

# Acceleration Methods for Radiance Transfer in Photorealistic Augmented Reality

Lukas Gruber<sup>1\*</sup>

Pradeep Sen<sup>2†</sup>

Tobias Höllerer<sup>2‡</sup>

Dieter Schmalstieg<sup>1§</sup>

<sup>1</sup>Graz University of Technology

<sup>2</sup>University of California, Santa Barbara

## ABSTRACT

Radiance transfer computation from unknown real-world environments is an intrinsic task in probe-less photometric registration for photorealistic augmented reality, which affects both the accuracy of the real-world light estimation and the quality of the rendering. We discuss acceleration methods that can reduce the overall ray-tracing costs for computing the radiance transfer for photometric registration in order to free up resources for more advanced augmented reality lighting. We also present evaluation metrics for a systematic evaluation.

**Index Terms:** H.5.1 [Information Interfaces and Presentation]: Artificial, Augmented, Virtual Realities—;I.4.8 [Image Processing and Computer Vision]: Photometric registration—3D Reconstruction;I.3.3 [Computer Graphics]: Image Generation—Ray Tracing—Spherical Harmonics;

## 1 INTRODUCTION

For many AR applications, the ultimate goal is to be unable to discern between real and virtual scene components. This is achieved by visually coherent rendering, which also implies the correct computation of illumination effects, e.g., applying real-world lighting onto virtual objects and vice versa. Naturally, a key problem of realistic AR lighting is the accurate measurement of the real-world lighting, also known as photometric registration. The two most common light estimation methods in AR are based on observing the reflection of a spherical object (passive light probe) [1] or measuring the environment light directly from an additional wide-angle camera system (active light probe) [6]. These methods are known to be invasive, inserting artificial physical light probes into the scene. Alternatively, we can also measure the real-world lighting from the scene directly [3], which is a non invasive method. Such measurement is known as probe-less photometric registration, which demands more algorithmic effort from the system and requires camera tracking, 3D scene reconstruction and inverse rendering. Radiance transfer is a major component of inverse rendering and has a great impact on the performance of the system. Therefore, we study and evaluate acceleration methods [4] for radiance transfer computation in this work.

## 2 PROBE-LESS PHOTOMETRIC REGISTRATION

Probe-less photometric registration in AR is built upon computing the radiance transfer (RT) from real-world scene geometry and based on the reflection equation 1:

$$R(\vec{x}, \vec{\omega}_0) = \int_{\Omega_{\vec{x}}} L(\vec{\omega}_i) A(\vec{x}) V(\vec{x}, \vec{w}_i) (\vec{\omega}_i \cdot \vec{n}) d\vec{\omega}_i. \quad (1)$$

\*e-mail:lgruber@icg.tugraz.at

†e-mail:psen@ece.ucsb.edu

‡e-mail:holl@cs.ucsb.edu

§e-mail:schmalstieg@icg.tugraz.at

IEEE International Symposium on Mixed and Augmented Reality 2013  
Science and Technology Proceedings  
1 - 4 October 2013, Adelaide, SA, Australia  
978-1-4799-2869-9/13/\$31.00 ©2013 IEEE

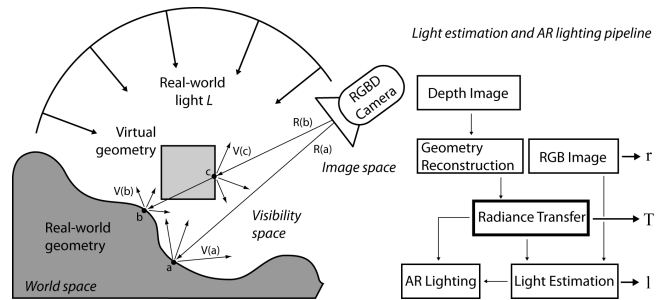


Figure 1: Left: Working volume with different sample spaces. Right: AR photometric registration pipeline

We replace the BDRF with a spatially varying albedo  $A(\vec{x})$ . If we discretize the problem and represent the incident illumination  $L(\vec{\omega}_i)$  as vector  $l$  and the reflected radiance from a specific point and direction as  $r_j$ , the integral becomes an inner product between the lighting vector and a vector  $t_j$  representing the albedo, visibility, and cosine terms:  $r_j = t_j \cdot l$ . To do this for a vector  $r$  containing all of the observations of the scene by every pixel of a camera, Eq. 1 can be written as a linear system  $r = Tl$ , where the rows of the transport matrix  $T$  are vectors  $t_j$ . Note that vector  $l$  can be expressed as Spherical Harmonics (SH) or Wavelets coefficients to compactly represent the distant environment light  $L$  if the appropriate basis is incorporated into  $T$ . The problem of estimating the lighting  $l$  from observations  $r$  is now given by the minimization  $\min_l \|Tl - r\|_2$ . See Fig. 1 for an overview of the entire pipeline. Overall, a major task is the computation of RT of the current scene, which is given by matrix  $T$  and is required to both estimate the lighting as well as generate new renderings. Calculation of  $T$  can be arbitrarily complex, from the simple, directly-illuminated Lambertian surfaces we described in Eq. 1 to surfaces with general BRDFs with complex global illumination from multiple light bounces. These more complex light transport models still exhibit linearity between the light sources and the observations and can therefore be modeled with the same overall framework.

## 3 ACCELERATION METHODS

The following acceleration methods mainly evolve around sub-sampling the working volume and caching the RT workload.

### 3.1 Sampling Spaces and Visibility Signal

In Fig. 1, we illustrate the entire working volume, which can be divided into three sub-sampling spaces: the *world*, the *image* and the *visibility* space. The world space is a Cartesian space that contains the real and virtual scene geometry. From the camera's viewpoint (image space), we measure the reflections  $r$  of scene surfaces as the intensities of the camera pixels. To compute the potential radiance transfer, we cast rays from screen space into world space. Once a ray hits the geometry (e.g., at real-world geometry surface points  $a$  or  $b$  or at virtual geometry surface point  $c$ ), we trace rays uniformly

outward in a sphere to compute the visibility  $V$  of real-world lighting  $L$  for that surface point (visibility space). Note that we compute the radiance transfer twice, once for the real world only  $(a, b)$  and then for the real world and the virtual world together  $(a, b, c)$  to support differential rendering [2]. Because we only care about visible areas on the screen, the three different sample spaces can be expressed by the visibility signal which can be parameterized as a 4D function  $V(s, t, \phi, \theta)$ , where surface point  $x$  is projected onto a pixel  $(s, t)$  in the image plane and  $(\phi, \theta)$  encode the direction of the visibility ray in polar coordinates.  $V$  is a binary function, which is 0 if a surface point is completely blocked by another surface in direction  $(\phi, \theta)$  or 1 if the ray can “see” the distant light.

### 3.2 Regular Sampling

We sample image space using fixed regular intervals  $(s, t)$  and compute the RT for every  $n^{\text{th}}$  pixel  $(s, t)$  where  $n$  is the sample step size. The result is a 2D light map with a lower resolution. This naturally creates aliasing artifacts at surface edges. To fight aliasing we applied two improvements: One is a post-processing anti-aliasing approach proposed by [5]. The other one is an approach which samples surface edges at higher density. This necessitates the detection of edges from the virtual object and the boundaries between virtual and real objects. To prevent aliasing in image space we also considered sampling in visibility space,  $(\phi, \theta)$  by using subsets of all visibility ray directions. The consequence are lower quality shadows in intermediate frames.

### 3.3 Interleaved Sampling Over Time

A strategy for easing sampling artifacts in image space is distributing the regularly spaced samples over time and image space using an interleaved mapping scheme. Over a fixed period of time, the entire RT is computed. The time period measured in frames corresponds to the square of the sampling step size  $n$ . To cover the entire image space, the regular sampling is shifted each frame according to an appropriate interleaved mapping scheme. The final light map is then computed from  $n$  intermediate results. Since the camera may be moving, we have to perform a temporal re-projection to correctly assemble the pixels. Although this provides a more complete signal, processing may not be entirely free of errors, since the re-projection can fail for fast camera movements or changing geometry. Interleaved sampling can also be applied in visibility space.

### 3.4 Caching

A complementary acceleration method to sub-sampling is storing and reusing RT information (SH coefficients). RT naturally depends on the real world and virtual geometry. Therefore in our case the design of a RT cache has to support dynamically changing geometry. We built an agile 2D cache aligned to image space. The cache is based on a ping-pong buffer (2 buffers) and the cache indexing is done through re-projection over the reconstructed geometry. So far we update the cache on a fixed time interval.

## 4 EVALUATION RESULTS

We implemented 7 acceleration methods. Four methods are based on regular sampling, two with  $n = 2$  and  $n = 4$  step sizes respectively and one post processing and one edge based anti aliasing method based on  $n = 4$ . Two interleaved sampling methods, one in image space and one in visibility space. The seventh method is based on RT caching. The systematic evaluation covers the following AR design space: dynamic cameras, dynamic geometry and dynamic changing lighting.

### 4.1 Metrics

Our metrics - system performance, visual quality, and light estimation quality - are driven by key requirements of photorealistic

AR: real-time use and rendering quality, implying coherent real-world lighting. In the evaluation process, we compare results of the acceleration methods against results of the reference method, using synthetic generated (known light sources, geometry and camera movements) and real-world input data. The reference method is based on a full sampling solution.

### 4.2 System Performance

We measure the *performance* in frames per second (FPS). The slowest method is the reference method with 2 FPS. The performance of the regular sampling methods rise in respect to the sampling spacing with a max of 6 FPS. Interleaved sampling methods range from 4 to 5 FPS. The caching method is the fastest one (8 FPS) because of its reduced workload, but suffers from pronounced stalls during RT updates.

### 4.3 Visual Quality

We measured three different qualities: *image differences* and two perceptual visual metrics *visibility prediction* and *quality predictions* [7]. Overall regular sampling with edge based anti aliasing created the best visual results. The worst rendering quality resulted from interleaved sampling in image space because of re-projection errors. As expected, interleaved sampling in visibility space created slightly better results. Caching created acceptable results only with a static camera.

### 4.4 Light Estimation Quality

For this metric, we compare the dominant light direction vectors extracted from the light estimation. The results are relatively stable for all methods, except for the intermediate sampling method in image space, which shows a slightly larger divergence from the reference method.

## 5 CONCLUSION AND FUTURE WORK

The preliminary results of our evaluation show that our discussed methods are relevant for improving the performance of probe-less photometric registration. However, so far we did not achieve convincing performance gains, mostly due to many additional small bottlenecks (GPU memory management) in the system. In the future, we are going to reduce the system overhead and improve and benchmark the acceleration methods.

## REFERENCES

- [1] P. Debevec, A. Wenger, C. Tchou, A. Gardner, J. Waese, and T. Hawkins. A lighting reproduction approach to Live-Action compositing. Technical report, July 2002.
- [2] A. Fournier, A. S. Gunawan, and C. Romanzin. Common illumination between real and computer generated scenes. In *Proc. Graphics Interface*, 1993.
- [3] L. Gruber, T. Richter-Trummer, and D. Schmalstieg. Real-time photometric registration from arbitrary geometry. In *2012 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 119–128, Nov. 2012.
- [4] R. Herzog, E. Eisemann, K. Myszkowski, and H.-P. Seidel. Spatio-temporal upsampling on the gpu. In *Proceedings of ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games (I3D) 2010*. ACM, 2010.
- [5] J. Jimenez, J. I. Echevarria, T. Sousa, and D. Gutierrez. Smaa: Enhanced morphological antialiasing. *Computer Graphics Forum (Proc. EUROGRAPHICS 2012)*, 31(2), 2012.
- [6] M. Knecht, C. Traxler, O. Mattausch, W. Purgathofer, and M. Wimmer. Differential instant radiosity for mixed reality. In *2010 9th IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 99–107. IEEE, Oct. 2010.
- [7] R. Mantiuk, K. J. Kim, A. G. Rempel, and W. Heidrich. HDR-VDP-2: a calibrated visual metric for visibility and quality predictions in all luminance conditions. *ACM Trans. Graph.*, 30(4):40:14:0, July 2011.