Papers)*. Eurographics, May 2013.

[31] P. Clarberg and T. Akenine-Möller, “Exploiting Visibility Correlation in Direct Illumination,” *Computer Graphics Forum*, vol. 27, no. 4, pp. 1125–1136, 2008.

[32] T. Ritschel, T. Grosch, M. H. Kim, H.-P. Seidel, C. Dachsbacher, and J. Kautz, “Imperfect Shadow Maps for Efficient Computation of Indirect Illumination,” *ACM Trans. Graph.*, vol. 27, no. 5, 2008.

[33] A. Pajot, L. Barthe, M. Paulin, and P. Poulin, “Representativity for robust and adaptive multiple importance sampling,” *IEEE Trans. Visualization and Computer Graphics*, vol. 17, no. 8, pp. 1108–1121, 2011.

[34] W. Jakob and S. Marschner, “Manifold exploration: A markov chain monte carlo technique for rendering scenes with difficult specular transport,” *ACM Trans. Graph.*, vol. 31, no. 4, pp. 58:1–58:13, Jul. 2012.

[35] D. Burke, A. Ghosh, and W. Heidrich, “Bidirectional Importance Sampling for Direct Illumination,” in *Proc. Eurographics Symposium on Rendering*, 2005, pp. 147–156.

Heqi Lu received his Master and Bachelor degrees in computer science from Zhejiang University (China) and East China Normal University respectively. He is a PhD student at University of Bordeaux (France) with an Inria / Region Aquitaine fellowship, under the supervision of prof. Xavier Granier. His research interests include physically-based rendering, image-based lighting, numerical integration using Monte-Carlo methods, and GPU computing. He is a member of the LaBRI laboratory.

Romain Pacanowski received his Engineer degree from EFREI and his MS degree from the University of Bordeaux. He holds a PhD from the University of Bordeaux (France) and Université de Montréal (Canada). He was a postdoctoral fellow at the CEA in 2010 and for Disney Interactive Media Group in 2011. Since December 2011, he is working at the CNRS as Research-Engineer at the LP2N laboratory. His research interest includes realistic rendering as well as appearance modeling and acquisition.

Xavier Granier received his Engineer and MS degrees from the Grenoble Institute of Technology and his PhD Diploma for University Joseph Fourier in Grenoble (France). He is currently a professor at Insitut d'Optique Graduate School. He is leading the mixed reality research axis in the LP2N laboratory and leading the joint research group manao between University of Bordeaux, Institut d'Optique Graduate School, Inria, and CNRS on computer graphics and mixed reality. His research interest includes realistic lighting, appearance modeling and acquisition. He is a member for ACM and EUROGRAPHICS and associate member of the LaBRI laboratory.