Pacific Graphics 2021
E. Eisemann, K. Singh, and F.-L Zhang
(Guest Editors)

Volume 40 (2021), Number 7

A multi-pass method for accelerated spectral sampling

M. van de Ruit and E. Eisemann

Delft University of Technology

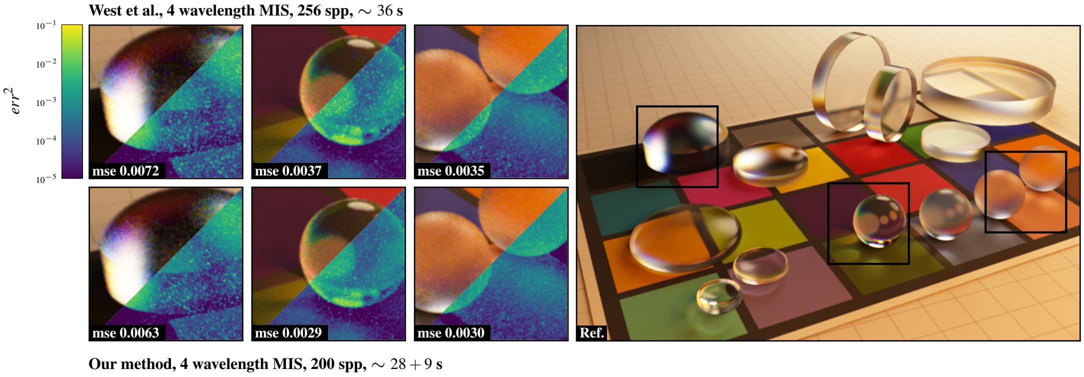

Figure 1: *Our method decreases error for wavelength-dependent scattering in the presence of non-uniformly distributed emission, reflectance, and transmission throughout the scene. We show approximately equal-time comparisons of (spectral) mean squared error between our method (bottom left) and the current state-of-the-art [WGGH20] (top left), for three areas selected from the reference image (right).*

Abstract

Spectral Monte Carlo rendering can simulate advanced light phenomena, such as chromatic dispersion, but typically shows a slow convergence behavior. Properly sampling the spectral domain can be challenging in scenes with many complex spectral distributions. To this end, we propose a multi-pass approach. We build and store coarse screen-space estimates of incident spectral radiance and use these to then importance sample the spectral domain. Hereby, we lower variance and reduce noise with little overhead. Our method handles challenging scenarios with difficult spectral distributions, many different emitters, and participating media. Finally, it can be integrated into existing spectral rendering methods for an additional acceleration.

CCS Concepts

• ***Computing methodologies*** → ***Ray tracing;***

1. Introduction

When producing photorealistic imagery, modern rendering systems typically employ advanced Monte Carlo light transport algorithms. Many of these systems are *trichromatic*, modeling all light, color, and different spectral distributions as a combination of three (RGB) values. This trichromatic approximation is known to be insufficient for accurate color reproduction [Bor91]. Further, it makes it profoundly difficult to simulate physical phenomena such as chromatic light dispersion, diffraction, fluorescence, and polarization. Although extended spectral light transport algorithms have overcome these limitations, these are far more computationally expensive. They necessitate sampling of the spectral domain, significantly increasing the sample rates required to avoid noise.

In recent years, techniques decreased spectral noise by either tackling specific problems such as path-reuse during wavelength-dependent scattering [EM99; RBA09; WND*14; WGGH20], or through the application of novel rendering techniques [PBE18]. Despite major improvements, modern spectral renderers may still converge poorly, which is partially attributed to the additional sampling of the spectral domain. Spectral sampling is generally done uniformly or with respect to sensor responses [RBA09]. This increases variance when the observed radiance becomes highly non-uniform due to a multitude of complicated emission and reflectance spectra. Although *spectral power distributions* (SPDs) can be leveraged for importance sampling [EM99; WGGH20], we show that this is not an optimal solution for many but the simplest scenarios.

Our contribution consists of an extended light transport algorithm where, before rendering, we invest time to build coarse screen-space estimates of incident spectral radiance distributions, and sample these distributions in a manner which avoids bias. This allows us to improve convergence behavior in complicated scenes with many different non-uniform spectra. We extend a unidirectional path tracer and show that our method improves performance where others may currently fall short. We additionally demonstrate a combination with the recent *continuous multiple importance sampling* (CMIS) [WGGH20], leading to further improvements.

After covering the basic spectral light transport notations and providing an overview of the state-of-the-art (Section 2), we expand on the different components of our method (Section 3). We next discuss our implementation (Section 4) and evaluate it in a variety of scenarios (Section 5) before concluding (Section 6).

2. Background and Related Work

Spectral light transport Physically-based renderers are generally concerned with evaluating the *light transport equation* [ICG86; Kaj86], for which we use an extended form of the more applicable path-integral formulation [Vea98]. We measure the spectral radiance I entering a single pixel j as

$$I_j = \int_\Lambda \int_\Omega f_j(\bar{x},\lambda)\, d\mu(\bar{x})\, d\lambda, \tag{1}$$

where Λ denotes the *spectral domain* of wavelengths and Ω is the *path space* of all light transport paths $\bar{x} = x_0,\ldots,x_{n-1}$ of finite lengths n along which light viably travels from a light source to our sensor. The light throughput for a single path and a given wavelength is then measured by $f_j(\bar{x},\lambda)$. As there are likely infinitely many paths in Ω, this is a difficult equation to solve directly. Instead, we apply Monte Carlo integration to form an estimator as

$$\widehat{I_j} = \frac{1}{N}\sum_{i=1}^{N} \frac{f_j(\bar{x}_i,\lambda_i)}{p(\bar{x}_i,\lambda_i)}, \tag{2}$$

which converges towards the correct solution as $N \to \infty$. Here $p(\bar{x}_i,\lambda_i)$ describes a *probability density function* (PDF) for the sampling of combined path-wavelength pairs, decomposed as

$$p(\bar{x},\lambda) = p(\lambda)\cdot p(\bar{x} \mid \lambda). \tag{3}$$

For a uniform distribution, the convergence rate of this estimator is typically $\mathcal{O}(N^{-1/2})$. If a distribution is similar in shape to the integrand, variance may be reduced as samples are focused on places of interest — known as *importance sampling*. If a distribution differs significantly from the integrand, a slower convergence is likely.

The wavelength sampling distribution $p(\lambda)$ can be efficiently constructed as the product of a sensor response p_s and another distribution p_e, i.e. $p(\lambda) = p_s(\lambda)\cdot p_e(\lambda)$. The latter distribution is typically uniform or, preferably, proportional to emission in a scene. Evans and McCool [EM99] proposed selecting a random emitter in the scene to leverage for p_e. More recently, West et al. [WGGH20] used a mixture of a scene's emission spectra for their technique.

Multiple wavelength sampling Evans and McCool [EM99] first noted that Equation 2 evaluates a single wavelength per light transport path, and proposed to instead propagate wavelength clusters until wavelength-dependency occurs, at which point all wavelengths but one are discarded to prevent exponential path growth. This was extended by Radziszewski et al. [RBA09] to propagate multiple wavelengths along a single path in the case of non-specular dispersive scattering, and was later formalized by Wilkie et al. [WND*14] with *hero wavelength spectral sampling* (HWSS). The authors select a single hero wavelength λ_h for which they compute a light transport path. A set of C wavelengths is then stratified across the spectrum using a rotation function

$$\lambda_s = (\lambda_h - \lambda_{min} + \frac{s}{C}\bar{\lambda})\ mod\ \bar{\lambda} + \lambda_{min}. \tag{4}$$

where $\lambda_{min},\lambda_{max}$ are the bounds of the spectral range, and $\bar{\lambda} = \lambda_{max} - \lambda_{min}$. These wavelengths are measured across the same path, and results are combined using *multiple importance sampling* [Vea98] (MIS), leading to the following estimator:

$$\widehat{I_j} = \frac{1}{N}\sum_{i=1}^{N}\sum_{s=1}^{C} \frac{f_j(\bar{x}_i,\lambda_i^s)}{\sum_{k=1}^{C} p(\bar{x}_i,\lambda_i^k)}. \tag{5}$$

This concept was afterwards extended to handle fluorescent effects [MFW18]. In a slightly different approach, Petitjean et al. [PBE18] apply the seminal work of gradient-domain rendering [LKL*13; KMA*15] to spectral rendering. They do not evaluate secondary wavelengths for the same path, but instead generate and later reconnect additional subpaths, allowing them to estimate spectral image gradients. These are used to reduce variance in a later step.

More recently, West et al. [WGGH20] show in their work on CMIS that stratifying wavelengths as in Equation 4 is inefficient if $p(\lambda)$ is not uniform. As an alternative, they propose rotating their samples over the invertible *cumulative density function* (CDF) P_λ of this distribution, yielding the following rotation function:

$$\lambda_s = P_\lambda^{-1}((u + \frac{s}{C})\ mod\ 1)\cdot\bar{\lambda} + \lambda_{min}. \tag{6}$$

Here a uniformly distributed random variable u is instead stratified across a uniform distribution, and secondary wavelengths are then recovered through inversion transform sampling. As West et al. [WGGH20] use a mixture of emitter SPDs for $p_e(\lambda)$, their technique shows improvements especially for spiky illuminants.

3. Methodology

We reason that for spectral importance sampling, the distribution $p(\lambda)$ in Equation 3 should be optimally proportional to the spectral distribution of I_j, as then wavelengths — and by extension paths — with significant contribution are more densely sampled. Due to the complex nature of light transport, spectral renderers typically employ a predefined distribution for wavelength sampling. As mentioned, it is the product of sensor response p_s and a distribution p_e likely based on emission spectra. For example, a mixture of a scene's N_e emitter spectra $E_1,\ldots,E_{N_e}$ can be constructed as

$$p_e(\lambda) = \frac{1}{w_e}\sum_{i=1}^{n} E_i(\lambda) \quad : \quad w_e = \sum_{i=1}^{n}\sum_{j=1}^{\lambda} E_i(j) \tag{7}$$

where w_e serves as a normalization constant. Such a mixture distribution is suitable for scenes with few or similar emitters, where it is

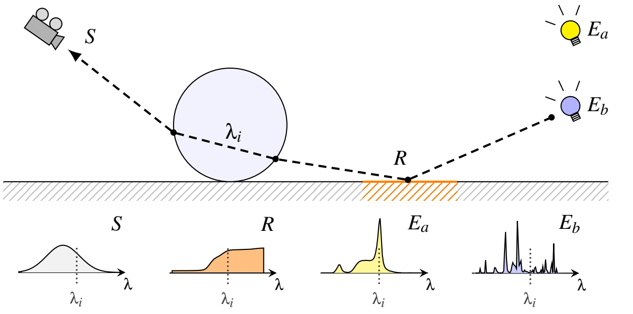

Figure 2: ***Wavelength sampling.*** *A light path passes through a dispersive medium and is evaluated for a wavelength λ_i, on which emitter E_a contributes greatly. As shown, optimal wavelength sampling is proportional to the sensor response curve S, absorption at surface R, and the emitter E_b to which the path connects.*

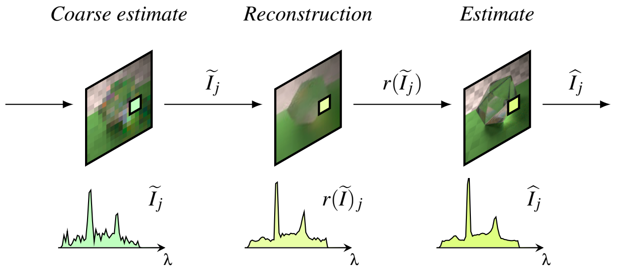

Figure 3: ***Method overview.*** *For every pixel j, we obtain a coarse but unbiased spectral-radiance estimate $\widetilde{I}_j$, wich we process in a reconstruction function r, and subsequently use as a wavelength sampling distribution, improving estimator efficiency.*

likely proportional to observed radiances. Unfortunately, this strategy can be suboptimal even in simple scenarios (Figure 2). With non-uniform reflectance and participating media, observed radiances are usually not proportional to emitted radiances, and may vary significantly on a per-pixel basis. Further, in the presence of multiple emitters with thin, non-overlapping spectral bands, a mixture distribution is likewise suboptimal as no single emitter is sampled efficiently.

We propose to instead generate a viable p_e on a per-pixel basis, by simply prepending additional render passes to a conventional light transport algorithm. We split such a pass into two stages (Figure 3). In the first stage, we obtain a coarse but unbiased estimate $\widetilde{I}$ of the entire image. We do not map this image into a color space, but instead store the produced spectral-radiance values. In the second stage, we use a reconstruction function r on this image to obtain a biased but relatively noise-free estimate. In any subsequent pass, we can then employ the filtered radiances in $r(\widetilde{I})$ as distributions for wavelength importance sampling. Instead of fitting a single spectral distribution to a scene, we essentially learn per-pixel distributions that are proportional to incident radiance. This expands Equation 2 to the following:

$$\widehat{I}_j = \frac{1}{N}\sum_{i=1}^{N} \frac{f_j(\bar{x}_i, \lambda_i)}{r(\widetilde{I}_j)(\lambda_i) \cdot p_s(\lambda_i) \cdot p(\bar{x}_i \mid \lambda_i)}, \tag{8}$$

where $r(\widetilde{I}_j)(\lambda) \cdot p_s(\lambda)$ produces a viable probability density for sampling λ based on the estimate now available in pixel j. We detail our approach in the following.

Coarse estimate In a first pass, we obtain the coarse estimate $\widetilde{I}$. This step is straightforward: it has to remain cheap as it directly impacts total runtime. As the estimate should capture all aspects of incident spectral radiance, we cannot rely on biased alternatives and employ a path tracer with simple cost-saving measures, inducing:

1. A separate sample rate $\widetilde{N} < N$, reducing accuracy.
2. A separate image resolution, scaling the original resolution by a factor $\alpha_s \leq 1$, increasing error at discontinuities.
3. Restriction to tracing only paths of interest fully. Light paths not encountering wavelength-dependent phenomena after a number of bounces are terminated early.

The reconstruction function will recover a spectral distribution that is adequate for our purposes. If necessary, said function will filter noise, perform resampling, and account for culled paths with a fallback distribution, such as Equation 7. We evaluate the impact of image scaling and sample rates in Section 5. One cost-saving measure we do not consider is a reduced spectral resolution. This may cause thin wavelength bands to be insufficiently represented for effective importance sampling. Further, it is often an implementation constant for vectorization and performance [Jak10], making it impractical to modify during runtime without significant re-engineering.

Reconstruction function The reconstruction function r processes the coarse image estimate $\widetilde{I}$. Due to its performance, we employ a joint bilateral filter [ED04; PSA*04] for fast edge-preserving filtering. It leverages a secondary *guide image* I' to mark discontinuities; we rely on secondary scene information such as direct depth, normals, and reflectance color. Conventional renderers can access these attributes easily and a single ray per pixel is sufficient for us. Discontinuities are marked by the difference in pixel values in the guide image; therefore similar pixels separated by some discontinuity can still be considered during filtering. The filter is defined as

$$\begin{aligned} r_{filt}(\widetilde{I}_j) &= \frac{1}{w_j}\sum_{k\in\Omega} G_{\sigma_s}(\|k-j\|)\, G_{\sigma_r}(\|I'_k - I'_j\|)\, \widetilde{I}_k \\ : \quad w_j &= \sum_{k\in\Omega} G_{\sigma_s}(\|k-j\|)\, G_{\sigma_r}(\|I'_k - I'_j\|), \end{aligned} \tag{9}$$

where Ω is a local image neighborhood of pixels around j, and G_{σ_r} and G_{σ_s} are range and spatial Gaussian filters with standard deviations σ_r and σ_s, respectively. The weight w_j is a normalization factor that ensures all weights sum to 1, even in a discrete filter. We can combine this function with a resampling step using joint bilateral upsampling [KCLU07], to perform a combined edge-aware resampling and filtering to a higher resolution.

While we considered more advanced filters such as NL-means [BCM05], we saw little improvement in quality at a considerable computational overhead. Further, while we considered more recent path tracing specific denoisers, these do not fit our method as they typically focus on trichromatic rendering, while we must retain high-resolution spectral distributions after filtering.

The filtered distributions should sufficiently cover the contributing wavelengths, as these might otherwise introduce bias when importance sampling. To avoid this issue, even for low sample rates, we employ defensive mixture sampling, adding an offset to our sampling distribution. Instead of a constant offset, we define a spectral distribution p_ε based on emitter spectra present in the scene:

$$\forall_\lambda \; p_\varepsilon(\lambda) = \begin{cases} w_\varepsilon & \text{if } \exists\, E : E(\lambda) \neq 0, \\ 0 & \text{else.} \end{cases} \tag{10}$$

Here, w_ε is a normalization weight ensuring a normalized distribution. The application of mixture sampling yields the defensive function $r_{def}(\widetilde{I}_j) = \varepsilon \cdot p_\varepsilon + (1-\varepsilon) \cdot \widetilde{I}_j$, where the choice of $\varepsilon \in [0,1]$ trades off potential benefits and detriments of the distributions in $\widetilde{I}$. Note that, if a collection of emitters contributed on all wavelengths uniformly, p_ε would become a constant offset.

Multiple pre-passes The described procedure lends itself to multiple passes. One pass serves as input to sample the next estimate, generating spectral distributions of continually improving quality. For K passes, we recursively define:

$$\widetilde{I}^k_j = \frac{1}{\widetilde{N}^k} \sum_{i=1}^{\widetilde{N}^k} \frac{f_j(\bar{x}_i, \lambda_i)}{p^k_j(\lambda_i) \cdot p_s(\lambda) \cdot p(\bar{x}_i \mid \lambda_i)}, \quad k \in [1, \ldots, K], \tag{11}$$

where $p^k_j(\lambda)$ in turn samples the $k-1^{\text{th}}$ pass as

$$p^k_j(\lambda) = \begin{cases} r(\widetilde{I}^{k-1}_j)(\lambda) & k > 1, \\ p_e(\lambda) & k = 1. \end{cases} \tag{12}$$

The first pass ($k = 1$) simply becomes Equation 2, falling back to a default distribution such as an emitter mixture (Equation 7). The image scale α^k_s and sample rate $\widetilde{N}^k$ of the coarse estimate now vary in subsequent passes. We analyze different configurations (passes, scaling, and sample rate) in Section 5.

Sample reuse Note that samples from earlier passes are not combined with samples in later passes. As each pass forms an independent estimator with a unique sampling distribution, an unbiased combination of different passes requires accurate knowledge of the variance of each estimator. While robust combinations of estimators have been explored in the context of path guiding [VHH*19], these introduce bias. A combined result of the different passes effectively eliminates our method's overhead, but only if the potential bias tradeoff is considered acceptable. In the interest of predictive spectral rendering, we do not include this combination.

Fallback mechanism A limitation of our method is the handling of near-uniform spectral distributions. Uniformly distributed incident radiance is essentially a product of uniform reflectance/transmittance and the best strategy remains to sample wavelengths either uniformly, or based on an emitter. As our estimate is always slightly noisy, it will never match such distributions perfectly. Consequently, we would expect reduced efficiency. To counteract this, we use the previously described multi-pass method to detect such cases in an earlier, cheaper pass, skipping the computation of all affected pixels in later passes.

We use a threshold on the *mean squared error* (MSE) between our (normalized) distribution, and a default distribution E such as an emitter mixture (Equation 7), to determine when to fall back. We establish threshold parameter h and define a threshold function

$$t_h(\widetilde{I}) = \frac{1}{\Lambda} \sum_{i=1}^{\Lambda} (\widetilde{I}[i] - E[i])^2 \leq h, \tag{13}$$

where we assume a discrete spectral representation with Λ bins, using $\widetilde{I}[i]$ and $E[i]$ to access the spectral entries. Integrating this function into Equation 12 yields:

$$p^k_j(\lambda) = \begin{cases} r(\widetilde{I}^{k-1})_j(\lambda) & k > 1 \;\wedge\; t_h(\widetilde{I}^{k-1}_j), \\ p(\lambda) & k = 1 \;\vee\; \neg\, t_h(\widetilde{I}^{k-1}_j). \end{cases} \tag{14}$$

Later passes are only produced for pixels with significantly different spectral distributions, hereby eliminating most of the overhead of applying our method when it is not needed.

Integration with multiple wavelength sampling Reusing a light transport path for multiple wavelengths as in HWSS [WND*14] remains an efficient technique for spectral noise reduction. West et al. [WGGH20] recently demonstrated the advantages of independent sample placement across spectral distributions, as opposed to a stratified one. We follow their approach, sampling a number of wavelengths warped according to our method's derived spectral distributions as per Equation 6. We construct an MIS estimator to combine C independently placed wavelengths as

$$\widehat{I}_j = \frac{1}{N} \sum_{i=1}^{N} \sum_{s=1}^{C} \frac{f_j(\bar{x}_i, \lambda^s_i) \cdot w(\bar{x}_i, \lambda^s_i)}{r(\widetilde{I}_j)(\lambda^s_i) \cdot p_s(\lambda) \cdot p(\bar{x}_i \mid \lambda^s_i)}, \tag{15}$$

where $w(\bar{x}, \lambda)$ denotes the MIS weight that accommodates for repeated sampling of $\bar{x}_i$ using the different wavelengths. By applying, for example, the *balance heuristic* [Vea98]

$$w(\bar{x}_i, \lambda^s_i) = \frac{p(\bar{x}_i \mid \lambda^s_i)}{\sum_{k=1}^{C} p(\bar{x}_i \mid \lambda^k_i)}, \tag{16}$$

we obtain the following estimator

$$\widehat{I}_j = \frac{1}{N} \sum_{i=1}^{N} \sum_{s=1}^{C} \frac{f_j(\bar{x}_i, \lambda^s_i)}{r(\widetilde{I}_j)(\lambda^s_i) \cdot p_s(\lambda) \cdot \sum_{k=1}^{C} p(\bar{x}_i \mid \lambda^s_i)}. \tag{17}$$

This is identical to the CMIS estimator described by West et al. [WGGH20] but, instead of a fixed distribution, leverages our per-pixel distribution.

4. Implementation

We implement our method within Mitsuba [Jak10], a research-oriented *C*++-based renderer. To simulate a simple form of wavelength dependency, we extend specular and non-specular dielectric BSDFs with Cauchy's equation, so that light dispersion can occur in materials such as glass. Mitsuba uses a discrete *binned* spectral representation, which we configure to use 64 equally-sized bins over a $360-830nm$ range (approximately $7nm$ per bin in the visible light spectrum), which is the default range used in Mitsuba and its color matching functions. We find that this suffices for accurately representing emitters with thin wavelength bands, which are common for fluorescent lights and other gas-discharge lamps.

Our method has two parts: a preprocessing part, which consists of earlier render passes with reconstruction passes in between, and

Name	Short	LSPDD idx.	Type	Color temp.
GE Candle	INC	2484	Inc.	2450K
Philips Candlelight	LED1	2471	LED1	2700K
Ledtech PAR20	LED2	2470	LED2	5828K
Globe Twister	CFL1	2488	CFL	4749K
ELume PAR30 LN Flood	CFL2	2627	CFL	4066K
Energystar Twister	CFL3	2479	CFL	2700K

Table 1: ***Emission spectra****. Name, LSPDD index [RA19], emitter type and color temperature.* Short names *are referenced in the text.*

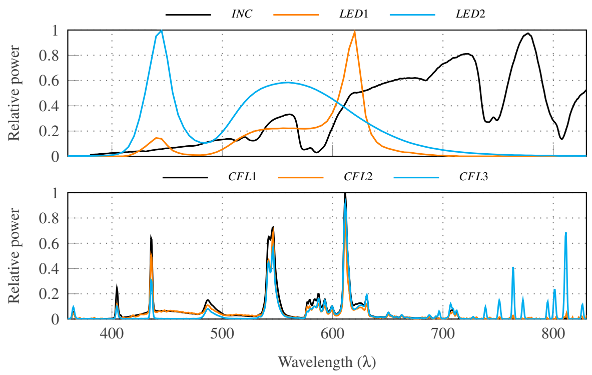

Figure 4: ***Emission spectra****. We show used emission spectra (Table 1) obtained from the LSPDD [RA19]. Note the significant differences between incandescent, LED and CFL emitters.*

a rendering part, which produces a final estimate. Given this distinction, we can prepend the preprocessing to clones of Mitsuba's unbiased path tracing and volumetric path tracing integrators. These support *next-event-estimation*, and we extend them to leverage HWSS [WND*14] and CMIS [WGGH20] for wavelength-dependent paths. For non-wavelength-dependent paths, all wavelengths are propagated. The only further modification to these integrators is the replacement of their respective wavelength sampling distributions with our distributions of choice. As a sensor response curve, we adapt the curve described in [RBA09], which is the same for all methods evaluated in Section 5.

5. Results

We evaluate our method on scenes with challenging combinations of spectral distributions; the full set of reflectance spectra from a *Macbeth Color Checker* [Bab19] and emission spectra covering common types of emitters, such as LEDs, incandescent bulbs, and fluorescent lights, from the *Lamp Spectral Power Distribution Database* (LSPDD) [RA19] under CC-Y-NC-ND 2.5 CA license, listed in Table 1 and displayed in Figure 4. We also use participating media parameters [NGD*06], which are readily available in Mitsuba [Jak10]. For each render, we produce RGB and full spectral-radiance data, the latter of which we use for error computations. References are produced with adequate samples ($N = 256k$ for smaller scenes and $N = 512k$ for larger scenes) with an unbiased unidirectional path tracer. We provide comparable error metrics as MSE. For a fair comparison, the measured runtimes of our method always include the preprocessing overhead.

5.1. Parameter Evaluation

To acquire suitable parameters, we define a baseline configuration and then vary specific parameters. To avoid overfitting on test scenes, we use a separate geometrically simple scene with spectra not applied in the rest of the paper. Derived parameters are kept constant for all further results. We manually fix parameters whose influence is minimal; the range component of the bilateral filter $\sigma_r = 0.015$ by visually determining that edge preservation is maintained, the amount of safe defensive mixture sampling with $\varepsilon = 0.05$ (lower values occasionally produce undersampled wavelengths, but have little influence on effectiveness), and the fallback threshold $h = 0.0002$ (hereby, it only triggers on distributions near-identical to Equation 7).

The sample rate $\widetilde{N}$ does affect convergence (Figure 5, left), and we render our test scene for increasing $\widetilde{N}$ while generating a single pre-pass at full image scale ($K = 1, a_s = 1, \sigma_s = 1$). The influence on convergence diminishes for larger values, indicating that a low sample rate of $\widetilde{N} = 128$ suffices for our test scene. Our method's overhead is evident: a doubling of $\widetilde{N}$ incurs an expected doubling of preprocessing time, implying the importance of a careful choice. We consider three configurations of multiple passes (Figure 5, right). The topmost configuration (image scaled) quarters the number of pixels in each earlier pass, while the middle configuration (sample scaled) quarters the sample rate instead. The bottom configuration (both scaled) halves both parameters in each earlier pass. Each configuration requires near-equal preprocessing times, barring minor differences in scheduling and filtering. For each configuration, we show the influence of different sample rates ($\widetilde{N}$) and spatial filtering (σ_s) over two passes ($K = 2$). Using ($\widetilde{N} = 128, \sigma_s = 1$) we then vary the number of passes. Evidently, an increased sample rate provides minor benefits at significant cost, so we retain $\widetilde{N} = 128$. We further select $\sigma_s = 1.75$ as a suitable spatial filter. Finally, while improvements from two or more passes are almost negligible, we select $K = 2$ as this allows us to leverage the fallback mechanism, reducing preprocessing overhead where a simpler sampling strategy suffices. Ultimately, differences between the three configurations are minimal, as such we select the sample scaled configuration, which remains the simplest approach.

5.2. Method Evaluation

We evaluate three wavelength sampling distributions for p_e: a uniform distribution (Unif.), the emitter mixture described in Equation 7 (Em.), and our per-pixel distribution (Ours), each multiplied by a sensor response p_s. We provide measured MSE over time for each method, and additionally show difference images for $N = 256$ spp. We evaluate both single wavelength sampling and multiple wavelength sampling using 4 wavelengths. Note that, for the latter, sampling a uniform distribution does not equate HWSS [WND*14]. Said method stratifies wavelengths across the full spectrum (Equation 4), which can be problematic for a wide spectral range with a limited sensor response. We instead stratify across the distribution (Equation 6) as demonstrated in CMIS [WGGH20] to optimally handle this scenario. For the emitter distribution, multiple wavelength sampling evaluates the current state-of-the-art technique developed by West et al. [WGGH20].

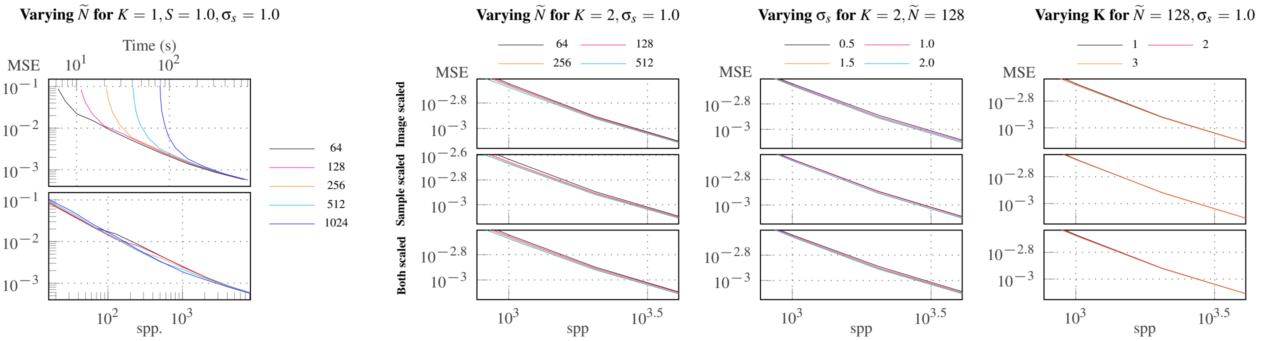

Figure 5: ***Parameter evaluation***. *We evaluate method parameters. We show the influence of a single pre-pass with varying sample rate on estimator convergence and runtime (left), and then show the influence of different parameters in conjunction with two or more passes (right).*

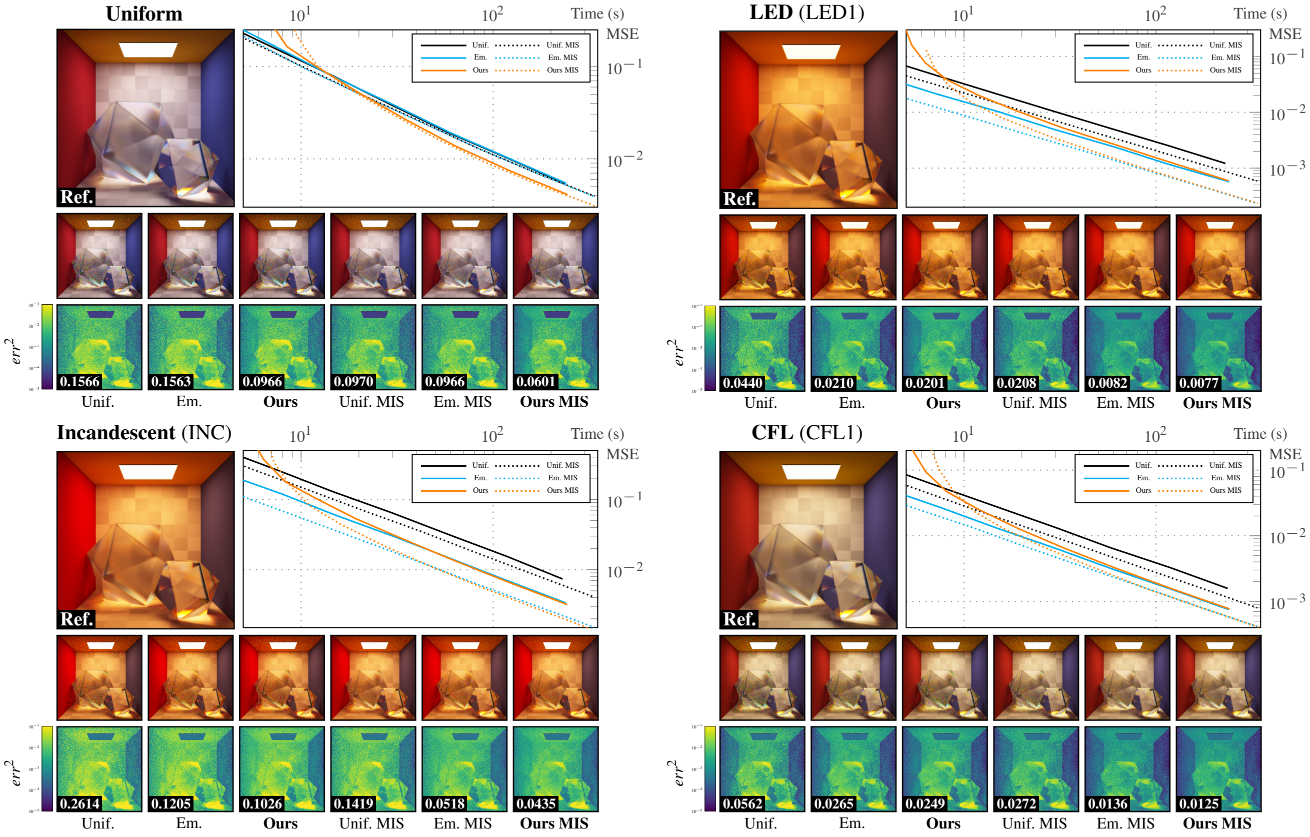

Figure 6: ***Single emitter results***. *We compare uniform sampling (*Unif.*), emitter sampling (*Em.*) and our method (*Ours.*) across four scenes with a single emitter whose emission is varied. Refer to Table 1 for the corresponding spectral distribution. In these scenes, emitter sampling is near-optimal. Our method matches emitter sampling in effectiveness, and outperforms it when emitter sampling becomes suboptimal.*

Single emitter scenes We first compare each method in four scenes (Figure 6: *Uniform, LED, Incandescent, CFL*) containing a single emitter whose spectral distribution is made to vary. Emitter sampling is near-optimal in these scenes, and we expect our method to match it in performance. As demonstrated, this is the case. Our method delivers lower error in all cases, but is offset by an increase in runtime, which can be attributed to its preprocessing overhead. As the estimator converges, this overhead comparatively diminishes relative to the total runtime. Interestingly, a notable improvement is visible in the *Uniform* scene, where our method affects areas with heavy absorption. Emitter sampling necessarily becomes uniform sampling, losing most of its efficiency. Even with multiple wavelength sampling - which typically increases runtime costs - this difference remains significant.

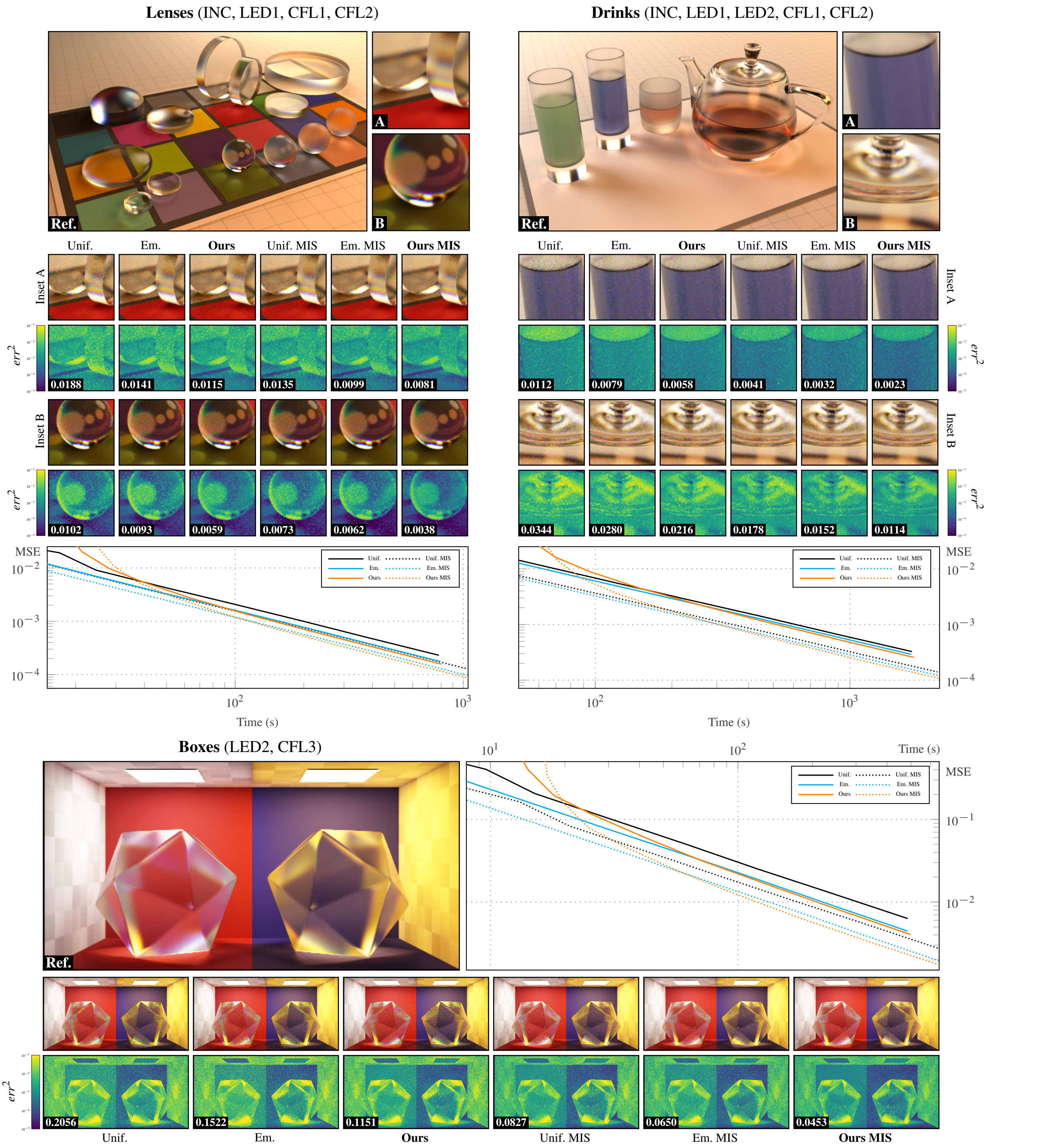

Figure 7: ***Multiple emitter results****. We compare uniform sampling (*Unif.*), emitter sampling (*Em.*) and our method (*Ours*) across three scenes using a varied number of emitter spectra (listed near scene titles). Refer to Table 1 for the corresponding spectral distributions. Plotted graphs are acquired over the whole image. Listed errors are for highlighted image insets. As demonstrated, our method is generally effective, but shows the strongest improvement in areas where there is significant absorption or chromatic dispersion.*

Multiple emitter scenes We next compare each method in three scenes (Figure 7: *Lenses, Drinks, Boxes*) of varying complexity, each containing multiple different emitters. In the *Lenses* scene, we highlight two insets (A, B, $N = 256$ spp.) where highly dispersive phenomena are visible, demonstrating that our method provides a suitable sampling distribution for fine details. Error decrease varies strongly across the image, as in many places methods either do not necessitate wavelength sampling, or our method's fallback mechanism triggers. For $N = 1024$ spp, we see an overall decrease in error compared to emitter sampling by 19.2%. The error decrease is small in some areas (inset A: 9.4%) but comparatively strong in others (inset B: 33.9%). We then demonstrate handling of participating media in the *Drinks scene*. Areas undergoing heavy absorption show strong improvements (inset A: 21.1%), though complex highlights caused by different emitters are also handled (inset B, 20.1%). Overall, for $N = 1024$ spp., this error decrease incurs a 17.6% time overhead. This overhead diminishes to 3.7% for $N = 4096$ spp., while the error decrease is only slightly reduced (inset A: 18.3%, inset B: 15.1%). Finally, we construct an optimized but challenging scenario in the *Boxes* scene: only 25% of light from either emitter can reach the other box due to a blocker in the center of the scene, and said light likely undergoes some absorption. As shown, our method reduces error especially in the absorbing regions, where an emitter sampling distribution is suboptimal. For $N = 1024$ spp., we see an overall error decrease of 22.9% compared to emitter sampling, and 40.2% compared to uniform sampling.

Across all tests, emitter sampling remains comparatively performant. While for simpler scenarios our method is on par, the difference becomes pronounced when there is heavy absorption. Uniform sampling is easily outperformed by either method in all test cases, demonstrating the benefits of a proper sampling distribution.

6. Conclusion

We have developed a multi-pass method for accelerated spectral rendering, which is a simple method that counteracts wavelength sampling problems in spectral light transport. We demonstrated that investing compute time to derive an approximate spectral-radiance distribution per pixel can improve convergence and reduce variance when using this distribution for importance sampling. Our method handles complex and non-uniform spectral distributions, which are common in real-world emission and reflectance spectra. Given the benefits for difficult scenarios, we hope that it will contribute to making the use of realistic spectral data more common.

In the future, we hope that integration with other spectral effects, such as fluorescence, becomes possible. This may be possible by storing unshifted wavelengths in the pre-pass.

7. Acknowledgements

We would like to thank Wenzel Jakob and his collaborators for making the Mitsuba Renderer [Jak10] publicly available. We would additionally like to thank Johanne Roby, Martin Aubé et al. for their work on the Lamp Spectral Distribution Database [RA19].

References

[Bab19] BABELCOLOR COMPANY. *The Colorchecker Pages*. http://www.babelcolor.com/colorchecker.htm. 2019 5.

[BCM05] BUADES, A., COLL, B., and MOREL, J. "A non-local algorithm for image denoising". *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Vol. 2. 2005, 60–65 vol. 2 3.

[Bor91] BORGES, C. F. "Trichromatic approximation for computer graphics illumination models". *ACM SIGGRAPH Computer Graphics* 25.4 (July 1991), 101–104. DOI: 10.1145/127719.122729 1.

[ED04] EISEMANN, E. and DURAND, F. "Flash photography enhancement via intrinsic relighting". *ACM Transactions on Graphics* 23.3 (Aug. 2004), 673. DOI: 10.1145/1015706.1015778 3.

[EM99] EVANS, G. F. and MCCOOL, M. D. "Stratified Wavelength Clusters for Efficient Spectral Monte Carlo Rendering". *Graphics Interface*. Morgan Kaufmann Publishers Inc., 1999, 42–49 1, 2.

[ICG86] IMMEL, D. S., COHEN, M. F., and GREENBERG, D. P. "A radiosity method for non-diffuse environments". *ACM SIGGRAPH Computer Graphics* 20.4 (Aug. 1986), 133–142. DOI: 10.1145/15886.15901 2.

[Jak10] JAKOB, W. *Mitsuba - Physically Based Renderer*. Accessed 2019-10-01 at https://www.mitsuba-renderer.org. 2010 3–5, 8.

[Kaj86] KAJIYA, J. T. "The Rendering Equation". *SIGGRAPH Comput. Graph.* 20.4 (Aug. 1986), 143–150. DOI: 10.1145/15886.15902 2.

[KCLU07] KOPF, J., COHEN, M. F., LISCHINSKI, D., and UYTTENDAELE, M. "Joint bilateral upsampling". *ACM Transactions on Graphics* 26.3 (July 2007), 96. DOI: 10.1145/1276377.1276497 3.

[KMA*15] KETTUNEN, M., MANZI, M., AITTALA, M., et al. "Gradient-domain path tracing". *ACM Transactions on Graphics* 34.4 (July 2015), 1–13. DOI: 10.1145/2766997 2.

[LKL*13] LEHTINEN, JAAKKO, KARRAS, TERO, LAINE, SAMULI, et al. "Gradient-domain metropolis light transport". *ACM Transactions on Graphics* 32.4 (Aug. 2013), 1. DOI: 10.1145/2461912.2461943 2.

[MFW18] MOJZÍK, M., FICHET, A., and WILKIE, A. "Handling Fluorescence in a Uni-directional Spectral Path Tracer". *Computer Graphics Forum* 37.4 (2018), 77–94. DOI: 10.1111/cgf.13477 2.

[NGD*06] NARASIMHAN, S. G., GUPTA, M., DONNER, C., et al. "Acquiring scattering properties of participating media by dilution". *ACM Transactions on Graphics* 25.3 (July 2006), 1003–1012. DOI: 10.1145/1141911.1141986 5.

[PBE18] PETITJEAN, V., BAUSZAT, P., and EISEMANN, E. "Spectral Gradient Sampling for Path Tracing". *Computer Graphics Forum* 37.4 (July 2018), 45–53. DOI: 10.1111/cgf.13474 1, 2.

[PSA*04] PETSCHNIGG, G., SZELISKI, R., AGRAWALA, M., et al. "Digital photography with flash and no-flash image pairs". *ACM Transactions on Graphics* 23.3 (Aug. 2004), 664. DOI: 10.1145/1015706.1015777 3.

[RA19] ROBY, J. and AUBÉ, M. *Lamp Spectral Power Distribution Database (LSPDD)*. Accessed 2021-06-03 at https://lspdd.org/app/fr/home. 2019 5, 8.

[RBA09] RADZISZEWSKI, M., BORYCZKO, K., and ALDA, W. "An Improved Technique for Full Spectral Rendering". *Journal of WSCG* 17 (2009), 9–16 1, 2, 5.

[Vea98] VEACH, E. "Robust Monte Carlo Methods for Light Transport Simulation". PhD thesis. Stanford University, 1998 2, 4.

[VHH*19] VORBA, J., HANIKA, J., HERHOLZ, S., et al. "Path Guiding in Production". *ACM SIGGRAPH 2019 Courses*. ACM, 2019, 18:1–18:77. DOI: 10.1145/3305366.3328091 4.

[WGGH20] WEST, R., GEORGIEV, I., GRUSON, A., and HACHISUKA, T. "Continuous Multiple Importance Sampling". *ACM Trans. Graph.* 39.4 (July 2020). DOI: 10.1145/3386569.3392436 1, 2, 4, 5.

[WND*14] WILKIE, A., NAWAZ, S., DROSKE, M., et al. "Hero Wavelength Spectral Sampling". *Computer Graphics Forum* 33.4 (July 2014), 123–131. DOI: 10.1111/cgf.12419 1, 2, 4, 5.