REAL-TIME VIDEO BASED LIGHTING USING GPU RAYTRACING

Joel Kronander[1], *Johan Dahlin*[2], *Daniel Jönsson*[1], *Manon Kok*[2], *Thomas B. Schön*[3], *Jonas Unger*[1]

[1] Dept. of Science and Technology, Linköping University
[2] Dept. of Electrical Engineering, Linköping University
[3] Dept. of Information Technology, Uppsala University

ABSTRACT

The recent introduction of HDR video cameras has enabled the development of image based lighting techniques for rendering virtual objects illuminated with temporally varying real world illumination. A key challenge in this context is that rendering realistic objects illuminated with video environment maps is computationally demanding.

In this work, we present a GPU based rendering system based on the NVIDIA OptiX [1] framework, enabling real time raytracing of scenes illuminated with video environment maps. For this purpose, we explore and compare several Monte Carlo sampling approaches, including bidirectional importance sampling, multiple importance sampling and sequential Monte Carlo samplers. While previous work have focused on synthetic data and overly simple environment maps sequences, we have collected a set of real world dynamic environment map sequences using a state-of-art HDR video camera for evaluation and comparisons.

***Index Terms*—** Image Based Lighting, HDR Video, Video Based Lighting

1. INTRODUCTION

Image based lighting (IBL) [2] enables photo-realistic rendering and seamless integration of virtual objects into photographs and videos captured in real scenes. This is carried out by driving the lighting simulation during rendering using illumination captured in the real scene, using carefully calibrated High Dynamic Range (HDR) images. Traditional approaches capture a single panoramic image to represent the incident illumination in the scene [2, 3]. While a single panoramic image works well for still images, seamless integration of rendered objects into video footage requires capturing the temporal variations in the illumination. This requirement has entailed the development of IBL methods using panoramic HDR video to capture the scene illumination as a sequence of environment maps [4, 5].

This work was funded by the Swedish Foundation for Strategic Research (SSF) through grant IIS11-0081, the project Probabilistic modeling of dynamical systems (Contract number: 621-2013-5524) funded by the Swedish Research Council and the Linnaeus Environment CADICS also funded by the Swedish Research Council.

However, realtime rendering with such video based lighting techniques has previously been limited to diffuse materials and low-frequency illumination [4, 6]. In this work, we show how realtime raytracing in the OptiX [1] framework can be used to render glossy materials in high-frequency video environment maps. To this end, we explore both approaches sampling each frame separately [15, 18] and that exploit the correlation among frames using sequential Monte Carlo (SMC) samplers [7].

We focus the comparison on realtime HDR video sequences captured using a state-of-the art HDR video camera. These sequences pose several challenges as that they reflect a much wider range of possible temporal variation than previously considered. The main contributions of this work are

- A GPU based solution for realtime raytracing using video environment map illumination.
- Evaluation and comparison of Monte Carlo estimators for rendering with video environment map illumination.

2. CAPTURING HDR VIDEO PANORMAS

To represent the incident illumination, we capture light probe images for each frame by utilizing a standard non-central catadioptric imaging system based on a near orthographic lens and a mirror ball, depicted in Figure 2. To efficiently match the captured incident illumination to a synchronized backplate sequence, we use a consumer video camera mounted on top of the HDR camera capturing the light probe. This arrangement allows for high-resolution background footage, but incurs a small error if considerable spatial variations in the incident illumination is present.

To accurately capture the full dynamic range of the incident illumination varying over time in a video light probe, we relay on recently developed HDR video technology [8]. At present, there exist a multitude of different methods for capturing HDR video. Many of these are of limited use for a versatile IBL pipeline, as they often still can not accurately capture the complete dynamic range of an outdoor environment containing direct sunlight and deep shadows. Instead, many commercially available HDR video cameras are in practice

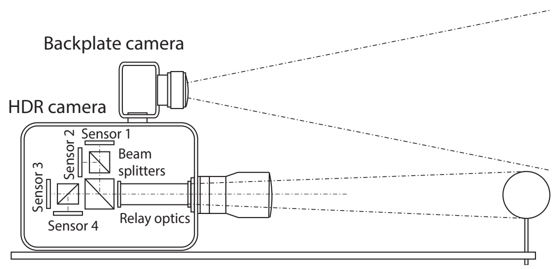

Fig. 1. The video light probe is captured using a multi-sensor HDR video camera synchronized to a high resolution backplate video camera.

limited to 15-17 f-stop. Arguably, the currently best HDR video camera systems for temporal IBL in terms of resolution, noise characteristics, and overall image quality are based on setups with multiple, synchronized sensors that simultaneously capture different exposures of the scene. For this work, we use a state-of-the-art multi-sensor HDR video camera [9, 10] developed by Linköping University and Spheron VR, capable of capturing a dynamic range of 24 f-stops.

Before further processing, the light probe images are stabilized, removing vibrations in the light probe relative to the camera, so that the camera lens is in the center of the light probe images. The light probes are then converted into a latitude longitude mapping, color corrected to match the background footage and blurred slightly in the hidden directions directly behind the light probe, for details of this process see section 7.4.1 in [11].

3. RENDERING

The reflected radiance $L_r(\mathbf{x}, \omega_o, t)$ leaving a point $\mathbf{x}$ on a surface in direction ω_o at frame t is given by

$$L_r(\mathbf{x}, \omega_o, t) = \int_\Omega L_t(\mathbf{x}, \omega)\rho(\mathbf{x}, \omega_o, \omega)V(\mathbf{x}, \omega)(\omega \cdot \mathbf{n})\mathrm{d}\omega, \tag{1}$$

where Ω denotes the visible hemisphere, $L_t(\mathbf{x}, \omega)$ is the incident radiance arriving at the point $\mathbf{x}$ from direction ω for frame t. Here, $\rho(\mathbf{x}, \omega_o, \omega)$ is the bidirectional reflectance distribution function (BRDF), $V(\mathbf{x}, \omega)$ is a binary visibility function and $\mathbf{n}$ denotes the surface normal.

In this work, we are concerned with the case when $L_t(\mathbf{x}, \omega)$ is represented by a HDR environment map describing the incident radiance onto the objects in the scene. To render an animation we use a separate environment map representing the incident illumination at each frame, $L_1(\mathbf{x}, \omega), L_2(\mathbf{x}, \omega), ...$ To render the scene, we need to evaluate the reflected radiance given by (1) for a large number of sampled scene points for each frame.

The integral in (1) is often evaluated using Monte Carlo importance sampling. For the static case, treating each frame separately, several different importance functions have been investigated in previous work. Approaches sampling proportional to the BRDF [12] performs better for glossy materials in low-frequency environments. When the environment maps contains high-frequency content, environment sampling approaches [13, 14] performs better. In scenes with possibly both glossy BRDFs and high frequency environment maps, approaches sampling proportional to the product [15] is necessary for efficient rendering.

Relighting applications using video environment maps also enables utilizing the correlation among frames. Havran *et al.* [4] considered temporal filtering of the samples proposed from the environment maps in each frame, however this approach can lead to overly smooth results. Other approaches have focused on off-line rendering when the complete sequence of environment maps are available by stratifying samples from the environment map sequence in both the spatial and temporal domain [16, 17]. Ghosh *et al.* [7] proposed to propagate a carefully tuned approximation of the product between the BRDF and the environment map between frames using SMC sampling. Using a CPU rendered they were able to show reduced variance compared to other approaches in equal computational time.

In this work, we focus on approaches that enables rendering general material appearance under a range of different illumination conditions, including low frequency and high frequency in the same video environment map sequence. For this purpose we have chosen to investigate further three representative approaches. The simplest and most widespread is multiple importance sampling (MIS) [18], efficiently combining samples proposed from the BRDF and the environment map in each frame. The second, bidirectional importance sampling is a product sampling approach [15], proposing samples from the product of the BRDF and the environment map. Finally we also compare these approaches which treat each frame separately to the approach by Ghosh *et al.* [7] which exploits the correlation among frames using Sequential Monte Carlo samplers.

4. MULTIPLE IMPORTANCE SAMPLING

To estimate the reflected radiance, MIS samples both the environment map and the BDRF. These samples are weighted such that large variance spikes due to mismatch between one of the proposals and the shape of the integrand. The result is a very robust method that can handle both glossy materials and high frequency illumination. Drawing N_ρ samples from the BRDF and N_L samples from the environment map, the MIS estimator of the reflected radiance (1) using the balance heuristic [18] is given by

$$\widehat{L}_{\text{MIS}} = \sum_{i=1}^{N_\rho+N_L} \frac{L_t(\mathbf{x},\omega^i)\rho(\mathbf{x},\omega_o,\omega^i)V(\mathbf{x},\omega^i)(\omega^i\cdot\mathbf{n})}{q_\rho(\omega^i)+q_L(\omega^i)}, \tag{2}$$

where $q_\rho(\omega^i)$ is the importance function used to draw samples proportional to the BRDF and $q_L(\omega^i)$ is the importance function used to draw samples from the environment map.

5. BIDIRECTIONAL IMPORTANCE SAMPLING

In some cases, one factor of the integrand is more expensive to compute than the other. For this end, Burke *et al.* [15] proposed to draw samples from a target distribution given as the product of the other factors using sampling importance resampling. In general the most costly factor to evaluate is the visibility factor, thus the target distribution is given by

$$\frac{\tilde{\gamma}_t(\omega)}{Z} = \frac{L_{Y,t}(\mathbf{x},\omega)\rho_Y(\mathbf{x},\omega_o,\omega)(\omega\cdot\mathbf{n})}{\int_\Omega L_{Y,t}(\mathbf{x},\omega)\rho_Y(\mathbf{x},\omega_o,\omega)(\omega\cdot\mathbf{n})\mathrm{d}\omega}, \tag{3}$$

where $L_{Y,t}(\mathbf{x},\omega)$ and $\rho_Y(\mathbf{x},\omega_o,\omega)$ are the luminance of the color valued $L_t(\mathbf{x},\omega)$ and $\rho(\mathbf{x},\omega_o,\omega)$, respectively, $\tilde{\gamma}_t(\omega) = L_{Y,t}(\mathbf{x},\omega)\rho_Y(\mathbf{x},\omega_o,\omega)(\omega\cdot\mathbf{n})$ is the unnormalized target and $Z = \int_\Omega L_{Y,t}(\mathbf{x},\omega)\rho_Y(\mathbf{x},\omega_o,\omega)(\omega\cdot\mathbf{n})\mathrm{d}\omega$ is a normalization constant corresponding to the un-occluded reflected radiance. By first sampling from a distribution, $q(\omega)$, for example proportional to the BRDF, these samples can then be weighted and resampled to obtain a new set of samples approximating the target distribution. This is done by the empirical approximation $\widehat{\gamma}_t(\omega) = \sum_{i=1}^N \frac{1}{N}\delta_{\omega_t^i}(\omega)$. Using this approximated target distribution, the reflected illumination can be estimated using

$$\widehat{L}_r(\mathbf{x},\omega_o,t) = \widehat{Z}\sum_{i=1}^N \frac{\tilde{L}_t(\mathbf{x},\omega_t^i)\rho(\mathbf{x},\omega_o,\omega_t^i)V(\omega_t^i)}{\tilde{L}_{Y,t}(\mathbf{x},\omega_t^i)\rho_Y(\mathbf{x},\omega_o,\omega_t^i)}, \tag{4}$$

where $\widehat{Z} = \frac{1}{N}\sum_{i=1}^N \tilde{L}_{Y,t}(\mathbf{x},\omega^i)\rho_Y(\mathbf{x},\omega_o,\omega^i)(\omega^i\cdot\mathbf{n})$.

6. SMC SAMPLERS

Instead of approximating the target distribution for $t = 1, 2, ...$ independently we can use SMC samplers to exploit the correlation of the target distribution between frames. This is motivated by the fact that for real-time HDR video environment maps the incident illumination often varies relatively slowly, i.e. $L_t(\mathbf{x},\omega) \approx L_{t-1}(\mathbf{x},\omega)$.

Sequential Monte Carlo [19] is a family of methods for sampling from a set of target distributions of growing dimension. For the sequence of target distributions we are interested in here (3) the dimension is constant and therefore standard SMC methods cannot be applied directly. However, it is possible to reformulate the problem by introducing artificial target distributions defined on a space of increasing dimension. By this construction, the desired target distribution can be found by marginalizing over the auxiliary dimensions, see [20]. To approximate the target distribution (3) the resulting SMC sampling algorithm is carried out in two steps, a *propagation step* and an *adaption step*.

6.1. Propagation step

Assume that there exists a set of N weighted samples denoted $\{\omega_{t-1}^i, w_{t-1}^i\}_{i=1}^N$. At frame $t-1$ this set approximates the target by the empirical approximation $\widehat{\gamma}_t(\omega) = \sum_{i=1}^N W_{t-1}^i \delta_{\omega_{t-1}^i}(\omega)$ where $W_{t-1}^i = \frac{w_{t-1}^i}{\sum_{i=1}^N w_{t-1}^i}$ denote the normalized weights. For the first frame, such an approximation can be sampled using any product sampling approach. In this work, we use the bidirectional importance sampling proposed discussed in Section 5.

To approximate the target at t, the samples at $t-1$ are first propagated forward using sequential importance sampling by simply reweighing the existing samples. With an appropriate choice of artificial target distributions, for details see [20], the new unnormalized weights simplify to

$$w_t^i = W_{t-1}^i \frac{\gamma_t(\omega_{t-1}^i)}{\gamma_{t-1}(\omega_{t-1}^i)}. \tag{5}$$

To limit the degeneracy of the approximation over time, i.e. only a few samples receiving significant weights, resampling is performed when the effective sample size (ESS) defined by $\sum_{i=1}^N ({W_t^i}^2)^{-1}$ is below a pre-specified threshold. In this work, we use $\frac{2}{3}N$ as this threshold. The resampling step draws new samples ω_t^i with replacement from the weighted sample set with a probability proportional to the weights. The new sample set is composed of the N resampled samples with equal weights, i.e. $w_t^i \equiv \frac{1}{N}$.

6.2. Adaption step

To improve the approximation of the target distribution at frame t, the samples $\{\omega_t^i, w_t^i\}_{i=1}^N$ are further adapted using an Markov Chain Monte Carlo (MCMC) kernel $K_t(\omega,\omega')$ with the desired target as the invariant distribution. The MCMC kernel is constructed using the Metropolis-Hastings (MH) algorithm. Using MH, the MCMC kernel is described by the acceptance probability

$$a(\omega\to\omega') = \min\left(1, \frac{\widetilde{\gamma}_t(\omega_t'^j)}{\widetilde{\gamma}_t(\omega_t^j)}\frac{q(\omega_t'^j\to\omega_t^j)}{q(\omega_t^j\to\omega_t'^j)}\right), \tag{6}$$

where ω_t^j is the current sample, $\omega_t'^j$ is the proposed sample using the proposal distribution $q(\cdot)$ and $a(\cdot)$ denotes the acceptance probability of the transition.

We follow [7] and design a proposal distribution using a mixture of local moves with some probability ν and and independent moves with probability $1-\nu$. This is done to prevent the samples from getting stuck in local narrow modes. The local moves are represented by a uniform random perturbations of the current samples by a few degrees. The independent moves are represented by drawing new samples from the environment map or the BRDF.

6.3. Reflected illumination estimate

Given the weighted sample set obtained by from the two steps in the SMC algorithm, we can estimate the reflected surface radiance (1) as

$$\widehat{L}_r(\mathbf{x}, \omega_o, t) = Z_t \sum_{i=1}^{N} W_t^i \frac{\tilde{L}_t(\mathbf{x}, \omega_t^i)\rho(\mathbf{x}, \omega_o, \omega_t^i)V(\omega_t^i)}{\tilde{L}_{Y,t}(\mathbf{x}, \omega_t^i)\rho_Y(\mathbf{x}, \omega_o, \omega_t^i)}. \quad (7)$$

One advantage of the SMC rendering algorithm is that the normalization constant Z_t can be incrementally estimated by using the relation

$$Z_t \approx Z_{t-1} \sum_{i=1}^{N} w_t^i. \quad (8)$$

For details on the derivation of this expression see [7]. Z_1 can be estimated from the samples obtained via bidirectional importance sampling in the first frame.

The SMC algorithm presented here works well under the assumption that $\gamma_t(\omega) \approx \gamma_{t-1}(\omega)$. If the discrepancy between two successive target distributions is large, this results in a high variance in the sample weights. This as the weights are computed before applying the MCMC kernel, which can correct the discrepancy to some amount. The variance in the particle weights can be reduced using resampling. This results in a good approximation of the unnormalized target distribution $\tilde{\gamma}_t(\omega)$. However, as the resampling operation does not affect the computation of the normalization constant Z_t, this estimate is likely to be poor when the target distribution changes rapidly. This is typically the case for real HDR video environment maps. To smooth the transition one can use a set of intermediate distributions [7] to guide the samples smoothly between the targets. As recommended in [7] we use one MCMC move for each intermediate distribution to adapt the samples gradually.

A big drawback of the original SMC rendering algorithm from [7] is that as time progresses, the variance of the estimated normalizing constant tends to increase. This can lead to visually disturbing artifacts. To counter this, we propose a simple but effective approach where we monitor the change in $\frac{Z_t}{Z_{t-1}}$ and when a large increase occurs we reinitialize the particle buffer and normalizing constant estimate using bidirectional importance sampling.

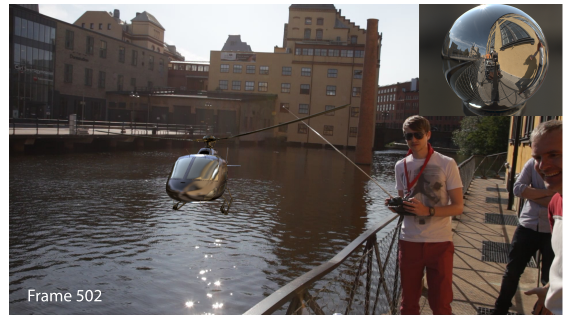

Fig. 2. Real-time rendered helicopter model (25 fps using SMC) composited into the backplate video sequence

7. IMPLEMENTATION

To enable real-time raytracing with video environment maps we have implemented the three rendering algorithms described above in the CUDA based OptiX framework of NVIDIA [1], running on the GPU. We use a regular sampling of the image plane to spawn a set of rays into the scene. Our implementation currently only considers direct illumination, however it's trivial to extend it to path tracing as well, by spawning a new ray from the shading point. To handle several samples per pixel we utilize two OptiX kernels. The first kernel spawns R rays per pixel and updates the associated sample buffer. The second kernel filters the reflected radiance of the sampled ray locations using an reconstruction filter and tonemaps the image for display. In the examples presented here, we used a box filter for reconstruction and a gamma mapping for tonemapping.

To sample from the environment map we use the numerical inversion method presented in []. The environment maps at frame t and $t-1$ are accessed through two texture samplers, enabling efficient lookups using the texture hardware on the GPU. To draw samples from the environment map we precompute tabulated column and row CDF on the CPU and upload this to a read-only global GPU buffer before rendering the frame.

For the SMC rendering algorithm we for each queried shading point we read, compute and store the sampled directions and weights $\{\omega_t^i, w_t^i\}_{i=1}^N$ in a 3D floating point buffer residing in global GPU memory indexed using the ray origin.

8. RESULTS AND COMPARISIONS

To evaluate the performance of our method we used a NVIDIA geforce GTX [[XXX describe results...]]

REFERENCES

[1] Steven G Parker, James Bigler, Andreas Dietrich, Heiko Friedrich, Jared Hoberock, David Luebke, David McAllister, Morgan McGuire, Keith Morley, Austin Robison, et al., "OptiX: a general purpose ray tracing engine," *ACM Transactions on Graphics (TOG)*, vol. 29, no. 4, pp. 66, 2010.

[2] Paul Debevec, "Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography," in *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*, New York, NY, USA, 1998, SIGGRAPH '98, pp. 189–198, ACM.

[3] Francesco Banterle, Alessandro Artusi, Kurt Debattista, and Alan Chalmers, *Advanced high dynamic range imaging: theory and practice*, CRC Press, 2011.

[4] Vlastimil Havran, Miloslaw Smyk, Grzegorz Krawczyk, Karol Myszkowski, and Hans-Peter Seidel, "Interactive system for dynamic scene lighting using captured video environment maps," in *Proceedings of the Sixteenth Eurographics Conference on Rendering Techniques*, Aire-la-Ville, Switzerland, Switzerland, 2005, EGSR'05, pp. 31–42, Eurographics Association.

[5] Jonas Unger, Joel Kronander, Per Larsson, Stefan Gustavson, and Anders Ynnerman, "Temporally and spatially varying image based lighting using hdr-video," in *Proceedings of EUSIPCO'13: Special Session on HDR-video*, 2013.

[6] Saghi Hajisharif, "Real-time image based lighting with streaming hdr-light probe sequences," 2012.

[7] Abhijeet Ghosh, Arnaud Doucet, and Wolfgang Heidrich, "Sequential sampling for dynamic environment map illumination," in *Proceedings of the 17th Eurographics conference on Rendering Techniques*. Eurographics Association, 2006, pp. 115–126.

[8] Karol Myszkowski, *High Dynamic Range Video*, Morgan and Claypool Publishers, 1 edition, 2008.

[9] Joel Kronander, Stefan Gustavson, Gerhard. Bonnet, and Jonas. Unger, "Unified HDR reconstruction from raw CFA data," in *Computational Photography (ICCP), 2013 IEEE International Conference on*, April 2013, pp. 1–9.

[10] Joel Kronander, Stefan Gustavson, Gerhard Bonnet, Anders Ynnerman, and Jonas Unger, "A unified framework for multi-sensor HDR video reconstruction," *Signal Processing: Image Communication*, 2013.

[11] Christian Bloch, *The HDRI Handbook 2.0: High Dynamic Range Imaging for Photographers and CG Artists*, Rocky Nook, 2012.

[12] Jason Lawrence, Szymon Rusinkiewicz, and Ravi Ramamoorthi, "Efficient brdf importance sampling using a factored representation," in *ACM Transactions on Graphics (TOG)*. ACM, 2004, vol. 23, pp. 496–505.

[13] Sameer Agarwal, Ravi Ramamoorthi, Serge Belongie, and Henrik Wann Jensen, "Structured importance sampling of environment maps," in *ACM Transactions on Graphics (TOG)*. ACM, 2003, vol. 22, pp. 605–612.

[14] Thomas Bashford-Rogers, Kurt Debattista, and Alan Chalmers, "Importance driven environment map sampling," *IEEE Transactions on Visualization and Computer Graphics*, 2013.

[15] David Burke, Abhijeet Ghosh, and Wolfgang Heidrich, "Bidirectional importance sampling for direct illumination.," *Rendering Techniques*, vol. 5, pp. 147–156, 2005.

[16] Liang Wan, Tien-Tsin Wong, and Chi-Sing Leung, "Spherical q 2-tree for sampling dynamic environment sequences," in *Proceedings of the Sixteenth Eurographics conference on Rendering Techniques*. Eurographics Association, 2005, pp. 21–30.

[17] Liang Wan, Shue-Kwan Mak, Tien-Tsin Wong, and Chi-Sing Leung, "Spatiotemporal sampling of dynamic environment sequences," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 17, no. 10, pp. 1499–1509, 2011.

[18] Eric Veach and Leonidas J Guibas, "Optimally combining sampling techniques for Monte Carlo rendering," in *Proceedings of SIGGRAPH*, New York, NY, USA, 1995, ACM, pp. 419–428.

[19] Arnaud Doucet and Adam M Johansen, "A tutorial on particle filtering and smoothing: 15 years later," in *Oxford Handbook of Nonlinear Filtering*. Oxford University Press, 2011.

[20] Pierre Del Moral, Arnaud Doucet, and Ajay Jasra, "Sequential monte carlo samplers," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 68, no. 3, pp. 411–436, 2006.