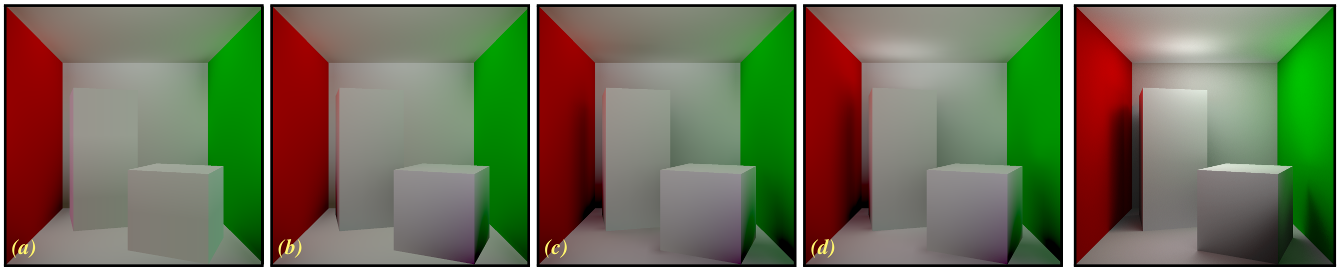
Delta Radiance Transfer

Bradford J. Loos[1,2] Derek Nowrouzezahrai[3] Wojciech Jarosz[3] Peter-Pike Sloan[1]
[1]Disney Interactive Studios [2]University of Utah [3]Disney Research Zurich



Standard MRT (indirect-only: 1.7ms) + Interreflections onto the clutter from the walls + Indirect shadows from the clutter onto the walls + Interreflections onto walls from clutter (1.25ms total) ≈ Ground Truth

Figure 1: *Modular Radiance Transfer (a) does not accurately model light scattered onto "clutter objects" (e.g. the two boxes) (b) or indirect shadows (c) and interreflections (d) from clutter onto the scene. We improve accuracy by adding these effects, and at little cost to performance.*

Abstract

Modular Radiance Transfer (MRT) is a recent technique for computing approximate direct-to-indirect transport. Scenes are dynamically constructed by warping and connecting simple shapes and compact transport operators are only precomputed on these simple shapes. MRT ignores fine-scale transport from "clutter" objects inside the scene, and computes light transport with reduced dimensional operators, which allows extremely high performance but can lead to significant approximation error. We present several techniques to alleviate this limitation, allowing the light transport from clutter in a scene to be accounted for. We derive additional low-rank *delta* operators to compensate for these missing light transport paths by modeling indirect shadows and interreflections from, and onto, clutter objects in the scene. We retain MRT's scene-independent precomputation and augment its scene-dependent initialization with clutter transport generation, resulting in increased accuracy without a performance penalty. Our implementation is simple, requiring a few small matrix-vector multiplications that generate a delta lightmap added to MRT's output, and does not adversely affect the performance benefits of the overall algorithm.

CR Categories: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Color, shading, shadowing, and texture;

Keywords: direct-to-indirect light, real-time global illumination

1 Introduction

Interactive graphics applications have started integrating approximate global illumination, often to satisfy art-driven requirements. Precomputation based approaches (e.g., direct-to-indirect transport) are capable of satisfying these requirements, however long preprocessing times limit their wider-scale adoption.

Recently, Loos et al. [2011] introduced *Modular Radiance Transfer* (MRT), a coarse-scale direct-to-indirect transport approach leveraging *scene-independent* precomputation. MRT aggregates indirect transport inside and between *simple shapes*, modeling how light is transported when shapes are warped and attached to each other. Level designers can transform and connect shapes to author new scenes or form lighting volumes inside existing scenes. This new style of light transport authoring eliminates time-consuming scene-wide precomputation.

While MRT models coarse-scale light transport within and between shapes, it ignores the effects of finer-scale "clutter geometry" (e.g., a pillar or desk in a room). Specifically, objects inside the simple shapes do not affect light transport at all: they do not cast indirect shadows, nor do they reflect indirect light onto the shape and its neighbors. MRT's generality precludes efficient incorporation of these effects: introducing sharp shadows/interreflections breaks many of MRT's assumptions about the nature of the light transport's dimensionality.

We introduce **Delta Radiance Transfer** (DRT) to carefully remove these constraints while maintaining the important benefits of MRT:

- **High performance:** maintaining the extremely high-performance of MRT is necessary to promote its applicability to content-generation pipelines;
- **Low-dimensional rendering:** in adding finer-scale occlusion and interreflection to MRT, we need to extend its low-dimensional rendering formulation to support this added complexity without incurring a substantial performance overhead;
- **Rendering extensions:** supporting dynamic vector irradiance and volume light probes in the presence of occlusion and (potentially near-field) interreflection enables compatibility with other common high-fidelity real-time rendering techniques.

While MRT models indirect light between large scene blocks, we introduce three new compact light transport operators to model the following transport paths missing from MRT (see Figure 2):

- indirect shadows from clutter onto the scene (Section 4.1),
- interreflections from clutter onto the scene (Section 4.2), and

- interreflections from the scene back onto clutter (Section 4.3).

In MRT and DRT, indirect light is computed as a weighted sum of dynamically generated lightmap textures. These textures are specially constructed to represent basis-space light transport. Each of our new operators are computed by ray-tracing against the lightmaps parameterized over the scene geometry, or the clutter, depending on the transport path being modeled.

2 Previous Work

Our work addresses limitations of MRT, and so we first overview the previous work that motivated MRT. We then discuss work that is most closely related to our novel contributions.

Direct-to-indirect approaches [Hasan et al. 2006] map direct lighting on the surfaces of a static scene to indirect lighting. During precomputation, the direct-to-indirect transport operator is constructed using e.g. ray-tracing and stored for evaluation at runtime. As we will discuss in Section 3, MRT represents a significant departure from traditional direct-to-indirect approaches: shifting precomputation from a scene-centric to a *scene-independent* step and willingly sacrificing accuracy in exchange for extremely high performance.

MRT also borrows and extends ideas from the precomputed radiance transfer (PRT) literature [Sloan et al. 2002; Lehtinen 2007] by representing light transport quantities and operators with basis-space expansions. Shadow field PRT approaches [Zhou et al. 2005; Iwasaki et al. 2007] store PRT vectors in volumetric grids around rigid objects and then couple transport between these objects and the surrounding scene. We also couple transport between the coarse-scale scene and clutter objects, however the transport coupling is computed on-the-fly directly in low-dimensional subspaces. Unlike PRT, MRT constructs data-driven bases (as opposed to using analytic bases) and introduces the idea of a *direct lighting prior* (Section 3.2). These ideas are similar in spirit to the local light precomputation used by Kristensen et al. [2005].

In DRT, we introduce an additional low-dimensional direct lighting space defined over clutter objects, as well as operators for casting shadows and interreflections from the clutter to the scene, and operators for casting indirect light from the scene back onto clutter.

Many approximate global illumination techniques perform all computations at run-time, without any precomputation [Wang et al. 2009]. Real-time ray-tracing approaches [Parker et al. 1999] are gaining popularity, although scalability and the high-performance requirements of interactive gaming engines still preclude the widespread adoption of this technology. Another popular set of techniques is *instant radiosity* approaches [Keller 1997] which trace and deposit light particles in a scene, and then compute direct lighting from these *virtual point lights* (VPLs) to approximate diffuse interreflections [Dachsbacher and Stamminger 2005; Dachsbacher and Stamminger 2006; Ritschel et al. 2008]. MRT and DRT target low-end graphics platforms such as the iPhone and iPad, where evaluating a single unshadowed point light takes at least 25 ms, precluding the feasibility of VPL techniques.

Screen-space approaches consider geometry that is only directly visible to the viewer, effectively treating the scene as a height field [Nowrouzezahrai and Snyder 2009], in order to accelerate computation [Nichols and Wyman 2009; Nichols et al. 2009]. Unlike MRT and DRT, these approaches introduce spatial and temporal artifacts in the case where the height field assumption is invalid, do not scale favorably to larger scenes, perform all computation in high-dimensional spaces without exploiting coherence in the underlying light transport operators, or do not meet the strict performance constraints of modern gaming engines.

More recently, Kaplanyan and Dachsbacher introduced Light Propagation Volumes (LPVs) [2010] which store and propagate radiance (represented using low-order spherical harmonics) in a discrete volume grid encompassing the scene, similarly to discrete ordinance (DO) techniques used in radiative transfer. Like VPL techniques, LPVs achieve good performance, do not require precomputation, and model smooth indirect shadows. However, LPVs benefit significantly from hardware accelerated computation on the GPU which is not available on lower-end platforms. Furthermore, they suffer from the same energy loss issues inherent in standard DO approaches and cannot model light propagation across large spatial distances, such as the maze scenes we use.

Our new transport operators are motivated by *antiradiance* and *implicit visibility* approaches [Dachsbacher et al. 2007; Dong et al. 2007]. These approaches iteratively compute global illumination without explicitly evaluating visibility. In order to capture direct and indirect shadows, *negative radiance* distributions are propagated along with standard radiance. We similarly propagate negative basis-space light to simulate indirect shadows (Section 4.1).

3 Background - Modular Radiance Transport

Given direct light in a scene, e.g. generated with shadow mapping, MRT computes coarse-scale dynamic indirect light, as well as dynamic vector and volumetric radiance to support high-frequency normal variation (e.g. normal maps) and limited shading of clutter.

MRT computes these effects with high-performance on a range of hardware platforms using the key ideas of *modularity* and *low-rank computation*: indirect light transport is decomposed into the effects *within* and *between* shapes, and these transport paths are computed and coupled entirely in optimized low-dimensional subspaces.

After giving a brief introduction to light transport using matrix operators, we will discuss the modularity and low-rank nature of MRT and then build on top of these concepts in Section 4.

3.1 Matrix Light Transport and Naïve SVD

Indirect light L_{ind} can be computed by applying a continuous linear operator to the direct light L_{d} in a scene as

$$L_{\text{ind}}(x) = \int_{\Omega_n} L_{\text{d}}(x', -\omega)\, f(x,\omega)\, (n_x \cdot \omega)\, d\omega = \mathcal{F}\{L_{\text{d}}\}(x),$$

where x is a point in the scene, n_x is the normal at x, Ω_n is the set of all unit direction vectors in the upper hemisphere about n_x, $x' = \mathbf{ray}(x + t\,\omega)$ is the nearest surface point from x in direction ω given by the ray-tracing operator **ray**, f is the BRDF at x, and $\mathcal{F}$ is the continuous one-bounce direct-to-indirect transport operator.

We assume diffuse relighting, where $\mathcal{F}$ can be discretized to yield the (discrete) direct-to-indirect transport equation: $\mathbf{l_{ind}} = \mathbf{F}\,\mathbf{l_d}$, where indirect light $\mathbf{l_{ind}}$ is computed by applying the (discrete) one-bounce operator $\mathbf{F}$ to the direct light $\mathbf{l_d}$. Each element of $\mathbf{l_{ind}}$ and $\mathbf{l_d}$ represents outgoing radiance at a different surface location.

Evaluating this discrete equation is expensive and limits the performance of direct-to-indirect transport since $\mathbf{F}$ grows proportionally with $\mathcal{O}(d^2)$, where d is the spatial discretization of the scene.

A common acceleration strategy is to take the singular value decomposition (SVD) of $\mathbf{F}$ and approximate the matrix-vector product using a rank-reduced $\mathbf{F} = \mathbf{U}_f\ \mathbf{\Sigma}_f\ \mathbf{V}_f^{\text{T}}$, where $\mathbf{U}_f$ and $\mathbf{V}_f^{\text{T}}$ contain the left and right singular vectors of $\mathbf{F}$, and $\mathbf{\Sigma}_f$ is a matrix with the singular values σ_i of $\mathbf{F}$ along its diagonal. The discrete transport equation can be approximated by keeping the r largest σ_i.

Unfortunately, as discussed in [Loos et al. 2011], the singular values of $\mathbf{F}$ fall-off too slowly to yield high-performance using this approximation technique. We will discuss how MRT accelerates evaluation of the (discrete) direct-to-indirect transport equation while inducing a more controlled degradation of accuracy.

3.2 Lighting Prior and Implicit Lights

MRT takes a unique approach to accelerating the (approximate) computation of $\mathbf{l_{ind}}$, computing light transport entirely in low-dimensional spaces, by exploiting two key observations:

1. *plausible* direct lighting in a scene lies in a low-dimensional, highly-correlated subspace of all input signals, and
2. applying $\mathbf{F}$ to these highly-correlated direct lighting signals yields highly-correlated indirect illumination patterns.

Using the SVD of $\mathbf{F}$ to accelerate the matrix-vector product does not account for possible correlations in the input (direct) light patterns, $\mathbf{l_d}$. This SVD is optimal if the $\mathbf{l_d}$s are drawn from *an arbitrary distribution*; however, in reality, they are drawn from the more restrictive set of *possible direct lighting signals*. This set has many correlations, leading to correlations in the resulting indirect light.

Lighting Prior. MRT precomputes direct illumination from a set of lights placed uniformly in the volume of a scene shape. By treating each direct light output as a column in a matrix and taking its SVD, the first n left singular vector columns $\mathbf{P}$ yield a low-dimensional basis for direct illumination, called the *lighting prior*.

Correlations in direct lighting cause the effective dimension n to be much lower than the explicit dimension (number of surface locations). To exploit this correlation when computing indirect lighting, MRT first defines $\mathbf{M} = \mathbf{F}\,\mathbf{P}\,\mathbf{S}$, where $\mathbf{S}$ is the diagonal singular value matrix associated with $\mathbf{P}$. Taking the SVD of $\mathbf{M} = \mathbf{U}\,\mathbf{\Sigma}\mathbf{V}^{\mathrm{T}}$, MRT's low-dimensional direct-to-indirect transport equation is

$$\mathbf{l_{ind}} \approx \mathbf{U_b}\,\mathbf{T_{d\rightarrow b}}\,\mathbf{l_d} = \mathbf{U_b}\,\mathbf{b}\,, \qquad (1)$$

where $\mathbf{U_b} = \mathbf{U}\,\mathbf{\Sigma}$ and $\mathbf{T_{d\rightarrow b}} = \mathbf{V}^{\mathrm{T}}\,\mathbf{S}^{-1}\,\mathbf{P}^{\mathrm{T}}$ projects[1] direct light $\mathbf{l_d}$ to the correlated low-rank indirect light space (with corresponding coefficients $\mathbf{b}$). Despite the derivations outlined above, MRT simply precomputes $\mathbf{U_b}$ and $\mathbf{T_{d\rightarrow b}}$ (in a few seconds), and all run-time computations are performed in the low-dimensional spaces.

Implicit Lighting. The columns of $\mathbf{U_b}$ are indirect *basis light patterns*. Once $\mathbf{l_d}$ is reduced to $\mathbf{b}$ with $\mathbf{T_{d\rightarrow b}}$, Equation 1 scales these patterns by the elements of $\mathbf{b}$ to yield dynamic indirect light.

MRT exploits an alternative approach for generating the columns of $\mathbf{U_b}$. Namely, basis lighting patterns result from the application of the one-bounce operator to a set of *implicit lighting patterns*, $\mathbf{U_b} = \mathbf{F}\,\mathbf{L_{imp}}$, where $\mathbf{L_{imp}} = \mathbf{P}\,\mathbf{S}\,\mathbf{V}$.

DRT also uses $\mathbf{L_{imp}}$ to construct operators which model indirect shadows from clutter and interreflections *from* the scene *onto* clutter (Sections 4.1 and 4.3). DRT computes higher-order radiance representations (e.g., for normal mapping) and radiance volumes with $\mathbf{L_{imp}}$, incorporating the occlusion/interreflection effects of clutter.

3.3 Inter- and Intra-shape Transport

MRT allows artists to map simple geometric primitives (e.g., cubes with a number of faces removed) to existing scenes or to build new scenes with them. Given a set of these (potentially warped) connected shapes, resulting from the content creation process, MRT dynamically updates indirect lighting in a modular fashion.

First, direct-to-indirect transport is computed *within* each shape as described in Section 3.2. Next, direct-to-indirect transport is propagated from each shape to all others. To do so, at level creation time MRT quickly computes a shape connectivity graph and, for each shape, concatenates light-transport operators which progressively propagate the lighting to all affected shapes. A handful of low-dimensional operators, precomputed for each *interface* of a shape, map $\mathbf{b}$ coefficients for a shape to a low-dimensional lightfield at the interface. Additional precomputed operators map basis-radiance at the lightfields back onto the surfaces of adjacent shapes.

DRT additionally models the effects of finer-scale occlusions and interreflections (Section 4), introducing *delta occlusion operators*, *delta reflection operators*, and *clutter gather operators* (Sections 4.1, 4.2 and 4.3) to update block and clutter lightmaps. The later stages of our approach either output lightmaps over the clutter objects $\mathbf{l_c}$, or directly update the scene lightmap $\mathbf{l_{ind}}$.

4 Indirect Occlusions and Interreflections

Given an empty scene composed of blocks with implicit lighting environments $\mathbf{L_{imp}}$ (computed with MRT), as well as the clutter geometry for the scene, we define new low-dimensional operators to capture the indirect shadows and interreflections from the additional clutter objects. We also need new operators to bounce light from the scene back onto the clutter. MRT uses volume samples for this last scene-to-clutter lighting, whereas we will more accurately and explicitly model this transport with *clutter gather operators* (Section 4.3). Our scene-dependent computation also increases performance (see Section 5).

A brute-force solution is to recompute direct light in the scene (including clutter) several times, regenerate an updated lighting prior $\mathbf{P}$, and finally regenerate the scene's $\mathbf{U_b}$ operator.

This approach is unnecessarily complex, expensive, and ineffective: direct light on a scene with clutter has high-frequency shadows from the clutter, requiring more bases (a higher n) in the lighting prior $\mathbf{P}$, for accurate results. This would reduce MRT's performance and generality ("scene-independent" shapes would have to include clutter). However, the resulting low-frequency indirect occlusions and interreflections motivate an alternative approach.

The operators we define below are applied to clutter at each block, however the resulting transport is propagated beyond the current block to all blocks within a pre-described neighborhood.

4.1 Scene Occlusions with Delta Occlusion Operators

We first consider indirect shadows *from* the clutter *onto* the scene.

After indirect lighting is computed on the scene surfaces using MRT (ignoring the effects of clutter), we subtract occlusion from this unshadowed shading similarly to antiradiance [Dachsbacher et al. 2007], while operating entirely in the existing low-dimensional subspace. To do so, we rely on the existing implicit lighting to construct the necessary light transport operator, as discussed below.

Recall that applying the direct-to-indirect operator $\mathbf{F}$ to the implicit lighting $\mathbf{L_{imp}}$ yields the $\mathbf{U_b}$ operator. We require an operator, $\mathbf{\Delta U_b}$, to capture (negative) basis light due to clutter occlusions on the scene surfaces.

We compute $\mathbf{\Delta U_b}$ directly using $\mathbf{L_{imp}}$, meaning we do not explicitly recompute direct-to-indirect transport: at each $\mathbf{\Delta U_b}$ texel

[1] We use rank-reduced matrices where necessary, e.g. $\mathbf{V}^{\mathrm{T}}$.

in the scene, we shoot many uniformly distributed shadow rays. Each ray that intersects the clutter corresponds to a direction that will occlude light. We trace the intersected ray beyond the clutter until it hits a surface in the scene, where it is weighted by the $\mathbf{L_{imp}}$ basis function value associated with the texel at the surface location. Rays that do not hit the clutter clearly do not contribute to indirect occlusion and are ignored. The average of all ray values yields the $\mathbf{\Delta U_b}$ entry for the currently processed texel (Figure 2, left).

Conceptually, the columns of $\mathbf{\Delta U_b}$ are indirect *shadowed basis light patterns*, and we compute shading *including* indirect clutter shadows using $\overline{\mathbf{U_b}} = \mathbf{U_b} + \mathbf{\Delta U_b}$ instead of $\mathbf{U_b}$. This amounts to adding the negative light necessary to account for the indirect shadows from the clutter.

In practice, we need to address spatial sampling issues arising from resolution mismatches and misalignments between clutter and scene lightmaps. We discuss this issue in Section 5, as well as our proposed inpainting solution (Figure 4).

4.2 Interreflections with Delta Reflection Operators

Next, we consider interreflections *from* the clutter *onto* the scene.

We note that all the operators we have introduced so far (e.g., $\overline{\mathbf{U_b}}$) are not parameterized over the clutter geometry: $\mathbf{b}$ coefficients are only computed from $\mathbf{l_d}$ defined over the scene's surfaces. For example, in the case of indirect shadows from clutter (Section 4.1), this shadowing only depends on the *presence* of the clutter geometry, not on the value of lighting over its surface. As such, the indirect shadows are dependent (in a parametric sense, as opposed to a geometric one) on the $\mathbf{b}$ coefficients.

This is not the case when computing direct-to-indirect (basis-space) transport from the clutter onto the scene, since we must derive new operators to map direct illumination *over the surfaces of the clutter* to interreflected light *onto the scene*. Here, the mere presence of clutter is not sufficient to parameterize this lighting effect.

As with direct lighting on the scene, direct light on the clutter is computed in a lightmap and projected into a low-rank basis. We found that using a simple orthonormal basis (constant radiance over each clutter mesh face) was sufficient for our examples, but for more complex clutter we could construct a lighting prior to obtain an optimal basis (as in MRT). In this simplified case, $\mathbf{T_{d\rightarrow c}}$ is our $m \times f$ transformation matrix that projects direct light into this basis ($\mathbf{c}$ coefficients), where m is the lightmap resolution and f is the number of clutter mesh faces.

Given this basis over the clutter, we precompute indirect lighting responses *on the scene* to any signal represented in the basis (e.g., direct light) *over the clutter*: at each texel in the scene lightmap, we shoot many gather rays and intersect against the nearby clutter. These intersected rays correspond to directions which will bounce light from the clutter onto that surface location on the scene.

Each gather ray that intersects a clutter object samples the clutter lighting basis and accumulates diffuse (or vector valued) basis-lighting into our $\mathbf{\Delta U_c}$ operator (see Figure 2, center). This transport accumulation approach is similar to multi-bounce PRT transfer computation [Sloan et al. 2002].

At runtime we compute direct light $\mathbf{l_d}$ in the clutter lightmap and project it into the reduced basis. The resulting $\mathbf{c}$ coefficients are used to scale interreflection response textures (columns of $\mathbf{\Delta U_c}$) to yield bounced light from the clutter onto the scene.

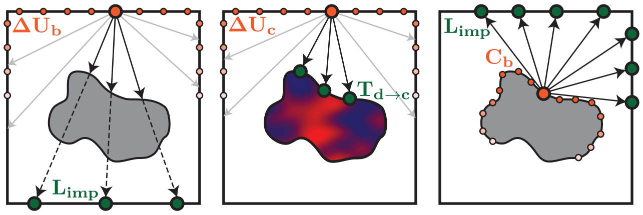


Figure 2: *The three additional transfer operators we compute for clutter. From left to right: occlusions onto scene (anti-radiance), diffuse reflections onto scene, and diffuse reflections onto clutter.*

4.3 Interreflections with Clutter Gather Operator

Lastly, we model interreflections *from* the scene *onto* the clutter. Our solution replaces the *volume samples* from standard MRT and also supports clutter shadowing and higher sampling densities.

Similarly to delta occlusion maps, we define a $\mathbf{C_b}$ operator that maps $\mathbf{b}$ to indirect light (in a lightmap) over the clutter. The bounced light clearly depends on the scene's $\mathbf{b}$ coefficients, but the output space is defined over the surface of the clutter.

For each row (texel) of $\mathbf{C_b}$, we gather light by tracing rays in all directions from the current clutter surface point, averaging the rows of $\mathbf{L_{imp}}$ associated with each ray hit locations on the scene surface (Figure 2, right). As in Section 4.1, rays that do not intersect the scene do not contribute bounced light.

5 Implementation Details

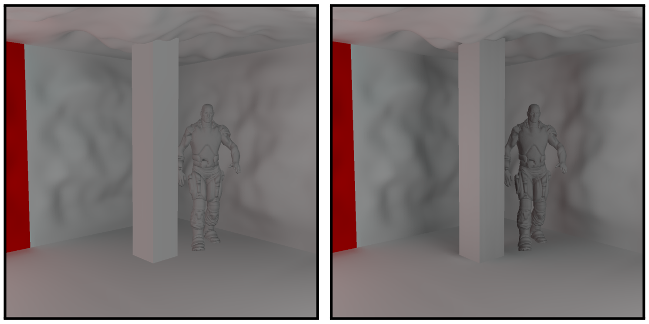
Our simple runtime requires only a handful of small matrix-vector multiplies. Our additions to the standard MRT runtime are:

- Add indirect shadows to the scene's $\mathbf{l_{ind}}$ due to clutter, using the *delta occlusion operator*: $\mathbf{\Delta U_b}\,\mathbf{b}$,
- Compute clutter $\mathbf{l_d}$ and project to $\mathbf{c}$ with $\mathbf{T_{d\rightarrow c}}$,
- Add interreflections *to the scene's* $\mathbf{l_{ind}}$ from clutter: $\mathbf{\Delta U_c}\,\mathbf{c}$,
- Compute interreflections *from* the scene *onto* the clutter geometry: $\mathbf{l_c} = \mathbf{C_b}\,\mathbf{b}$.

New response textures, computed using our operators, model changes due to occlusions/interreflections to the base shading (computed with standard MRT). We simply blend these intermediate textures into either the scene ($\mathbf{l_{ind}}$) or clutter lightmaps ($\mathbf{l_c}$). We outline the end-to-end algorithm below, as well as details that need to be considered during data generation.

End-to-end Algorithm. Direct illumination is first computed using any standard approach; we use shadow mapping. Notably, direct illumination must be explicitly computed or mapped to the spatially sub-sampled surface locations that are used to parameterize the $\mathbf{l_d}$ input vector. Sub-sampled direct illumination is mapped to low-dimensional indirect lighting coefficients $\mathbf{b}$, which will drive the remainder of the MRT components of the algorithm, as well as DRT's indirect shadows from clutter onto the scene and interreflections from the scene onto clutter.

For each block, indirect lighting is computed, ignoring clutter geometry, using Equation 1. Light between each block (still ignoring clutter) is propagated using low-dimensional lightfield propagation operators [Loos et al. 2011]. MRT also pays careful attention to compute padding regions in the output lightmaps to avoid visible seems when blocks are connected together.



Modular Radiance Transfer (1.5ms) **Delta Radiance Transfer (2.1ms)**

Figure 3: *MRT vs. DRT: note that the dynamic soldier object is also shadowed/lit by the clutter/scene, using volume probes. Timings exclude direct lighting which took 8.3 ms.*

At this point, DRT computes the $\mathbf{\Delta U_b}$, $\mathbf{\Delta U_c}$, and $\mathbf{C_b}$ transport operators, as well as the $\mathbf{T_{d\rightarrow c}}$ basis projection matrix.

Indirect shadows from the clutter onto the scene are computed as $\mathbf{\Delta U_b\, b}$, after which we compute (spatially sub-sampled) direct illumination on the clutter and project it onto the direct light subspace using $\mathbf{T_{d\rightarrow c}}$, yielding $\mathbf{c}$ coefficients.

Indirect bounced light between the scene and clutter are computed in the last stages of DRT. First, interreflections from the clutter onto the scene are computed as $\mathbf{\Delta U_c\, c}$, and then interreflections from the scene onto the clutter are computed as $\mathbf{C_b\, b}$.

Transport Inpainting. $\mathbf{\Delta U_b}$ texels corresponding to scene locations *inside* clutter will have all their rays intersecting a clutter object, and the resulting transport will be incorrect[2]. In order to populate these texels with meaningful transport entries, we average transport from all valid (e.g., not inside clutter) neighboring texels.

Note that, when inpainting from valid neighbor entries, we average $\overline{\mathbf{U_b}}$ entries and then subtract the unshadowed $\mathbf{U_b}$ of the destination texel, instead of averaging $\mathbf{\Delta U_b}$ entries. Other inpainting methods, e.g. gradient interpolation schemes, are also suitable, however our simpler approach yields pleasing results (Figure 4). It is possible that inpainting can fail if there are no samples with valid $\mathbf{\Delta U_b}$ entries nearby (e.g. with high frequency geometry), but we did not encounter this limitation in our test scenes.

Vector and Volume Response. We also compute *vector-valued* $\mathbf{\Delta U_b}$, $\mathbf{\Delta U_c}$ and $\mathbf{C_b}$ operators ($\mathbf{\Delta U}_{\vec{b}}$, $\mathbf{\Delta U}_{\vec{c}}$ and $\mathbf{C}_{\vec{b}}$) to support high-frequency surface details with normal maps. We do so by projecting basis-space radiance into spherical harmonics (SH) and mapping the SH coefficients into MRT's hemispherical basis.

Moreover, we compute a new operator $\mathbf{U_{dyn}}$ to map $\mathbf{b}$ **and** $\mathbf{c}$ vectors to volumetric SH radiance probes. We do so by combining the clutter and scene gathers in Section 4 and outputting into a volume texture. These dynamic light probes are applied to animated objects (e.g. characters) in the scene, capturing shadows and interreflection from the clutter and scene at minimal performance cost (Figure 3).

MRT uses volume probes to shade clutter, but this only captures a subset of the transport (light bounced from the scene onto clutter) and introduces error (Figure 1a). Our *clutter gather operator* computes this transport path more accurately (Figure 1b) and with higher-performance, since radiance is computed directly in the lightmap as opposed to using an SH volume probe. For example, Figure 1a uses MRT volume probes and Figures 1b-d do not: DRT *outperforms* MRT in this case as volume texture sampling and SH shading are avoided.

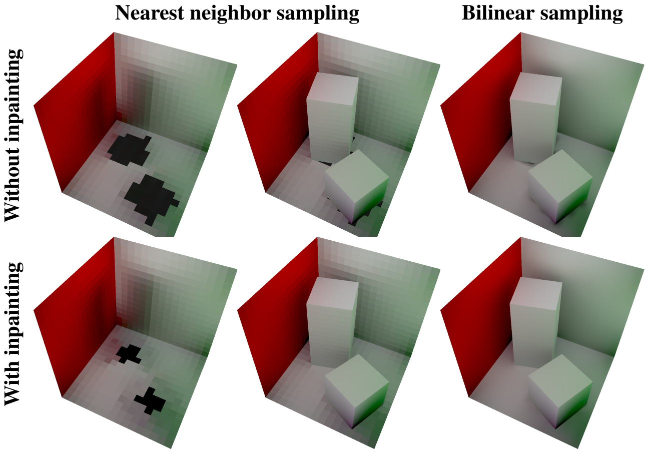


Figure 4: *We inpaint regions inside clutter, ensuring smooth shading around objects that do not align with the scene lightmap texels.*

Settings. Our results use the following settings: 1024 rays per texel to construct our operators, and 16 × 16 lightmaps on each shape or clutter face. Our runtime data requirements are similar to MRT: delta occlusion operators (Section 4.1) and delta reflection operators (Section 4.2) use 32×16^2 values per face (32 modes for 256 texels). Clutter gather operators (Section 4.3) use fewer modes, requiring 5×16^2 values per clutter face. We illustrate DRT on simple clutter geometry (five-faced pillars), but we can handle arbitrary clutter objects as the position and normal samples required for the ray tracing are generated by rasterizing clutter into UV atlases.

Even in a complex scene with four pillars (partially visualized in Figure 3), all of our new operators require only 0.52MB total additional storage and the application of our new operators has a negligible cost on the overall shading, especially when direct lighting computation (using shadow mapping) is included. Updating the operators requires roughly 5 seconds on a 12-core Intel Xeon X5670. We also naïvely re-raytrace the same scene shapes to account for distant light transport, leading to redundant raytracing. Optimized ray-tracing is left to future work as the additional latency is negligible for e.g. level design use cases.

6 Discussion

Advantages. With DRT, we simulate important light transport paths ignored in MRT, allowing clutter geometry to contribute to indirect illumination in the scene. This increases the accuracy of the approach, but maintains the high-performance behavior of the overall algorithm. We avoid brute-force computation of full light transport by exploiting low-dimensional basis-space lighting parameterized over the scene and clutter, which we use to efficiently construct our low-rank operators. Once clutter geometry is placed or moved in the scene, DRT can compute the additional transport operators in just a few seconds (using unoptimized CPU ray-tracing), allowing for fast scene design and shading response. At any time, the direct lighting can be changed with the indirect illumination updated immediately. As with MRT, DRT can easily scale to low-end graphics platforms such as the iPad and iPhone.

[2] Consider Figure 2 (left) with an orange sample inside the grey region.

Disadvantages. DRT introduces scene-dependent computation and data to the original MRT framework, which can be viewed as a disadvantage, however this added flexibility allows clutter geometry to be added to a scene at run-time with light transport updated in only a few seconds. As discussed in [Loos et al. 2011], adding more items to the library of precomputed shapes (e.g. to model clutter geometry) would increase the entropy of the various transport operators, precluding accurate low-rank approximations.

Given that we use the implicit lighting environment generated *without* the clutter present in the scene, indirect light reflected onto the clutter from the scene can suffer from artifacts. The closest box in Figure 1 exhibits this subtle artifact with incorrect shadow colors. Given the scene-independent precomputed lightmap bases from MRT, DRT's approach still computes the most suitable approximation to these basis-space operators. A fundamentally different approach would be necessary to completely eliminate these artifacts, however it is unclear if such an approach would still fit into the scene-independent framework of MRT.

Lastly, DRT shares the limitations of MRT: transport that is not modeled in the light prior is not supported and we clearly only handle low-frequency light transport effects.

7 Conclusions and Future Work

We extend MRT to more accurately handle large clutter objects, reducing approximation error (see Figure 1) at a negligible cost to performance and memory. We dynamically compute light transport operators to model indirect shadows and interreflections from the clutter onto the scene, as well as interreflections from the scene back onto the clutter. A handful of basis lightmaps are generated on-the-fly and parameterized over the scene and clutter, and raytracing directly against these basis textures allows for rapid construction of our additional low-rank transport operators. This direct reduced-dimensional operator construction is efficient, allowing these new scene-dependent operators to complement MRT's scene-independent precomputation.

There are many approximations in DRT which could possibly be improved. Using more modes to represent the direct light on the clutter would increase the quality of the interreflections (i.e. difference in the highlight on the ceiling in Figure 1). The implicit lighting could also possibly be extended to more accurately model shadows, which would increase the quality of the $\mathbf{\Delta U_b}$ operator.

While only a single $\mathbf{U_b}$ is needed, our choice of $\mathbf{T_{d\rightarrow c}}$ mandates separate $\mathbf{T_{d\rightarrow c}}/\mathbf{\Delta U_c}$ pairs for each lightmapped clutter object. A tailored low-rank basis for direct light (and associated $\mathbf{\Delta U_c}$) over the clutter, as mentioned earlier, could reduce these costs.

We also plan to investigate more careful sampling and reconstruction techniques for our inpainting solution [Kavan et al. 2011]. Moreover, we use clutter-aware operators to generate volume samples for dynamic object in the scene, but more faithful transport coupling for dynamic objects is necessary for higher-fidelity results.

References

DACHSBACHER, C., AND STAMMINGER, M. 2005. Reflective shadow maps. In *ACM Symposium on Interactive 3D Graphics and Games*.

DACHSBACHER, C., AND STAMMINGER, M. 2006. Splatting indirect illumination. In *ACM Symposium on Intearactive 3D Graphics and Games*.

DACHSBACHER, C., STAMMINGER, M., DRETTAKIS, G., AND DURAND, F. 2007. Implicit visibility and antiradiance for interactive global illumination. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 26*, 3 (July), 61:1–61:10.

DONG, Z., KAUTZ, J., THEOBALT, C., AND SEIDEL, H.-P. 2007. Interactive global illumination using implicit visibility. In *Pacific Conference on Computer Graphics and Applications*, IEEE Computer Society, Washington, DC, USA.

HASAN, M., PELLACINI, F., AND BALA, K. 2006. Direct-to-indirect transfer for cinematic relighting. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 25*, 3 (July), 1089–1097.

IWASAKI, K., DOBASHI, Y., YOSHIMOTO, F., AND NISHITA, T. 2007. Precomputed radiance transfer for dynamic scenes taking into account light interreflection. In *Rendering Techniques 2007: 18th Eurographics Workshop on Rendering*, 35–44.

KAPLANYAN, A., AND DACHSBACHER, C. 2010. Cascaded light propagation volumes for real-time indirect illumination. In *ACM Symposium on Interactive 3D Graphics and Games*.

KAVAN, L., BARGTEIL, A. W., AND SLOAN, P.-P. 2011. Least squares vertex baking. *Computer Graphics Forum (Proceedings of EGSR 2011) 30*, 4, 1319–1326.

KELLER, A. 1997. Instant radiosity. In *SIGGRAPH*.

KRISTENSEN, A. W., AKENINE-MÖLLER, T., AND JENSEN, H. W. 2005. Precomputed local radiance transfer for real-time lighting design. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 24*, 3 (Aug.), 1208–1215.

LEHTINEN, J. 2007. A framework for precomputed and captured light transport. *ACM Transactions on Graphics 26*, 4 (Oct.).

LOOS, B. J., ANTANI, L., MITCHELL, K., NOWROUZEZAHRAI, D., JAROSZ, W., AND SLOAN, P.-P. 2011. Modular radiance transfer. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia) 30*, 6 (Dec.). accepted for publication.

NICHOLS, G., AND WYMAN, C. 2009. Multiresolution splatting for indirect illumination. In *ACM Symposium on Interactive 3D Graphics and Games*.

NICHOLS, G., SHOPF, J., AND WYMAN, C. 2009. Hierarchical image-space radiosity for interactive global illumination. *Computer Graphics Forum 28*, 4.

NOWROUZEZAHRAI, D., AND SNYDER, J. 2009. Fast global illumination of dynamic height fields. *Computer Graphics Forum 28*, 4.

PARKER, S., MARTIN, W., SLOAN, P.-P. J., SHIRLEY, P., SMITS, B., AND HANSEN, C. 1999. Interactive ray tracing. In *ACM Symposium on Interactive 3D Graphics*.

RITSCHEL, T., GROSCH, T., KIM, M. H., SEIDEL, H.-P., DACHSBACHER, C., AND KAUTZ, J. 2008. Imperfect shadow maps for efficient computation of indirect illumination. *ACM Trans. Graph.*.

SLOAN, P.-P., KAUTZ, J., AND SNYDER, J. 2002. Precomputed radiance transfer for real-time rendering in dynamic, low-frequency lighting environments. *ACM Transactions on Graphics (Proceedings of SIGGRAPH) 21*, 3 (July), 527–536.

WANG, R., WANG, R., ZHOU, K., PAN, M., AND BAO, H. 2009. An efficient gpu-based approach for interactive global illumination. *ACM Trans. Graph. 28*, 3.

ZHOU, K., HU, Y., LIN, S., GUO, B., AND SHUM, H.-Y. 2005. Precomputed shadow fields for dynamic scenes. In *ACM SIGGRAPH 2005 Papers*, ACM, New York, NY, USA, SIGGRAPH '05, 1196–1201.