View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Elsevier - Publisher Connector

Available online at www.sciencedirect.com

ScienceDirect

Solar Energy 114 (2015) 327–336

SOLAR ENERGY

www.elsevier.com/locate/solener

Progressive photon mapping for daylight redirecting components

R. Schregle *, L. Grobe, S. Wittkopf

Lucerne University of Applied Sciences and Arts, Competence Centre Envelopes and Solar Energy, Switzerland

Received 12 May 2014; received in revised form 12 December 2014; accepted 29 January 2015
Available online 27 February 2015

Communicated by: Associate Editor Jean-Louis Scartezzini

Abstract

Daylight redirecting components (DRCs) are characterised by complex transmissive and reflective behaviour that is difficult to predict accurately largely due to their highly directional scattering, and the caustics this produces. This paper examines the application of progressive photon mapping as a state of the art forward raytracing technique to efficiently simulate the behaviour of such DRCs, and how this approach can support architects in assessing their performance.

Progressive photon mapping is an iterative variant of static photon mapping that effects noise reduction through accumulation of results, as well as a reduction in bias inherent to all density estimation methods by reducing the associated bandwidth at a predetermined rate. This not only results in simplified parametrisation for the user, but also provides a preview of the progressively refined simulation, thus making the tool accessible to non-experts as well.

We demonstrate the effectiveness of this technique with an implementation based on the RADIANCE photon mapping extension and a case study involving retroreflecting prismatic blinds as a representative DRC.
© 2015 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

Keywords: Daylight simulation; Light redirection; Caustics; Raytracing; Physically based rendering; Monte Carlo

1. Introduction

The accurate simulation of daylight redirecting components (DRCs) is essential in assessing their performance and predicting their energy saving potential through daylight autonomy. Raytracing techniques have proven to be particularly expedient in this application as they accurately model the light transport within the components (assuming an accurate representation of material properties) and how it propagates in a typical office environment.

Although light transport along a ray is inherently bidirectional (reversing the direction of light propagation does not invalidate the model), there is a distinction between backward and forward raytracers; the former emit rays at the view or measurement point(s), whereas the latter emit rays from the light sources. Forward raytracing is particularly effective at modelling highly specular DRCs with strong redirection to produce concentrated highlights (-caustics), which can compromise an office occupant's visual comfort.

Photon mapping (Jensen, 2001) is a forward raytracing technique which supplements a standard backward raytracer, resulting in bidirectional light transport. The technique mimics light particle transport by recording indirect hitpoints along with their associated energy, and uses density estimation to reconstruct the resulting irradiance on the surfaces.

The forward raytracing solution presented in this paper is based on a photon mapping extension to the RADIANCE

* Corresponding author at: Hochschule Luzern, CC Envelopes and Solar Energy (EASE), Technikumstr. 21, CH-6048 Horw, Switzerland. Tel.: +41 41 349 36 26.
E-mail address: roland.schregle@hslu.ch (R. Schregle).

http://dx.doi.org/10.1016/j.solener.2015.01.041
0038-092X/© 2015 The Authors. Published by Elsevier Ltd.
This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

rendering system originally developed by the author (Schregle, 2004). It extends the RADIANCE backward raytracing core (Ward, 1994) with a forward raytracer for bidirectional light transport as described above.

The standard photon mapping approach has since been superseded by recent developments in the computer graphics community; progressive photon mapping is now the state of the art forward raytracing approach, which overcomes a number of issues with the original implementation that improve its usability for non-experts, notably in the context of daylight simulation.

2. Previous work

A number of publications have documented raytracing simulations applied to a broad spectrum of DRCs, notably those with strong redirection for which raytracing is best suited.

de Boer (2006) presented a new method for modelling DRCs by representing the light transmitted though the system as a luminous intensity distribution obtained with raytracing, effectively presaging the `genBSDF` solution now bundled with RADIANCE. The results were validated with RADIANCE using measured BRDFs with wavelet based data compression. In his introduction, de Boer points out the necessity of supplementing existing backward raytracers with a forward raytracing pass for accurate simulation of DRCs.

Wittkopf et al. (2010) simulated light pipes and ducts fitted with different collector types using a commercial forward raytracer (Photopia) to obtain luminous intensity distributions. The results were then used to characterise the systems based on transmitted flux as performance criterion. Such DRCs could not be simulated with comparable accuracy and computation time using a backward raytracer due to excessive noise.

Klammt et al. (2012) simulated microstructured light redirecting devices using 2D raytracing; a comparison of the results with measurements indicated good agreement aside from deviations introduced by manufacturing tolerances, which are amplified by specular redirection.

A hybrid simulation using raytracing and radiosity was used by Chan and Tzempelikos (2012) to assess glare from specular venetian blinds in various configurations. Specular light transport is raytraced, while diffuse transport (from the underside of the blinds and room surfaces) is obtained from a radiosity solution. As the latter disregards all specular components, simulations using only radiosity revealed significant errors of up to 40% compared to the hybrid approach. Chan and Tzempelikos also validated their results against simulations with RADIANCE.

More recently, Appelfeld and Svendsen (2013) characterised glare and energy savings for light redirecting glass shading systems using RADIANCE's 3-phase method for annual daylight utilisation.

McNeil et al. (2013) described the recently developed `genBSDF` tool from the RADIANCE suite to obtain bidirectional scattering distribution functions (BSDFs) from fenestration systems and DRCs using raytracing. The resulting data was validated against analytically derived solutions for trivial cases, and against a commercial raytracer and goniometric measurements for more complex cases such as specular blinds and microperforated film.

There are few documented cases of photon mapping being used as forward raytracer in daylight simulation. Photon mapping is particularly efficient at simulating caustics, albeit subject to a bias/noise tradeoff (Schregle, 2003). Validated results of the photon mapping extension to RADIANCE were documented by Schregle and Wienold (2004). The simulation tool outlined in this paper is based on this software.

A more recent application of the RADIANCE photon map was documented by Su et al. (2012), who used the tool to evaluate the optical performance of lens-walled compound parabolic concentrators. In their work, they compared the results with those obtained from the Photopia forward raytracer and theoretical estimates; in both cases the deviations were within 5%. Su astutely noted that some deviations were probably attributed to the local bias inherent in the photon map's density estimates.

Progressive photon mapping was first proposed by Hachisuka et al. (2008) as an iterative extension of the standard static photon mapping approach as implemented in the RADIANCE extension. It combines multiple smaller photon maps to approximate a much larger one which may not fit into memory using the traditional approach. Through iteration, the process mitigates the noise inherent in Monte Carlo raytracing by combining successive results and averaging them. At the same time, the density estimate bandwidth[1] (radius or number of nearest photons) is gradually reduced to mitigate bias. As Hachisuka points out, the accumulated density estimates converge to an unbiased solution in the limit.

An alternative interpretation of progressive photon mapping was presented by Knaus and Zwicker (2011), who developed a statistical model for the variance and bias from photon density estimates to study their asymptotic behaviour as more photons are generated and the bandwidth is reduced. The approach is considerably simpler than Hachisuka's as there is no need to maintain local statistics from previously generated photon maps, and the iterations are independent and can thus be performed in parallel; this is leveraged in our implementation, which draws heavily on Knaus and Zwicker's work.

[1] Bandwidth describes the support, or area of influence, of a filter used to weight the photons retrieved from the photon map during a nearest neighbour lookup on a surface (Jensen, 2001). The resulting irradiance is proportional to the photon density, and the bandwidth is defined by the distance (radius) to the furthest photon found. In this paper, we generalise the term to describe either the radius or the number of nearest neighbours for a density estimate, depending on the implementation.

To our knowledge there are currently no documented instances of progressive photon mapping applied to daylight simulation. For reasons we will elaborate on, we consider it an evolution from the previous photon mapping extension to RADIANCE that will benefit architects and lighting engineers alike in terms of ease of use and efficient workflow.

3. Background

3.1. Retroreflecting prismatic blinds

In this paper, we draw upon retroreflecting prismatic blinds (Köster, 2004) as a representative DRC for exposition of our photon mapping algorithm. These are characterised by retroreflection of direct sunlight at high incident angles, and redirection of indirect sky light towards the interior (see Fig. 1).

Retroreflection is effected for light incident at high angles by the specular prismatic profile on the upper side of each lamella, which effectively constitutes a Fresnel mirror whose focal point lies just outside the fenestration. At the same time, light incident from low angles is reflected towards the diffuse underside of the lamella immediately above, whence it is scattered into the interior.

The prismatic structure therefore makes the blinds angularly selective, and in practice they require minor adjustment of the inclination angle. This behaviour is difficult to simulate accurately with backward raytracing due to the inability to predict ray directions which contribute to a caustic, making it an ideal case study for forward ray tracing, and therefore photon mapping.

3.2. Current limitations of the RADIANCE photon map

Standard (static) photon mapping as implemented in the RADIANCE photon map has a number of fundamental limitations which pose a challenge particularly to novice users:

3.2.1. Memory constraints

There is obviously a limit to how many photons can be generated and stored, either due to limited physical memory (in which case excessive paging to/from disk rapidly and severely degrades performance) or an OS-imposed soft limit. With extreme photon map sizes, the `mkpmap` forward raytracer and photon map generator will typically run out of memory during the forward pass after a substantial runtime.

3.2.2. Fixed photon map size

A user must decide *a priori* how many photons to store, often to realise the resulting quality is insufficient, requiring a new run of `mkpmap`. Inexperienced users will no doubt be frustrated by this need to "commit" themselves to a fixed photon map size, as it depends on a number of factors, including the scene extent, the geometry, the material properties, and the light sources. Even experienced users must exercise good judgement here.

3.2.3. Noise vs. bias

As a characteristic of Monte Carlo simulations, noise generally results from insufficient sampling; in the case of the photon map, this implies too few photons in the photon map overall and/or for the density estimate due to a low bandwidth. Setting the bandwidth too high will reduce the noise but introduce blurring, which constitutes a systematic error or *bias* (see Fig. 2). While noise is visually objectionable, bias is equally problematic as it distorts the local energy distribution represented by the photons collected in a nearest neighbour lookup (Schregle, 2003); caustics, in particular, will be diminished in intensity, resulting in compromised numeric accuracy.

3.3. Fundamentals of progressive photon mapping

Progressive photon mapping addresses the above mentioned practical issues by leveraging the following insights into the nature of photon mapping:

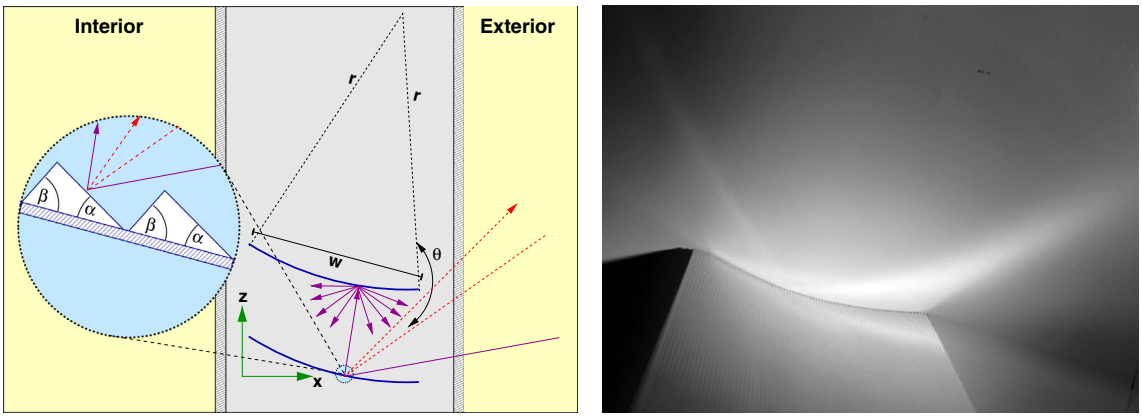


Fig. 1. Left: principle of operation of retroreflecting blinds with prismatic profile, patented by Helmut Köster (Köster, 2005, 2012). Retroreflection occurs for light incident from high elevation (red, dashed), while light incident from low elevation (magenta) is diffusely scattered off the lamella underside immediately above. indirectly illuminating the interior. Right: photograph of sample lamella with caustic resulting from retroreflection of light incident from right.

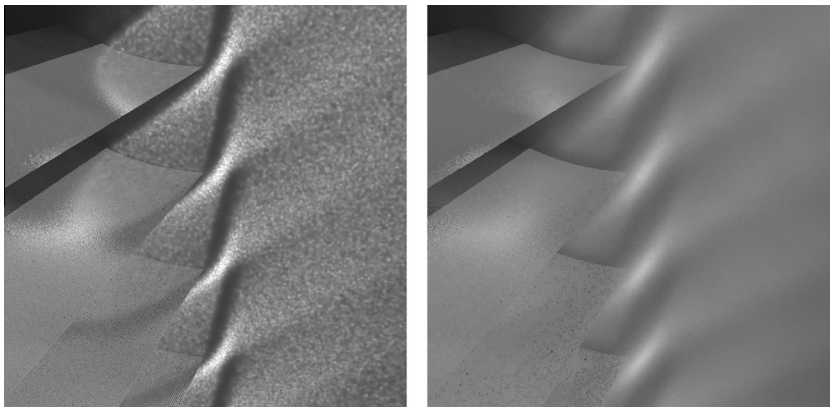


Fig. 2. Caustics from prismatic blinds rendered with the RADIANCE photon map using a bandwidth of 20 (left) and 2000 (right) photons per nearest neighbour lookup. The low bandwidth gives rise to noise but preserves the high frequency caustics, while the high bandwidth suppresses the noise at the expense of bias, resulting in blurred caustics with noticeably lower intensity.

3.3.1. Combining photon maps

The notion of combining density estimates from several smaller photon maps was first proposed by Christensen et al. (2004) to handle very complex photon maps which won't fit into memory at once. Assuming uniform density, a density estimate with a large bandwidth from one large photon map corresponds to the accumulated density estimates from several smaller constituent photon maps with proportionally reduced bandwidths (see Fig. 3). Thus a large photon map can be effectively broken up into smaller ones which can be handled independently within a reduced memory footprint.

3.3.2. Noise reduction through accumulation

By accumulating and averaging the photon densities of a number of smaller photon maps, we can effectively increase the number of samples, thus lowering the noise independently of the bandwidth. Crucial to this is the generation of disparate photon distributions by individually seeding the random number generator for each forward pass, as identical photon distributions will effect no reduction in noise whatsoever.

3.3.3. Bias reduction through bandwidth reduction

By reducing the bandwidth over a number of accumulated density estimates, the bias is reduced, while at the same time the noise drops through accumulation.

The essence of progressive photon mapping lies in iteratively generating a series of photon maps and accumulating density estimates from these using a progressively smaller bandwidth. A single parameter $\alpha \in (0,1)$ governs the relative reduction of variance and bias for each iteration. From these expected errors, Knaus and Zwicker's statistical model (Knaus and Zwicker, 2011) predicts the (likewise gradually reduced) search radius r_i for a density estimate at iteration number i:

$$\frac{r_{i+1}^2}{r_i^2} = \frac{\sigma^2[\epsilon_i]}{\sigma^2[\epsilon_{i+1}]} = \frac{i+\alpha}{i+1}, \tag{1}$$

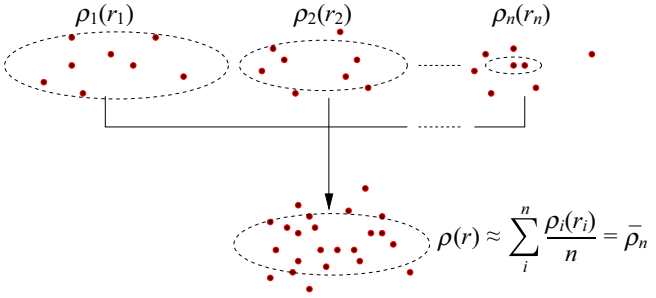


Fig. 3. Density estimates $\rho_1 \dots \rho_n$ from several smaller photon maps can be combined; their average $\bar{\rho}_n$ approximates a density estimate with a proportionally enlarged radius drawn from one large photon map containing the photons from the constituent maps.

where $\sigma^2[\epsilon_i]$ is the theoretical variance of the average density estimation error ϵ_i at iteration i. Allowing the variance to increase by the factor $(i+1)/(i+\alpha)$ implies a reduction in the radius r_i (and therefore in the bias), yielding the closed form equation:

$$r_i^2 = \frac{r_1^2}{i}\left(\prod_{k=1}^{i-1}\frac{k+\alpha}{k}\right). \tag{2}$$

This function requires an initial search radius r_1 for the first iteration, which is critical in that it affects the convergence rate (Kaplanyan and Dachsbacher, 2013). Since the RADIANCE photon map uses nearest neighbour lookups for photon density estimates, and the number of nearest photons k_i is proportional to the circular area πr_i^2 they occupy, we can simply replace the squared radius in the formulae above with k_i, and our initial squared radius r_1^2 is then an initial nearest neighbour count k_1. Defining bandwidth in this way has the advantage of automatically adjusting the radius to local density variations.

The dependence of each iteration solely on the initial bandwidth implies inherent parallelism which can be easily leveraged with commodity multi-core CPUs. Furthermore, it requires no modification to an existing photon mapping implementation beyond the ability to generate different

photon distributions for each iteration, effectively treating it as a "black box".

4. Overview of our implementation

Our approach to progressive photon mapping uses the existing RADIANCE photon map implementation as a self-contained module controlled by a script implementing the bandwidth reduction and an image based accumulation of density estimates. A summary of our method is shown in Algorithm 1.

Algorithm 1. Pseudocode for progressive photon mapping

procedure PROGPMAP($N_p, k_1, \alpha, \bar{\epsilon}_{max}, i_{max}, n_{proc}$)
  $i = 1$ ▷ Init iteration counter and bandwidth
  $k = k_1$
  repeat
    for all $1 \ldots n_{proc}$ **do**
      procedure SAMPTHREAD ▷ Launch sampling thread
        $PM_i \leftarrow$ mkpmap($N_p, i, \ldots$) ▷ Generate N_p photons with seed i
        $\rho_i \leftarrow$ rpict($PM_i, k, \ldots$) ▷ Render density estimates with bandwidth k
      end procedure
      if $k > k_{min}$ **then** ▷ Update bandwidth for next iteration
        $k \leftarrow k(i + \alpha)/(i + 1)$
      end if
      $i \leftarrow i + 1$
    end for
    $\bar{\rho}_i \leftarrow$ Combine($\rho_1, \ldots, \rho_i$) ▷ Combine density estimates
    $\bar{\epsilon}_i \leftarrow \|\bar{\rho}_i - \bar{\rho}_{i-1}\|$ ▷ Deviation of combined density estimate from previous
  until $\bar{\epsilon}_i < \bar{\epsilon}_{max}$ or $i > i_{max}$ or interrupt
  WriteImg($\bar{\rho}_i$) ▷ Save final combined density estimates to file
end procedure

Each iteration generates a photon map via `mkpmap` with an individual seed for the random number generator in order to avoid regenerating the same photon distribution. The generated photon map is then visualised with the current density estimate bandwidth using `rpict`. Iterations can be parallelised into a number of concurrent threads to accelerate convergence. The generated images are then accumulated in another concurrent thread to merge the density estimates from the constituent photon maps, and a preview is displayed using a simple linear tone-mapping with gamma correction.

Given a starting bandwidth of k_1 photons, the bandwidth is reduced according to Eq. (1) at each iteration. In addition, we clamp the bandwidth to a minimum k_{min} to avoid excessive noise from outliers in very dense regions. We have obtained good results with $k_{min} = 2$ photons.

The convergence criteria specified by the user include a threshold for the deviation between consecutive combined images, since this error drops on average as iteration progresses. In case the error plateaus due to excessive variance introduced by bandwidth reduction, a hard limit is imposed by a maximum iteration count. In addition, the user can prematurely terminate the progression by a keypress once (s) he is satisfied with the results.

The static RADIANCE photon map is usually used with a low resolution global photon map visualised indirectly via an ambient bounce (often referred to as *final gather* (Christensen, 1999)), and a dedicated high resolution caustic photon map accounting exclusively for specular light transport, which is visualised directly. In our progressive photon mapping approach we instead directly visualise the global photon map without generating a caustic photon map, as the former already includes caustics, albeit at lower resolution. This avoids the additional expense of the ambient component's stratified daughter rays, which would be significant if performed for every iteration. While this results in a noisier and coarser representation of global illumination per iteration, the effect is mitigated by accumulation, and at considerably lower cost than with final gather.

We have implemented the progressive RADIANCE photon map as a proof of concept in a Perl script using the Perl Data Language (2014) to accumulate the rendered images and evaluate the deviation using matrix operations. PDL is considerably more flexible and efficient at manipulating large datasets than using Perl arrays, and includes modules for data file import/export, bitmap display, and multi-threading with shared data.

4.1. Combining density estimates

4.1.1. Average

A straightforward way to combine the renderings of the photon density estimates is to simply average them as shown in Fig. 3. While this is fast and simple to implement, it is very susceptible to noise from outlying pixels, especially due to numerical instabilities as the density estimate bandwidth gets very low. More sophisticated algorithms perform selective averaging, resulting in much stronger noise rejection.

4.1.2. Sigma clip

We investigated *sigma clip*, a popular image processing technique in astronomy (Kennedy, 2012), as an alternative to nonselective averaging. Sigma clip rejects those pixels from contributing to the average whose deviation from the median exceeds $s\sigma$, where σ is the per-pixel standard deviation and $s > 0$ is a user specified factor.

One drawback of sigma clip is that it will omit all pixels which fall outside the deviation tolerance from the mean, potentially resulting in no pixels contributing to the average and leaving gaps in combined images. One simple solution included in our study was to draw on the single pixel with the lowest deviation (i.e. closest to the mean). While this leads to visible discontinuities in the combined images it is considerably less disturbing than outright omissions. In general, this fallback is only noticeable in the early stages of iteration before the density estimates stabilise.

This sophistication comes at a price as it does require maintaining a stack of images in memory, since recomputing the mean and standard deviation for each iteration requires *all* images. Not surprisingly, the computational load and memory consumption grows with each iteration, severely limiting the resolution of the renderings and number of iterations.

While the results with sigma clip were clearly superior to those obtained through nonselective averaging, the overhead made it impractical to the degree that it was only usable at very low resolutions and/or few iterations.

4.1.3. Sigma-weighted average

A practical alternative to sigma clip which avoids the outright rejection of pixels is to weight them according to the inverse of their deviation σ from the last averaged image[2]. The implementation only requires storing the weighted sum of all pixels and the sum of their weights, such that the weights can be renormalised and the average updated at every iteration.

Each pixel in the image ρ_i at iteration i is weighted by $1/(s\sigma_i + 1)$, where $\sigma_i = \|\rho_i - \bar{\rho}_{i-1}\|$ is the pixel's deviation from the last average, $\bar{\rho}_{i-1}$, and $s \geqslant 0$ is a user specified penalty factor which controls the average's deviation tolerance; $s = 0$ is equivalent to a nonselective average, and higher values "penalise" the pixel by lowering its weight in accordance with σ_i. The new average $\bar{\rho}_i$ then becomes:

$$\bar{\rho}_i = \frac{\sum_{j=1}^{i} w_j \rho_j}{\sum_{j=1}^{i} w_j}, \qquad w_j = \frac{1}{s\sigma_j + 1}, \; \sigma_j = \|\rho_j - \bar{\rho}_{j-1}\|, \; s \geqslant 0. \tag{3}$$

Note that only the sums in the numerator and denominator (weight normalisation) are necessary for updating the average on every iteration. These are stored on a per-pixel basis as matrices and results in a far more efficient implementation than sigma clip, yielding images of comparable quality. Also note that $\bar{\sigma}_i = \|\bar{\rho}_i - \bar{\rho}_{i-1}\|$ can then serve as a convergence metric.

The user has the ability to tailour the average's tolerance to deviations through the penalty factor s, but setting this too high can mitigate the contributions from newly generated images to the extent that the average hardly changes as iteration progresses. This in turn lowers $\bar{\sigma}_i$ and triggers premature convergence, resulting in an average characterised by noise from too few iterations and bias as bandwidth reduction has little effect. Good results were achieved with $s = 0.1$ as default value.

We have found sigma-weighted averaging to be a good alternative to sigma clip when combining images. It is simpler to implement, has a much more compact (and above all, bounded) memory footprint, and exhibits similar noise rejection characteristics at a computational load marginally higher than nonselective averaging.

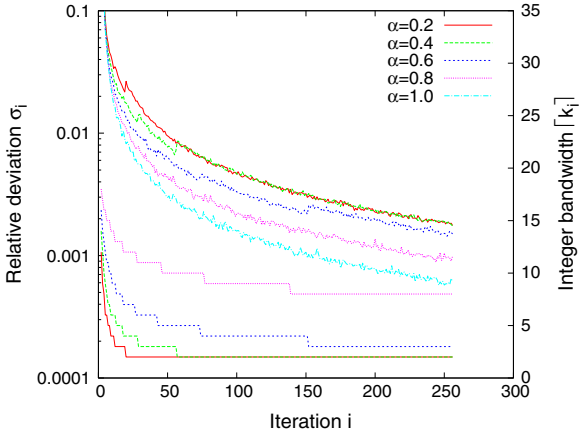
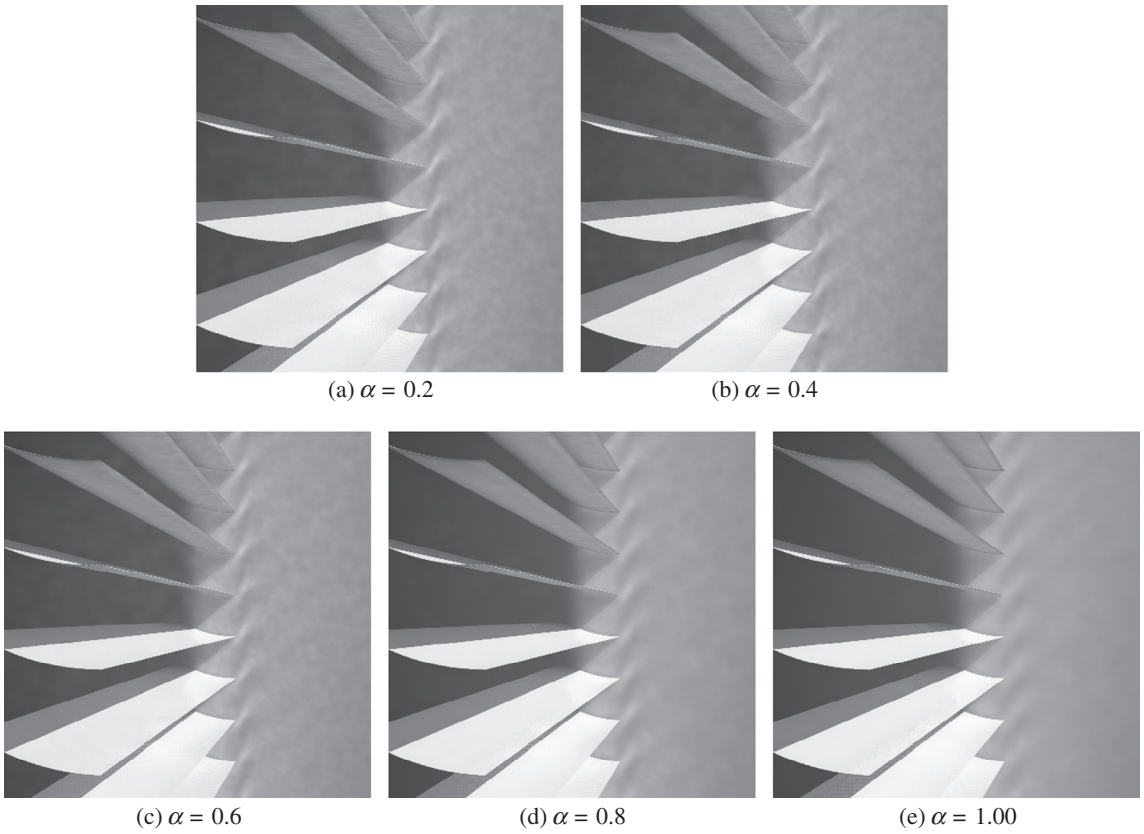


Fig. 4. Relative deviation between successive accumulated renderings of prismatic blinds averaged over all pixels (left axis, upper plots). The corresponding integer bandwidth is shown as step function (right axis, lower plots). Higher errors dominate for low values of α due to an increase in noise as the bandwidth is rapidly reduced. The general reduction in noise as more density estimates are accumulated is apparent for all values of α, however.

[2] Here we adhere to sigma-clip terminology, with the standard deviation σ being equivalent to the difference $\|\rho_i - \bar{\rho}_{i-1}\|$ for a single sample ρ_i.



(a) $\alpha = 0.2$ (b) $\alpha = 0.4$

(c) $\alpha = 0.6$ (d) $\alpha = 0.8$ (e) $\alpha = 1.00$

Fig. 5. Renderings of prismatic blinds for different bandwidth reduction rates α after 128 iterations. While residual bias in the caustics increases with α, the residual noise drops due to the inherent tradeoff between the two errors. At $\alpha = 1.0$ no bandwidth reduction takes place, and thus bias is most apparent.

5. Results

5.1. Analysis of bandwidth reduction

An error analysis was conducted with renderings of the prismatic blinds to assess the convergence of progressive photon mapping using different bandwidth reduction rates α. Fig. 4 is a graph of the relative deviation of successively accumulated images using unweighted averaging for the renderings of prismatic blinds shown in Fig. 5.

For each iteration, the graph quantifies the relative deviation between the current and previous accumulated image, which is used as a convergence criterion in our implementation. As expected, the error (mostly consisting of noise[3]) drops as more density estimates are accumulated. The graph also shows different convergence rates for α, since this parameter controls the increase in variance per sample image as the bandwidth is reduced. On average, however, the variance drops for all values of α as more sample images are accumulated.

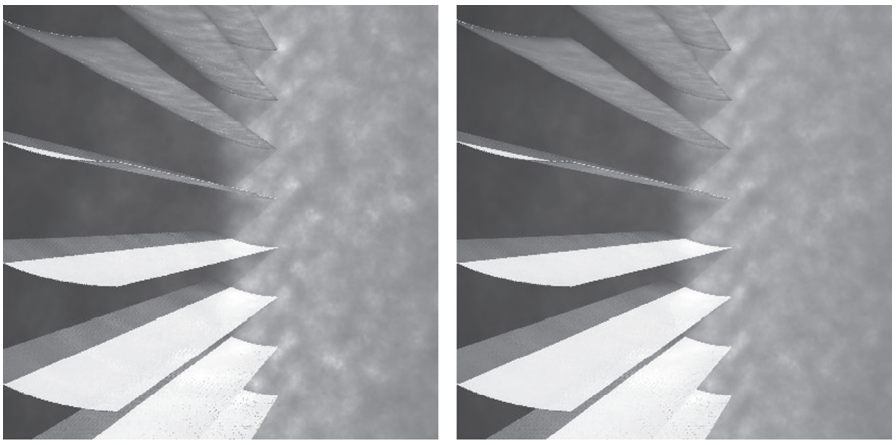
The graph also shows the bandwidths used at each iteration as a function of the reduction factor α. In the case of $\alpha = 1.0$, the initial bandwidth (20 photons) remains unchanged as no bandwidth reduction takes place as defined by Eq. (1), hence this is omitted in the figure. The bandwidth levels once the minimum bandwidth of 2 photons is reached.

While the script computes the bandwidth as floating point value for each iteration, the nearest neighbour lookups for density estimation can only search for an integer number of photons. Therefore, the bandwidth is clamped to its ceiling for actual rendering, and is consequently plotted as such in the figure.

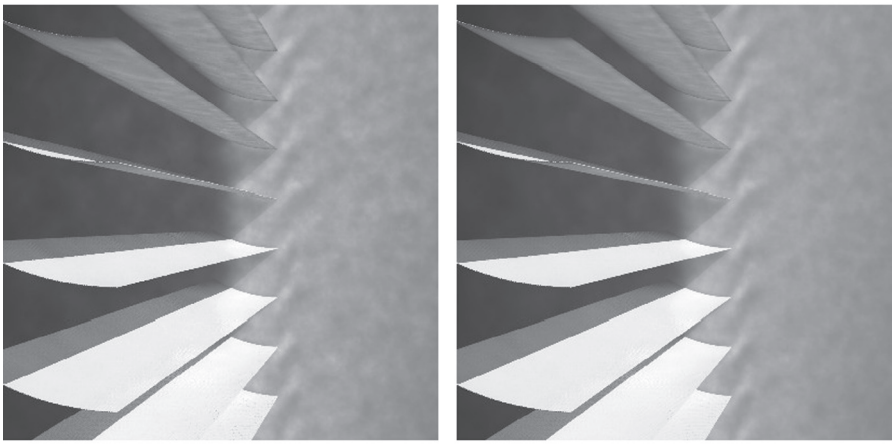
This explains the apparent "jumps" in the deviation particularly noticeable for values of α below 0.8; as the bandwidth drops, noise rises and the effect of omitting a photon from the density estimate due to the integer clamping becomes more pronounced. This results in apparent strata in the deviation, which is particularly noticeable once the minimum bandwidth is reached, as the deviation curves then coincide independently of α.

The corresponding renderings in Fig. 5 show different levels of residual bias and noise after 128 iterations. As expected, noise drops towards higher values of α as the bandwidth reduction is more gradual; consequently, the bias (apparent as blurring in the caustics) increases with

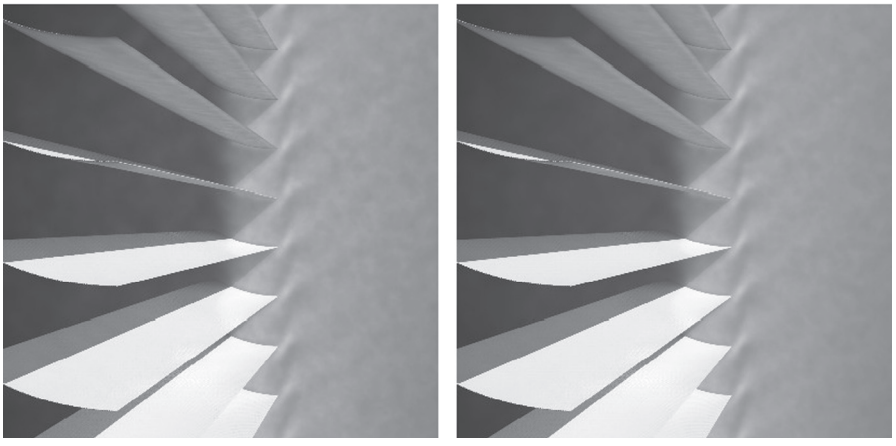
[3] The error primarily measures the reduction in noise and a small amount of relative bias between successive averages; the reduction in absolute bias cannot be quantified without knowing the actual photon density, which is precisely what is being estimated.



(a) 8 iterations (12 sec) (b) 16 iterations (22 sec)



(c) 32 iterations (42 sec) (d) 64 iterations (82 sec)



(e) 128 iterations (163 sec) (f) 256 iterations (328 sec)

Fig. 6. Retroreflecting prismatic blinds rendered at various stages of progression (iteration number and runtime). More than 10 million photons were generated over 256 iterations. Noise is suppressed by selective averaging using sigma-weighting, while bias in the retroreflected caustics is reduced through bandwidth reduction by a factor $\alpha = 0.6$.

α. At $\alpha = 1.0$ the bandwidth is constant, and the bias is clearly evident.

Kaplanyan and Dachsbacher (2013) analysed the asymptotic convergence rate of Knaus and Zwicker's progressive radiance estimate, obtaining an optimal value for the bandwidth reduction α of 0.6. With this parameter, the trade-off between variance and bias is balanced to the effect of asymptotically minimising the mean squared error. Note that this is not apparent in Fig. 4 as the bias is not taken into account. We adopt Kaplanyan and Dachsbacher's proposed optimal value of $\alpha = 0.6$ for the remainder of this paper.

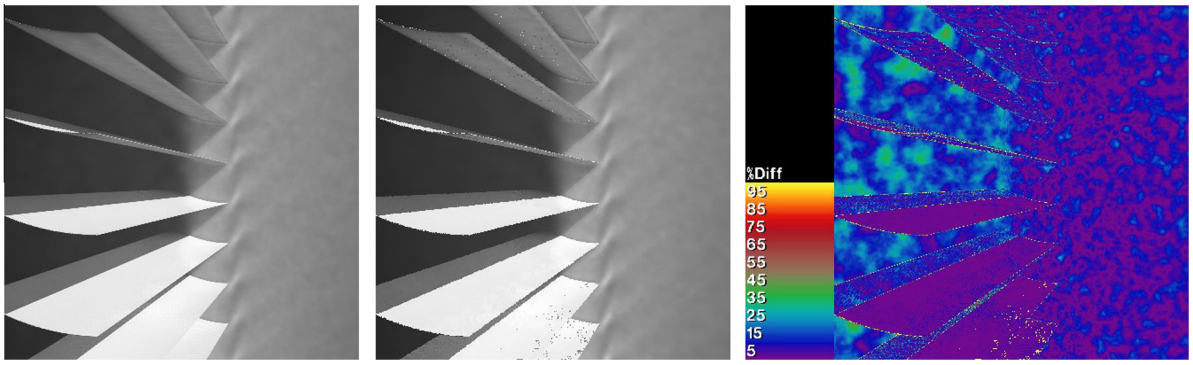


Fig. 7. Comparison of progressive photon mapping using sigma-weighting with 256 iterations (left) and static photon mapping with equivalent parameters (centre, 10,240,000 photons total, 597 photons bandwidth). The relative deviations in the falsecolour image (right) confirm both renderings agree within 4% on average.

5.2. Visual convergence

Fig. 6 shows a series of accumulated images of prismatic blinds generated with progressive photon mapping using selective averaging with sigma-weighting to combine the images. The initial bandwidth for nearest neighbour density estimates is 10 photons, which is reduced with the factor $\alpha = 0.6$ according to Eq. (1) until a minimum bandwidth of 2 is reached. At each iteration, 40,000 photons are distributed, such that an effective photon map size of over 10 M photons is obtained after 256 iterations. In this example, 8 iterations are computed in parallel at a resolution of 400×400. The time to render was just under 5.5 min on an 8-core Intel Xeon system running at 2.40 GHz.

As can be seen in this progression, noise in the density estimates is reduced through accumulation and selective averaging, while the bias is mitigated by bandwidth reduction, thus preserving the caustics visible on the window frame which characterise the blinds' retroreflective behaviour.

It would be challenging for a non-expert user to make an educated guess at a suitable number of photons for this simulation using static photon mapping. The user would also lack a visual feedback of the expected quality of the renderings if the parametrisation were infact sound, implying a degree of uncertainty until the results are available.

With progressive photon mapping, the user has the ability to ascertain the final quality of the rendering at an early stage, and can interrupt the progression, affording him/her a more immediate level of control compared to static photon mapping.

5.3. Validation with static photon mapping

As a non-systematic validation, we compared the renderings of the prismatic blinds obtained with progressive photon mapping with those from the standard RADIANCE photon map (validated in (Schregle and Wienold, 2004)) using parameters equivalent to the final accumulated progressive image. The equivalent parameters for static photon mapping are simply the sum of all constituent photon map sizes and bandwidths of the progressive rendering; after 256 iterations, this amounts to an effective photon map size of 10,240,000 photons and a bandwidth of 597 photons for $\alpha = 0.6$ and an initial bandwidth of 10 photons. The results are shown side by side in Fig. 7.

It is clear that the results are not only visually similar, but also numerically, as evidenced in the falsecolour image of the relative deviations, which lie on average at ca. 4%. The significant deviations are caused by aliasing in the specular component of the blinds due to undersampled primary rays in the static rendering (although this can be increased with an appropriate `rpict` parameter). Maximum deviations of around 30% are noticeable in the dark interior behind the blinds, which are attributed to low frequency noise.

Progressive photon mapping not only yielded similar results in this test but also effected nearly a twofold speedup (5.5 vs 11.8 min) over static photon mapping.

6. Conclusion and outlook

We have presented an application of progressive photon mapping for the accurate simulation of daylight redirecting components, using retroreflecting prismatic blinds as a representative case study. Our proof-of-concept implementation based on the validated RADIANCE photon map shows great potential as a valuable daylight planning tool for components exhibiting strong redirection.

In addition to the efficient simulation of caustics fundamental to all photon mapping variants, progressive photon mapping provides a preview rendering of the simulation as it undergoes refinement. As a result, the user is relieved of the intricate parametrisation of static photon mapping and can readily assess the quality of the simulation "on the fly". We believe that such a tool will be well received in the daylight simulation community by experts and novices alike.

Having served as a testbed, our current implementation in script form needs further development. While the partial photon maps generated at each iteration are saved, they are not reused. It would make sense to extend the functionality to include these in subsequent renderings (e.g at higher

resolutions or from different viewpoints), rather than regenerating the photon maps from scratch.

The Perl Data Language, while effective at handling large matrices, depends on many nonstandard Perl modules to run the script, which must be manually installed via CPAN (2014). This requires expert knowledge of Perl which cannot be generally expected from the target audience.

To address these issues, we plan a more robust and extensible reimplementation in a higher-level scripting language such as Python, or in RADIANCE's native C programming language.

Acknowledgements

This research was supported by the Swiss National Science Foundation as part of the project "Simulation-based assessment of daylight redirecting components for energy savings in office buildings" (#147053).

References

Appelfeld, D., Svendsen, S., 2013. Performance of a daylight-redirecting glass-shading system. Energy Build. 64 (0), 309–316, http://dx.doi.org/10.1016/j.enbuild.2013.05.017.

Chan, Y.-C., Tzempelikos, A., 2012. A hybrid ray-tracing and radiosity method for calculating radiation transport and illuminance distribution in spaces with venetian blinds. Solar Energy 86 (11), 3109–3124, http://dx.doi.org/10.1016/j.solener.2012.07.021.

Christensen, P.H., 1999. Faster photon map global illumination. J. Graph. Tools 4, 1–10.

Christensen, P.H., Jensen, H.W., Kato, T., Suykens, F., 2004. A practical guide to global illumination using ray tracing and photon mapping. ACM SIGGRAPH 2004 Course Notes. SIGGRAPH '04. ACM, New York, NY, USA, pp. 1–248, http://doi.acm.org/10.1145/1103900.1103920.

CPAN, 2014. The comprehensive perl archive network. <http://www.cpan.org>.

de Boer, J., 2006. Modelling indoor illumination by complex fenestration systems based on bidirectional photometric data. Energy Build. 38 (7), 849–868, special Issue on Daylighting Buildings. http://dx.doi.org/10.1016/j.enbuild.2006.03.018.

Hachisuka, T., Ogaki, S., Jensen, H.W., 2008. Progressive photon mapping. ACM Trans. Graph. 27 (5), 130:1–130:8, http://doi.acm.org/10.1145/1409060.1409083.

Jensen, H.W., 2001. Realistic Image Synthesis Using Photon Mapping. A.K. Peters, Ltd., Natick, MA, USA.

Kaplanyan, A.S., Dachsbacher, C., 2013. Adaptive progressive photon mapping. ACM Trans. Graph. 32 (2), 16:1–16:13, http://cg.ivd.kit.edu/english/APPM.php.

Kennedy, L.A., 2012. One-Shot Color Astronomical Imaging. In: Patrick Moore's Practical Astronomy Series. Springer, Boston.

Klammt, S., Neyer, A., Müller, H.F., 2012. Redirection of sunlight by microstructured components simulation, fabrication and experimental results. Solar Energy 86 (5), 1660–1666, http://dx.doi.org/10.1016/j.solener.2012.02.034.

Knaus, C., Zwicker, M., 2011. Progressive photon mapping: a probabilistic approach. ACM Trans. Graph. 30 (3), 25:1–25:13, http://doi.acm.org/10.1145/1966394.1966404.

Köster, H., 2004. Dynamic Daylighting Architecture. Birkhäuser Verlag, pp. 220–243.

Köster, H., Jan. 25 2005. Toothed daylight blinds. US Patent 6,845,805. <http://google.com/patents/US6845805>.

Köster, H., Oct. 4 2012. Lichtlenkjalousie mit prismatisch ausgeformten Lamellenoberflächen zur Ein- und Auslenkung von Sonnenlicht. DE Patent 102,009,056,362. <https://register.dpma.de/DPMAregister/pat/PatSchrifteneinsicht?docId=DE102009056362B4>.

McNeil, A., Jonsson, C., Appelfeld, D., Ward, G., Lee, E., 2013. A validation of a ray-tracing tool used to generate bi-directional scattering distribution functions for complex fenestration systems. Solar Energy, Part C 98 (0), 404–414, http://dx.doi.org/10.1016/j.solener.2013.09.032.

PDL, 2014. The perl data language. <http://pdl.perl.org>.

Schregle, R., 2003. Bias compensation for photon maps. Comput. Graph. Forum 22 (4), 729–742, http://dx.doi.org/10.1111/j.1467-8659.2003.00720.x.

Schregle, R., 2004. Daylight simulation with photon maps. Ph.D. thesis, Universität des Saarlandes, Saarbrücken. <http://scidok.sulb.uni-saarland.de/volltexte/2007/1171>.

Schregle, R., Wienold, J., 2004. Physical validation of global illumination methods: measurement and error analysis. Comput. Graph. Forum 23 (4), 761–781, http://dx.doi.org/10.1111/j.1467-8659.2004.00807.x.

Su, Y., Pei, G., Riffat, S.B., Huang, H., 2012. Radiance/pmap simulation of a novel lens-walled compound parabolic concentrator (lens-walled cpc). Energy Procedia 14 (0), 572–577, 2011 2nd International Conference on Advances in Energy Engineering (ICAEE). http://dx.doi.org/10.1016/j.egypro.2011.12.977.

Ward, G.J., 1994. The radiance lighting simulation and rendering system. Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH '94. ACM, New York, NY, USA, pp. 459–472, http://doi.acm.org/10.1145/192161.192286.

Wittkopf, S., Grobe, L.O., Geisler-Moroder, D., Compagnon, R., Kämpf, J., Linhart, F., Scartezzini, J.-L., 2010. Ray tracing study for non-imaging daylight collectors. Solar Energy 84 (6), 986–996, http://dx.doi.org/10.1016/j.solener.2010.03.008.