Overview

... follows the structure of the STAR

► Introduction & Welcome (Carsten)
► Many-Light Rendering Concepts (Jan)
  ► Basic Idea
  ► Improved Virtual Lights Generation
  ► Lighting with Virtual Lights
► Really Many Lights: Scalability (Carsten)
► Interactive and Real-Time Rendering (Carsten)
► Conclusions, Outlook, Q&A (Jan & Carsten)
Many-light Methods

Two major passes

Generation of VPLs

Lighting with VPLs
Many-light Methods

Pass 1: Generation of VPLs
Many-light Methods

Pass 1: Generation of VPLs
Many-light Methods

Pass 1: Generation of VPLs
Many-light Methods

Pass 1: Generation of VPLs
Many-light Methods

Pass 1: Generation of VPLs
Many-light Methods

Pass 1: Generation of VPLs
Many-light Methods

Pass 1: Generation of VPLs
Many-light Methods

Pass 1: Generation of VPLs
Many-light Methods

Pass 1: Generation of VPLs
Many-light Methods

Pass 1: Generation of VPLs
Many-light Methods

Pass 1: Generation of VPLs
Many-light Methods

Pass 1: Generation of VPLs
Many-light Methods

Pass 1: Generation of VPLs
Many-light Methods

Pass 1: Generation of VPLs
Many-light Methods

Pass 1: Generation of VPLs
Many-light Methods

Pass 2: Lighting with VPLs
Many-light Methods

Pass 2: Lighting with VPLs
Many-light Methods

Pass 2: Lighting with VPLs
Many-light Methods

Generation of VPLs

Lighting with VPLs
Many-light Methods

Generation of VPLs

Lighting with VPLs
Naive Generation of VPLs

Problem #1 (*in complex scenes*):

- many VPLs do not contribute
Naive Generation of VPLs

Problem #1 (*in complex scenes*):
- many VPLs do not contribute
Naive Generation of VPLs

Problem #1 (*in complex scenes*):

- many VPLs do not contribute
Naive Generation of VPLs

Problem #1 (*in complex scenes*):

- many VPLs do not contribute
Naive Generation of VPLs

Problem #1 (*in complex scenes*): 
- many VPLs do not contribute
Naive Generation of VPLs

Problem #1 *(in complex scenes)*:

- many VPLs do not contribute
Naive Generation of VPLs

Problem #1 (in complex scenes):
- many VPLs do not contribute

Scene

Instant radiosity

Reference

Images courtesy of Segovia et al.
Naive Generation of VPLs
Naive Generation of VPLs

Problem #2 (*in glossy scenes*):
- insufficient number of VPLs in certain areas
- glossy inter-reflections suffer from splotches
Naive Generation of VPLs

Problem #2 (in glossy scenes):

- insufficient number of VPLs in certain areas
- glossy inter-reflections suffer from splotches

Instant radiosity
Naive Generation of VPLs

Problem #2 (in glossy scenes):
- insufficient number of VPLs in certain areas
- glossy inter-reflections suffer from splotches

Instant radiosity

Clamped

missing inter-reflections
Naive Generation of VPLs

Problem #2 (in glossy scenes):

- insufficient number of VPLs in certain areas
- glossy inter-reflections suffer from splotches

Instant radiosity  Clamped  Reference

missing inter-reflections

Images courtesy of Davidović et al.
Naive Generation of VPLs

Problem #2 (*in glossy scenes)*:
- insufficient number of VPLs in certain areas
- glossy inter-reflections suffer from splotches

Instant radiosity

Reference

we need more VPLs!

Images courtesy of Davidovič et al.
Improved Generation of VPLs
Improved Generation of VPLs

Goal:

- place VPLs only where needed
Improved Generation of VPLs

Goal:
- place VPLs only where needed

Approaches:
- Rejection of unimportant VPLs  [Georgiev and Slusallek 2010]
- Bidirectional Instant Radiosity  [Segovia et al. 2006]
- Metropolis sampling for VPL distributions  [Segovia et al. 2007]
- Local Virtual Lights  [Davidovič et al. 2010]
Improved Generation of VPLs
Improved Generation of VPLs

Rejection of unimportant VPLs  [Georgiev and Slusallek 2010]
Improved Generation of VPLs

Rejection of unimportant VPLs  [Georgiev and Slusallek 2010]

Idea:

- probabilistically reject VPLs with low expected contribution
Improved Generation of VPLs

Rejection of unimportant VPLs  [Georgiev and Slusallek 2010]

Idea:

- probabilistically reject VPLs with low expected contribution
Improved Generation of VPLs

Rejection of unimportant VPLs  [Georgiev and Slusallek 2010]

Idea:

- probabilistically reject VPLs with low expected contribution
Improved Generation of VPLs

Rejection of unimportant VPLs  [Georgiev and Slusallek 2010]

Approach:
Improved Generation of VPLs

Rejection of unimportant VPLs  [Georgiev and Slusallek 2010]

Approach:
1. estimate average contribution of a VPL $\Phi_v$
Improved Generation of VPLs

Rejection of unimportant VPLs  [Georgiev and Slusallek 2010]

Approach:
1. estimate average contribution of a VPL $\Phi_v$
   - few pilot VPLs illuminate few surface points seen by the camera
Improved Generation of VPLs

Rejection of unimportant VPLs [Georgiev and Slusallek 2010]

Approach:

1. estimate average contribution of a VPL $\Phi_v$
   - few pilot VPLs illuminate few surface points seen by the camera

2. generate VPLs
   - for each VPL $i$
     - estimate its contribution $\Phi_i$ to points seen by the camera
Improved Generation of VPLs

Rejection of unimportant VPLs  [Georgiev and Slusallek 2010]

Approach:
1. estimate average contribution of a VPL $\Phi_v$
   - few pilot VPLs illuminate few surface points seen by the camera

2. generate VPLs
   - for each VPL $i$
     - estimate its contribution $\Phi_i$ to points seen by the camera
     - accept with probability $p_i = \min \left( \frac{\Phi_i}{\Phi_v} + \epsilon, 1 \right)$
Improved Generation of VPLs

Rejection of unimportant VPLs [Georgiev and Slusallek 2010]

Approach:
1. estimate average contribution of a VPL $\Phi_v$
   - few pilot VPLs illuminate few surface points seen by the camera

2. generate VPLs
   - for each VPL $i$
     - estimate its contribution $\Phi_i$ to points seen by the camera
     - accept with probability $p_i = \min \left( \frac{\Phi_i}{\Phi_v} + \epsilon, 1 \right)$
     - if accepted, divide its energy by $p_i$
Improved Generation of VPLs

Rejection of unimportant VPLs  [Georgiev and Slusallek 2010]

Approach:
1. estimate average contribution of a VPL $\Phi_v$
   - few pilot VPLs illuminate few surface points seen by the camera

2. generate VPLs
   - for each VPL $i$
     - estimate its contribution $\Phi_i$ to points seen by the camera
     - accept with probability $p_i = \min \left( \frac{\Phi_i}{\Phi_v} + \epsilon, 1 \right)$
     - if accepted, divide its energy by $p_i$
Improved Generation of VPLs

Rejection of unimportant VPLs [Georgiev and Slusallek 2010]

Advantages:

- cheap and simple to implement!
- VPLs have roughly equal contribution
Improved Generation of VPLs

Rejection of unimportant VPLs [Georgiev and Slusallek 2010]

Advantages:
- cheap and simple to implement!
- VPLs have roughly equal contribution
- works well most of the time
Improved Generation of VPLs

Rejection of unimportant VPLs  [Georgiev and Slusallek 2010]

Advantages:
▶ cheap and simple to implement!
▶ VPLs have roughly equal contribution
▶ works well most of the time

Disadvantages:
▶ increases the cost of VPL distribution
▶ “one-pixel image” assumption
▶ does not help with local inter-reflections
Improved Generation of VPLs

Rejection of unimportant VPLs  [Georgiev and Slusallek 2010]

Without rejection  

With rejection (7% acceptance)  

Images courtesy of Georgiev and Slusallek
Improved Generation of VPLs
Improved Generation of VPLs

Bidirectional sampling for VPL distributions [Segovia et al. 2006]
aka “Bidirectional Instant Radiosity”
Improved Generation of VPLs

Bidirectional sampling for VPL distributions  [Segovia et al. 2006]
aka “Bidirectional Instant Radiosity”

Idea:

▶ create a VPL at the 2\textsuperscript{nd} bounce from the camera
Improved Generation of VPLs

Bidirectional sampling for VPL distributions [Segovia et al. 2006]
aka “Bidirectional Instant Radiosity”

Idea:
▶ create a VPL at the 2nd bounce from the camera
Improved Generation of VPLs

Bidirectional sampling for VPL distributions  [Segovia et al. 2006]
aka “Bidirectional Instant Radiosity”

Idea:
➢ create a VPL at the 2\textsuperscript{nd} bounce from the camera
Improved Generation of VPLs
Improved Generation of VPLs

Metropolis sampling for VPL distributions [Segovia et al. 2007]
aka “Metropolis Instant Radiosity”
Improved Generation of VPLs

Metropolis sampling for VPL distributions [Segovia et al. 2007]
aKA “Metropolis Instant Radiosity”
Idea:

- generate VPLs by mutating paths
Improved Generation of VPLs

Metropolis sampling for VPL distributions [Segovia et al. 2007]
aka “Metropolis Instant Radiosity”

Idea:

- generate VPLs by mutating paths
Improved Generation of VPLs

Metropolis sampling for VPL distributions  [Segovia et al. 2007]
aka “Metropolis Instant Radiosity”

Idea:

- generate VPLs by mutating paths
Improved Generation of VPLs

Metropolis sampling for VPL distributions [Segovia et al. 2007]
aka “Metropolis Instant Radiosity”

▶ comparison with equal number of VPLs (1024)
Improved Generation of VPLs

Metropolis sampling for VPL distributions [Segovia et al. 2007]
aka “Metropolis Instant Radiosity”

▶ comparison with equal number of VPLs (1024)

Instant Radiosity (IR)  Bidirectional IR  Metropolis IR

Images courtesy of Segovia et al.
Improved Generation of VPLs

Metropolis sampling for VPL distributions  [Segovia et al. 2007]
aka “Metropolis Instant Radiosity”
Improved Generation of VPLs

Metropolis sampling for VPL distributions [Segovia et al. 2007]
aka “Metropolis Instant Radiosity”

Advantages:
▶ handles large and difficult scenes
▶ VPLs have equal contribution
Improved Generation of VPLs

Metropolis sampling for VPL distributions [Segovia et al. 2007]
aka “Metropolis Instant Radiosity”

Advantages:
- handles large and difficult scenes
- VPLs have equal contribution

Disadvantages:
- complicated implementation
- does not help with local inter-reflections
Improved Generation of VPLs
Improved Generation of VPLs

Local Virtual Lights  [Davidovič et al. 2010]

- improves glossy inter-reflections
- split light transport into a **global** and **local** component
Improved Generation of VPLs

Local Virtual Lights  [Davidović et al. 2010]

- improves glossy inter-reflections
- split light transport into a **global** and **local** component
Improved Generation of VPLs

Local Virtual Lights  [Davidovič et al. 2010]

- improves glossy inter-reflections
- split light transport into a **global** and **local** component
Improved Generation of VPLs

Local Virtual Lights  [Davidović et al. 2010]

- improves glossy inter-reflections
- split light transport into a **global** and **local** component
- details are mentioned later
Improved Generation of VPLs

Comparison
## Improved Generation of VPLs

### Comparison

<table>
<thead>
<tr>
<th></th>
<th>Moderately complex scenes</th>
<th>Complex scenes</th>
<th>Glossy scenes</th>
<th>Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Rejection of VPLs</strong></td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>[Georgiev and Slusallek 2010]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Metropolis IR</strong></td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>[Segovia et al. 2007]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Local Virtual Lights</strong></td>
<td>✓</td>
<td>✗</td>
<td>✓</td>
<td>✗</td>
</tr>
<tr>
<td>[Davidovič et al. 2010]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

✓ yes, easy
✗ no, difficult
Many-light Methods

Generation of VPLs

Lighting with VPLs
Many-light Methods

Generation of VPLs

Lighting with VPLs
Lighting with VPLs (assuming no media)
Lighting with VPLs (assuming no media)

Rendering Equation (area formulation)

\[ L(x_1 \rightarrow x_0) = L_e(x_1 \rightarrow x_0) + \int_A f(x_1) \cdot G(x_1, x_2) \cdot V(x_1, x_2) \cdot L(x_2 \rightarrow x_1) \, dA(x_2) \]
Lighting with VPLs (assuming no media)

Rendering Equation (area formulation)

\[ L(x_1 \rightarrow x_0) = L_e(x_1 \rightarrow x_0) + \int_A f(x_1) G(x_1, x_2) V(x_1, x_2) L(x_2 \rightarrow x_1) \, dA(x_2) \]
Lighting with VPLs (assuming no media)

Rendering Equation (area formulation)

\[ L(x_1 \rightarrow x_0) = L_e(x_1 \rightarrow x_0) + \int_A f(x_1) G(x_1, x_2) V(x_1, x_2) L(x_2 \rightarrow x_1) \, dA(x_2) \]

**BRDF**

\[ f(x_1) \]

**Geometry term**

\[ G(x_1, x_2) = \frac{\cos(\theta_1) \cos(\theta_2)}{\|x_1 - x_2\|^2} \]
Lighting with VPLs (assuming no media)

Rendering Equation (area formulation)

\[ L(x_1 \rightarrow x_0) = L_e(x_1 \rightarrow x_0) + \int_A f(x_1) \ G(x_1, x_2) \ V(x_1, x_2) \ L(x_2 \rightarrow x_1) \ dA(x_2) \]

**BRDF**

\[ f(x_1) \]

**Geometry term**

\[ G(x_1, x_2) = \frac{\cos(\theta_1) \cos(\theta_2)}{||x_1 - x_2||^2} \]

**Visibility term**

\[ V(x_1, x_2) \]
Lighting with VPLs (assuming no media)

Rendering Equation (area formulation)

\[
L(x_1 \rightarrow x_0) = L_e(x_1 \rightarrow x_0) + \int_A f(x_1) \ G(x_1, x_2) \ V(x_1, x_2) \ L(x_2 \rightarrow x_1) \ dA(x_2)
\]

BRDF
\[
f(x_1)
\]

Geometry term
\[
G(x_1, x_2) = \frac{\cos(\theta_1) \cos(\theta_2)}{||x_1 - x_2||^2}
\]

Visibility term
\[
V(x_1, x_2)
\]
Lighting with VPLs (assuming no media)

BRDF

\[ f(x_1) \]

Geometry term

\[ G(x_1, x_2) = \frac{\cos(\theta_1) \cos(\theta_2)}{||x_1 - x_2||^2} \]

Visibility term

\[ V(x_1, x_2) \]
Lighting with VPLs (assuming no media)

Paths between camera and light sources

\[
G(x_1, x_2) = \frac{\cos(\theta_1) \cos(\theta_2)}{||x_1 - x_2||^2}
\]
Lighting with VPLs (assuming no media)

Paths between camera and light sources

BRDF

Geometry term

Visibility term

\[ G(x_1, x_2) = \frac{\cos(\theta_1) \cos(\theta_2)}{||x_1 - x_2||^2} \]
Lighting with VPLs (assuming no media)

Paths between camera and light sources

**BRDF**

\[ f(x_1) \]

**Geometry term**

\[ G(x_1, x_2) \]

\[ G(x_1, x_2) = \frac{\cos(\theta_1) \cos(\theta_2)}{||x_1 - x_2||^2} \]

**Visibility term**

\[ V(x_1, x_2) \]
Lighting with VPLs (assuming no media)

Paths between camera and light sources

**BRDF**

\[ f(x_1) \]

\[ x_0 \rightarrow x_1 \rightarrow x_2 \]

**Geometry term**

\[ G(x_1, x_2) = \frac{\cos(\theta_1) \cos(\theta_2)}{||x_1 - x_2||^2} \]

\[ x_1 \rightarrow x_2 \]

**Visibility term**

\[ V(x_1, x_2) \]

\[ x_1 \rightarrow x_2 \]
Lighting with VPLs (assuming no media)

Paths between camera and light sources

\[ f(x_1) \quad G(x_1, x_2) \quad f(x_2) \quad \ldots \quad f(x_{k-1}) \quad G(x_{k-1}, x_k) \quad V(x_{k-1}, x_k) \]

**BRDF**

\[ f(x_1) \]

**Geometry term**

\[ G(x_1, x_2) = \frac{\cos(\theta_1) \cos(\theta_2)}{\|x_1 - x_2\|^2} \]

**Visibility term**

\[ V(x_1, x_2) \]
Lighting with VPLs (assuming no media)

Rendering Equation (area formulation)

\[ L(x_1 \rightarrow x_0) = L_e(x_1 \rightarrow x_0) + \int_A f(x_1) G(x_1, x_2) V(x_1, x_2) L(x_2 \rightarrow x_1) \, dA(x_2) \]
Lighting with VPLs (assuming no media)

Rendering Equation (area formulation)

\[
L(x_1 \rightarrow x_0) = L_e(x_1 \rightarrow x_0) + \int_A f(x_1) \ G(x_1, x_2) \ V(x_1, x_2) \ L(x_2 \rightarrow x_1) \ dA(x_2)
\]

\[
L(x_1 \rightarrow x_0) \approx L_e(x_1 \rightarrow x_0) + \sum_{i=1}^{N} f(x_1) \ G(x_1, x_2^i) \ V(x_1, x_2^i) \ f(x_2^i) \ \Phi_i
\]

Approximation using VPLs

- \( x_2^i \) - position of \( i^{th} \) VPL
- \( \Phi_i \) - “flux” of \( i^{th} \) VPL
Lighting with VPLs (assuming no media)

**Rendering Equation** (area formulation)

\[
L(x_1 \rightarrow x_0) = L_e(x_1 \rightarrow x_0) + \int_A f(x_1) G(x_1, x_2) V(x_1, x_2) L(x_2 \rightarrow x_1) \, dA(x_2)
\]

\[
L(x_1 \rightarrow x_0) \approx L_e(x_1 \rightarrow x_0) + \sum_{i=1}^{N} f(x_1) G(x_1, x_2^i) V(x_1, x_2^i) f(x_2^i) \Phi_i
\]

**Approximation using VPLs**

- \(x_2^i\) - position of \(i^{th}\) VPL
- \(\Phi_i\) - “flux” of \(i^{th}\) VPL
Lighting with VPLs (assuming no media)

Rendering Equation (area formulation)

\[
L(x_1 \rightarrow x_0) = L_e(x_1 \rightarrow x_0) + \int_A f(x_1) G(x_1, x_2) V(x_1, x_2) L(x_2 \rightarrow x_1) \, dA(x_2)
\]

\[
L(x_1 \rightarrow x_0) \approx L_e(x_1 \rightarrow x_0) + \sum_{i=1}^{N} f(x_1) G(x_1, x_{1,i}) V(x_1, x_{1,i}) f(x_{2,i}) \Phi_i
\]

Approximation using VPLs

- \( x_{2,i} \) - position of \( i^{th} \) VPL
- \( \Phi_i \) - “flux” of \( i^{th} \) VPL
Lighting with VPLs (assuming no media)

Rendering Equation (area formulation)

\[
L(x_1 \rightarrow x_0) = L_e(x_1 \rightarrow x_0) + \int_{A} f(x_1) G(x_1, x_2) V(x_1, x_2) L(x_2 \rightarrow x_1) \, dA(x_2)
\]

\[
L(x_1 \rightarrow x_0) \approx L_e(x_1 \rightarrow x_0) + \sum_{i=1}^{N} f(x_1) G(x_1, x_2^i) V(x_1, x_2^i) f(x_2^i) \Phi_i
\]

Approximation using VPLs

- \( x_2^i \) - position of \( i^{th} \) VPL
- \( \Phi_i \) - “flux” of \( i^{th} \) VPL
Lighting with VPLs (assuming no media)

Rendering Equation (area formulation)

\[
L(x_1 \rightarrow x_0) = L_e(x_1 \rightarrow x_0) + \int_A f(x_1) \ G(x_1, x_2) \ V(x_1, x_2) \ L(x_2 \rightarrow x_1) \ dA(x_2)
\]

\[
L(x_1 \rightarrow x_0) \approx L_e(x_1 \rightarrow x_0) + \sum_{i=1}^{N} f(x_1) \ G(x_1, x_i^i) \ V(x_1, x_i^i) \ f(x_i^i) \ \Phi_i
\]

Approximation using VPLs

- \(x_i^i\) - position of \(i^{th}\) VPL
- \(\Phi_i\) - “flux” of \(i^{th}\) VPL
Lighting with VPLs (assuming no media)

Rendering Equation (area formulation)

\[ L(x_1\rightarrow x_0) = L_e(x_1\rightarrow x_0) + \int_A f(x_1) \ G(x_1, x_2) \ V(x_1, x_2) \ L(x_2\rightarrow x_1) \ dA(x_2) \]

\[ L(x_1\rightarrow x_0) \approx L_e(x_1\rightarrow x_0) + \sum_{i=1}^{N} f(x_1) \ G(x_1, x_2^i) \ V(x_1, x_2^i) \ f(x_2^i) \ \Phi_i \]

Approximation using VPLs

- \( x_2^i \) - position of \( i^{th} \) VPL
- \( \Phi_i \) - “flux” of \( i^{th} \) VPL

recursion is hidden in the generation of VPLs
Lighting with VPLs (assuming no media)

Reference
Lighting with VPLs (assuming no media)

Reference

Approximation with VPLs
Lighting with VPLs (assuming no media)

Splotches!!!
Lighting with VPLs (assuming no media)

Splotches!!!

Reasons:
Lighting with VPLs (assuming no media)

Splotches!!!

Reasons:

- geometry term

\[
G(x_1, x_2) = \frac{\cos(\theta_1) \cos(\theta_2)}{||x_1 - x_2||^2}
\]
Lighting with VPLs (assuming no media)

Splotches!!!

Reasons:
- geometry term

\[ G(x_1, x_2) = \frac{\cos(\theta_1) \cos(\theta_2)}{\|x_1 - x_2\|^2} \]
Lighting with VPLs (assuming no media)

Splotches!!!

Reasons:

- geometry term

\[ G(x_1, x_2) = \frac{\cos(\theta_1) \cos(\theta_2)}{\|x_1 - x_2\|} \]

- correlation in the estimator

- all points are lit by the same set of VPLs
Lighting with VPLs

Splotches!!!
Lighting with VPLs

Splotches!!!

Solutions:
Lighting with VPLs

Splotches!!!

Solutions:

1. Bound the geometry term
   - removes energy, darkens the image
   - to get unbiased results, we need to compensate for the bounding

   [Kollig and Keller 2004], [Raab et al. 2008], [Davidovič et al. 2010], [Novák et al. 2011], [Engelhardt et al. 2012]
Lighting with VPLs

Splotches!!!

Solutions:

1. **Bound the geometry term**
   - removes energy, darkens the image
   - to get unbiased results, we need to compensate for the bounding
     
     [Kollig and Keller 2004], [Raab et al. 2008], [Davidovič et al. 2010], [Novák et al. 2011], [Engelhardt et al. 2012]

2. **Distribute the flux of a VPL over area (volume)**
   - redistributes energy, blurs the illumination
   - to get consistent results, progressively reduce the blurring
     
     [Hašan et al. 2009], [Novák et al. 2012a], [Novák et al. 2012b]
Lighting with VPLs

Splotches!!!

Solutions:

1. **Bound the geometry term**
   - removes energy, darkens the image
   - to get unbiased results, we need to compensate for the bounding
   

2. **Distribute the flux of a VPL over area (volume)**
   - redistributes energy, blurred the illumination
   - to get consistent results, progressively reduce the blurring

   [Hašan et al. 2009], [Novák et al. 2012a], [Novák et al. 2012b]
Bounding & Compensation
Bounding & Compensation

Bounding the geometry term

- prevent $G$ from being very high
- $b$ - user-defined maximum value (bound)

$$G_b(x_1, x_2) = \min(G(x_1, x_2), b)$$
Bounding & Compensation

Bounding the geometry term

- prevent $G$ from being very high
- $b$ - user-defined maximum value (bound)

$$G_b(x_1, x_2) = \min(G(x_1, x_2), b)$$

Advantages:

- extremely simple and fast
Bounding & Compensation

Bounding the geometry term

- prevent $G$ from being very high
- $b$ - user-defined maximum value (bound)

$$G_b(x_1, x_2) = \min(G(x_1, x_2), b)$$

Advantages:
- extremely simple and fast

Disadvantages:
- removes energy, darkens the image
Bounding & Compensation

Reference

VPLs

using $G(x_1, x_2)$
Bounding & Compensation

Reference

VPLs

VPLs with bounded $G$

using $G(x_1, x_2)$

using $G_b(x_1, x_2)$
Bounding & Compensation

Reference

Difference

VPLs with bounded $G$

\[ G(x_1, x_2) - G_b(x_1, x_2) \]

using $G_b(x_1, x_2)$
We need to compensate for the energy loss!
Bounding & Compensation

Expressing the energy loss
Bounding & Compensation

Expressing the energy loss

light transport operator $T$:

$$(TL)(x_1 \rightarrow x_0) = \int_A f(x_1) \, G(x_1, x_2) \, V(x_1, x_2) \, L(x_2 \rightarrow x_1) \, dA(x_2)$$
Bounding & Compensation

Expressing the energy loss

∀ light transport operator $T$:

$$(TL)(x_1 \rightarrow x_0) = \int_A f(x_1) \ G(x_1, x_2) \ V(x_1, x_2) \ L(x_2 \rightarrow x_1) \ dA(x_2)$$

∀ bounded light transport operator $T_b$:

$$(T_bL)(x_1 \rightarrow x_0) = \int_A f(x_1) \ \min(G(x_1, x_2), b) \ V(x_1, x_2) \ L(x_2 \rightarrow x_1) \ dA(x_2)$$
Bounding & Compensation

Expressing the energy loss

- light transport operator $T$:
  \[
  (TL)(x_1 \rightarrow x_0) = \int_A f(x_1) \ G(x_1, x_2) \ V(x_1, x_2) \ L(x_2 \rightarrow x_1) \ dA(x_2)
  \]

- bounded light transport operator $T_b$:
  \[
  (T_bL)(x_1 \rightarrow x_0) = \int_A f(x_1) \ \min(G(x_1, x_2), b) \ V(x_1, x_2) \ L(x_2 \rightarrow x_1) \ dA(x_2)
  \]

- residual light transport operator (compensation term) $T_r$:
  \[
  (T_rL)(x_1 \rightarrow x_0) = \int_A f(x_1) \ \max(G(x_1, x_2) - b, 0) \ V(x_1, x_2) \ L(x_2 \rightarrow x_1) \ dA(x_2)
  \]
Bounding & Compensation

Expressing the energy loss

- light transport operator $T$:
  \[ TL = T_b L + T_r L \]

- bounded light transport operator $T_b$:
  \[ (T_b L)(x_1 \to x_0) = \int_A f(x_1) \min(G(x_1, x_2), b) \, V(x_1, x_2) \, L(x_2 \to x_1) \, dA(x_2) \]

- residual light transport operator (compensation term) $T_r$:
  \[ (T_r L)(x_1 \to x_0) = \int_A f(x_1) \max(G(x_1, x_2) - b, 0) \, V(x_1, x_2) \, L(x_2 \to x_1) \, dA(x_2) \]
Bounding & Compensation

Expressing the energy loss

light transport operator $T$:

$$TL = TbL + TrL$$
Bounding & Compensation

Expressing the energy loss

- light transport operator $T$:

$$ TL = T_b L + T_r L $$

Estimate using VPLs
Bounding & Compensation

Expressing the energy loss

- light transport operator $T$:

$$ TL = T_b L + T_r L $$

Estimate using VPLs

Estimate “differently”
Bounding & Compensation

Expressing the energy loss

light transport operator $T$:

$$TL = T_{b}L + T_{r}L$$

- Estimate using VPLs
- Estimate “differently”

[Kollig and Keller 2004]
[Raab et al. 2008]
[Davidovič et al. 2010]
[Novák et al. 2011]
[Engelhardt et al. 2012]
Bounding & Compensation

Bias compensation  [Kollig and Keller 2004], [Raab et al. 2008]

- trace paths to compute the compensation term (residual transport)
Bounding & Compensation

Bias compensation  [Kollig and Keller 2004], [Raab et al. 2008]

- trace paths to compute the compensation term (residual transport)
Bounding & Compensation

Bias compensation  [Kollig and Keller 2004], [Raab et al. 2008]

► trace paths to compute the compensation term (residual transport)
Bounding & Compensation

Bias compensation  [Kollig and Keller 2004], [Raab et al. 2008]

• trace paths to compute the compensation term (residual transport)
Bounding & Compensation

Bias compensation  [Kollig and Keller 2004], [Raab et al. 2008]

- trace paths to compute the compensation term (residual transport)
Bounding & Compensation

Bias compensation  [Kollig and Keller 2004], [Raab et al. 2008]

- trace paths to compute the compensation term (residual transport)
Bounding & Compensation

Bias compensation [Kollig and Keller 2004], [Raab et al. 2008]

- trace paths to compute the compensation term (residual transport)
Bounding & Compensation

Bias compensation  [Kollig and Keller 2004], [Raab et al. 2008]

- trace paths to compute the compensation term (residual transport)
Bounding & Compensation

Bias compensation  [Kollig and Keller 2004], [Raab et al. 2008]

▶ trace paths to compute the compensation term (residual transport)
Bounding & Compensation

Bias compensation [Kollig and Keller 2004], [Raab et al. 2008]

_trace paths to compute the compensation term (residual transport)_)
Bounding & Compensation

Bias compensation  [Kollig and Keller 2004], [Raab et al. 2008]

▶ trace paths to compute the compensation term (residual transport)
Bounding & Compensation

Bias compensation  [Kollig and Keller 2004], [Raab et al. 2008]

- trace paths to compute the compensation term (residual transport)
Bounding & Compensation

Bias compensation  [Kollig and Keller 2004], [Raab et al. 2008]

▶ trace paths to compute the compensation term (residual transport)
Bounding & Compensation

Bias compensation [Kollig and Keller 2004], [Raab et al. 2008]

▶ trace paths to compute the compensation term (residual transport)

Advantages:

▶ recovers all missing energy
▶ makes the algorithm unbiased
Bounding & Compensation

Bias compensation [Kollig and Keller 2004], [Raab et al. 2008]

- trace paths to compute the compensation term (residual transport)

Advantages:

- recovers all missing energy
- makes the algorithm unbiased

Disadvantages:

- recursive; degenerates to path tracing!
- very expensive
  - “recovering 10% of energy may take 90% of the rendering time”
Bounding & Compensation

Bias compensation [Kollig and Keller 2004], [Raab et al. 2008]

Bounding only

Bounding & compensation

Images courtesy of Kollig and Keller
Bounding & Compensation
Bounding & Compensation

Local Virtual Lights  [Davidovič et al. 2010]

» global (bounded) transport: VPLs
» local (residual) transport: on-demand local virtual lights
Bounding & Compensation

Local Virtual Lights  [Davidović et al. 2010]

- global (bounded) transport: VPLs
- local (residual) transport: on-demand local virtual lights
Bounding & Compensation

**Local Virtual Lights**  [Davidovič et al. 2010]

- global (bounded) transport: VPLs
- local (residual) transport: on-demand local virtual lights
Bounding & Compensation

Local Virtual Lights [Davidović et al. 2010]

- global (bounded) transport: VPLs
- local (residual) transport: on-demand local virtual lights
Bounding & Compensation

Local Virtual Lights  [Davidovič et al. 2010]

- global (bounded) transport: VPLs
- local (residual) transport: on-demand local virtual lights
Bounding & Compensation

Local Virtual Lights [Davidović et al. 2010]

- global (bounded) transport: VPLs
- local (residual) transport: on-demand local virtual lights
Bounding & Compensation

Local Virtual Lights  [Davidović et al. 2010]

- global (bounded) transport: VPLs
- local (residual) transport: on-demand local virtual lights
Bounding & Compensation

Local Virtual Lights  [Davidovič et al. 2010]
- global (bounded) transport: VPLs
- local (residual) transport: on-demand local virtual lights

Advantages:
- faster than Kollig and Keller’s approach
- amortizes creation of each local light over several pixels
- handles glossy inter-reflection
Bounding & Compensation

Local Virtual Lights  [Davidovič et al. 2010]
- global (bounded) transport: VPLs
- local (residual) transport: on-demand local virtual lights

Advantages:
- faster than Kollig and Keller’s approach
- amortizes creation of each local light over several pixels
- handles glossy inter-reflection

Disadvantages:
- proposed (involved) implementation approximates visibility
Bounding & Compensation

Local Virtual Lights  [Davidovič et al. 2010]

Global (bounded)  +  Local (residual)  =  Composited

Images courtesy of Davidovič et a.
Bounding & Compensation
Bounding & Compensation

Screen-space Bias Compensation  [Novák et al. 2011]

- residual transport is localized and can be applied in post-process
Bounding & Compensation

Screen-space Bias Compensation  [Novák et al. 2011]

- residual transport is localized and can be applied in post-process

- Rendering Equation:

\[ L = L_e + TL \]
Bounding & Compensation

Screen-space Bias Compensation  [Novák et al. 2011]

- residual transport is localized and can be applied in post-process

- Rendering Equation:

\[ L = L_e + TL \]
\[ \approx L_e + TL_e + TL \]
Bounding & Compensation

Screen-space Bias Compensation [Novák et al. 2011]

- residual transport is localized and can be applied in post-process

Rendering Equation:

\[ L = L_e + TL \]

\[ \approx L_e + TL_e + TL \]

- emission
- direct illumination
- indirect illumination computed using VPLs
Bounding & Compensation

Screen-space Bias Compensation [Novák et al. 2011]

- residual transport is localized and can be applied in post-process

- Rendering Equation:

\[
L = L_e + TL \\
\approx L_e + TL_e + T\hat{L} \\
\approx L_e + TL_e + Tb\hat{L} + Tr\hat{L}
\]
Bounding & Compensation

Screen-space Bias Compensation  [Novák et al. 2011]

- residual transport is localized and can be applied in post-process

- Rendering Equation:

\[ L = L_e + TL \]
\[ \approx L_e + TL_e + TL \]
\[ \approx L_e + TL_e + T_b \hat{L} + T_r \hat{L} \]

bounded residual
Bounding & Compensation

Screen-space Bias Compensation  [Novák et al. 2011]

- residual transport is localized and can be applied in post-process

- Rendering Equation:

\[
L = L_e + TL \\
\approx L_e + TL_e + TL \hat{L} \\
\approx L_e + TL_e + Tb \hat{L} + Tr \hat{L} \\
\approx L_e + TL_e + Tb \hat{L} + Tr (L - L_e)
\]
Bounding & Compensation

Screen-space Bias Compensation  [Novák et al. 2011]

- residual transport is localized and can be applied in post-process

- Rendering Equation:

\[
L = L_e + TL \\
\approx L_e + TL_e + TL \hat{L} \\
\approx L_e + TL_e + Tb \hat{L} + Tr \hat{L} \\
\approx L_e + TL_e + Tb \hat{L} + Tr(L - L_e)
\]

recursively expand
Bounding & Compensation

Screen-space Bias Compensation  [Novák et al. 2011]

- residual transport is localized and can be applied in post-process

- Rendering Equation:

\[
L = L_e + TL
\]

\[
\approx L_e + TL_e + T\hat{L}
\]

\[
\approx L_e + TL_e + T_{b}\hat{L} + T_r \hat{L}
\]

\[
\approx L_e + TL_e + T_{b}\hat{L} + T_r (L - L_e)
\]

\[
\approx L_e + \sum_{i=0}^{\infty} T_r^i (TL_e + T_{b}\hat{L})
\]
Screen-space Bias Compensation [Novák et al. 2011]

- residual transport is localized and can be applied in post-process

Rendering Equation:

\[
L = L_e + TL
\]

\[
\approx L_e + TL_e + T\hat{L}
\]

\[
\approx L_e + TL_e + T_b\hat{L} + T_r\hat{L}
\]

\[
\approx L_e + TL_e + T_b\hat{L} + T_r(L - L_e)
\]

\[
\approx L_e + \sum_{i=0}^{\infty} T_r^i (TL_e + T_b\hat{L})
\]

compute once and store
Bounding & Compensation

Screen-space Bias Compensation [Novák et al. 2011]

- residual transport is localized and can be applied in post-process

- Rendering Equation:

\[
L = L_e + TL \\
\approx L_e + TLE + TL\hat{L} \\
\approx L_e + TLE + Tb\hat{L} + Tr\hat{L} \\
\approx L_e + TLE + Tb\hat{L} + Tr(L - L_e) \\
\approx L_e + \sum_{i=0}^{\infty} Tr^i (TLE + Tb\hat{L})
\]

Iteratively apply \( Tr \) and compute once and store
Bounding & Compensation

Screen-space Bias Compensation  [Novák et al. 2011]

- residual transport is localized and can be applied in post-process

- Rendering Equation:

\[ \approx L_e + \sum_{i=0}^{\infty} T^i_r (TL_e + T_b \hat{L}) \]
Bounding & Compensation

Screen-space Bias Compensation  [Novák et al. 2011]

- residual transport is localized and can be applied in post-process

- Rendering Equation:

  \[
  \approx L_e + \sum_{i=0}^{\infty} T^i_r(TL_e + T_b \hat{L})
  \]

  direct + bounded indirect illumination

  \[
  T^0_r(TL_e + T_b \hat{L})
  \]
Bounding & Compensation

Screen-space Bias Compensation  [Novák et al. 2011]

- residual transport is localized and can be applied in post-process

- Rendering Equation:

\[ \approx L_e + \sum_{i=0}^{\infty} T_r^i (TL_e + T_b \hat{L}) \]

bounding & compensation

direct + bounded indirect illumination

residual transport in screen-space

\[ T_r^0 (TL_e + T_b \hat{L}) \]

\[ T_r^1 (TL_e + T_b \hat{L}) \]
Bounding & Compensation

Screen-space Bias Compensation  [Novák et al. 2011]

- residual transport is localized and can be applied in post-process

Rendering Equation:

$$\approx L_e + \sum_{i=0}^{\infty} T_r^i (TL_e + T_b \hat{L})$$

- direct + bounded indirect illumination
- residual transport in screen-space
- residual transport in screen-space

$$T_r^0 (TL_e + T_b \hat{L}) + T_r^1 (TL_e + T_b \hat{L}) + T_r^2 (TL_e + T_b \hat{L}) + \ldots$$
Bounding & Compensation

Screen-space Bias Compensation  [Novák et al. 2011]

- residual transport is localized and can be applied in post-process
Bounding & Compensation

Screen-space Bias Compensation  [Novák et al. 2011]

- residual transport is localized and can be applied in post-process

Implementation:

- 2 steps are usually sufficient
- GPU-friendly, hierarchical screen-space integration
Bounding & Compensation

Screen-space Bias Compensation  [Novák et al. 2011]

- residual transport is localized and can be applied in post-process

Implementation:

- 2 steps are usually sufficient
- GPU-friendly, hierarchical screen-space integration

Advantages:

- fast! (1024x768 @ 20-30 milliseconds)
- can be implemented as an image filter
Bounding & Compensation

**Screen-space Bias Compensation**  [Novák et al. 2011]

- residual transport is localized and can be applied in post-process

**Implementation:**

- 2 steps are usually sufficient
- GPU-friendly, hierarchical screen-space integration

**Advantages:**

- fast! (1024x768 @ 20-30 milliseconds)
- can be implemented as an image filter

**Disadvantages:**

- approximative, uses information from surfaces visible to camera only
- must be conservative, otherwise artifacts can occur
Bounding & Compensation

Screen-space Bias Compensation  [Novák et al. 2011]

Direct + bounded indirect

1- and 2-bounce residual

Composited
Bounding & Compensation
Bounding & Compensation

Approximate Bias Compensation  [Engelhardt et al. 2012]

▷ efficient compensation for participating media
Bounding & Compensation

Approximate Bias Compensation  [Engelhardt et al. 2012]

- efficient compensation for participating media

Reference
Bounding & Compensation

Approximate Bias Compensation  [Engelhardt et al. 2012]

> efficient compensation for participating media

Reference  VPLs  VPLs with bounded $G$
Bounding & Compensation

Approximate Bias Compensation  [Engelhardt et al. 2012]

▷ efficient compensation for participating media

Reference
Bounding & Compensation

Approximate Bias Compensation [Engelhardt et al. 2012]

- efficient compensation for participating media

Reference
Bounding & Compensation

Approximate Bias Compensation  [Engelhardt et al. 2012]

- efficient compensation for participating media
Approximate Bias Compensation [Engelhardt et al. 2012]

- efficient compensation for participating media

Reference
Bounding & Compensation

Approximate Bias Compensation  [Engelhardt et al. 2012]

- efficient compensation for participating media

Reference
Bounding & Compensation

Approximate Bias Compensation  [Engelhardt et al. 2012]

► efficient compensation for participating media

Reference

![Graph showing reference and bounded VPLs with 2 bounces of BC]
Approximate Bias Compensation  [Engelhardt et al. 2012]

- efficient compensation for participating media
Bounding & Compensation

Approximate Bias Compensation  [Engelhardt et al. 2012]

- efficient compensation for participating media

Optimizations used for BC:

- assume locally homogeneous media
- omit testing local visibility
Bounding & Compensation

Approximate Bias Compensation [Engelhardt et al. 2012]

- efficient compensation for participating media

Optimizations used for BC:

- assume locally homogeneous media
- omit testing local visibility

Advantages:

- fast, GPU friendly
Bounding & Compensation

Approximate Bias Compensation  [Engelhardt et al. 2012]
- efficient compensation for participating media

Optimizations used for BC:
- assume locally homogeneous media
- omit testing local visibility

Advantages:
- fast, GPU friendly

Disadvantages:
- approximate, complicated
Bounding & Compensation

Approximate Bias Compensation [Engelhardt et al. 2012]

bounded: 39 min.
approx. bias comp.: 13 min.
Lighting with VPLs

Splotches!!!

Solutions:

1. Bound the geometry term
   - removes energy, darkens the image
   - to get unbiased results, we need to compensate for the bounding
     [Kollig and Keller 2004], [Raab et al. 2008], [Davidovič et al. 2010], [Novák et al. 2011], [Engelhardt et al. 2012]

2. Distribute the flux of a VPL over area (volume)
   - redistributes energy, blurs the illumination
   - to get consistent results, progressively reduce the blurring
     [Hašan et al. 2009], [Novák et al. 2012a], [Novák et al. 2012b]
Lighting with VPLs

Splotches!!!

Solutions:

1. **Bound the geometry term**
   - removes energy, darkens the image
   - to get unbiased results, we need to compensate for the bounding
     
     [Kollig and Keller 2004], [Raab et al. 2008], [Davidovič et al. 2010], [Novák et al. 2011], [Engelhardt et al. 2012]

2. **Distribute the flux of a VPL over area (volume)**
   - redistributes energy, blurs the illumination
   - to get consistent results, progressively reduce the blurring
     
     [Hašan et al. 2009], [Novák et al. 2012a], [Novák et al. 2012b]
Spreading the Energy

Virtual Spherical Lights  [Hašan et al. 2009]

- distribute the energy of the infinitesimal VPL over nearby surfaces inside a sphere
Spreading the Energy

Virtual Spherical Lights  [Hašan et al. 2009]

- distribute the energy of the infinitesimal VPL over nearby surfaces inside a sphere
Spreading the Energy

Virtual Spherical Lights [Hašan et al. 2009]

- distribute the energy of the infinitesimal VPL over nearby surfaces inside a sphere

point-to-point: $\Phi V(x_1, x_2) f(x_1) f(x_2) \frac{\cos \theta_1 \cos \theta_2}{||x_1 - x_2||^2}$
Spreading the Energy

Virtual Spherical Lights  [Hašan et al. 2009]

- distribute the energy of the infinitesimal VPL over nearby surfaces inside a sphere

$$\Phi \ V(\mathbf{x}_1, \mathbf{x}_2) \ f(\mathbf{x}_1) \ f(\mathbf{x}_2) \ \frac{\cos \theta_1 \cos \theta_2}{\|\mathbf{x}_1 - \mathbf{x}_2\|^2}$$
Spreading the Energy

Virtual Spherical Lights [Hašan et al. 2009]

- distribute the energy of the infinitesimal VPL over nearby surfaces inside a sphere

point-to-point: $$\Phi V(x_1, x_2) f(x_1) f(x_2) \frac{\cos \theta_1 \cos \theta_2}{||x_1 - x_2||^2}$$
Spreading the Energy

Virtual Spherical Lights [Hašan et al. 2009]

> distribute the energy of the infinitesimal VPL over nearby surfaces inside a sphere

point-to-point: \[ \Phi V(x_1, x_2) f(x_1) f(x_2) \frac{\cos \theta_1 \cos \theta_2}{\|x_1 - x_2\|^2} \]

approx. sphere-to-point: \[ \frac{\Phi}{\pi r^2} V(x_1, x_2) \int_\Omega f(x_1) f(x_2) \cos \theta_1 \cos \theta_2 d\omega \]
Spreading the Energy

Virtual Spherical Lights [Hašan et al. 2009]

► distribute the energy of the infinitesimal VPL over nearby surfaces inside a sphere
Spreading the Energy

Virtual Spherical Lights  [Hašan et al. 2009]

► distribute the energy of the infinitesimal VPL over nearby surfaces inside a sphere

cone sampling  BRDF1 sampling  BRDF2 sampling  Multiple importance sampling
Bounding & Compensation

Virtual Spherical Lights  [Hašan et al. 2009]

- distribute the energy of the infinitesimal VPL over nearby surfaces inside a sphere
Bounding & Compensation

Virtual Spherical Lights  [Hašan et al. 2009]

➤ distribute the energy of the infinitesimal VPL over nearby surfaces inside a sphere

Advantages:

➤ energy is blurred, not clamped
Bounding & Compensation

Virtual Spherical Lights  [Hašan et al. 2009]
- distribute the energy of the infinitesimal VPL over nearby surfaces inside a sphere

Advantages:
- energy is blurred, not clamped

Disadvantages:
- introduces bias
- requires an extra integration over the solid angle
Spreading the Energy

Virtual Spherical Lights [Hašan et al. 2009]

Reference 2.2 hours
Bounded 32 sec
VSLs 1 min 44 sec
Spreading the Energy

Virtual Spherical Lights [Hašan et al. 2009]

Reference

VSLs

VSLs converged

path tracing

5,000 VSLs

1,000,000 VSLs
Spreading the Energy

Virtual Ray Lights  [Novák et al. 2012a]

- many-light technique for participating media
- use segments of the random walk as light sources
Spreading the Energy

Virtual Ray Lights  [Novák et al. 2012a]

- many-light technique for participating media
- use segments of the random walk as light sources

Virtual Point Lights
Spreading the Energy

Virtual Ray Lights  [Novák et al. 2012a]

- many-light technique for participating media
- use segments of the random walk as light sources

Virtual Point Lights

Virtual Ray Lights

- higher sampling of path space
- provably reduce singularities
Spreading the Energy

Virtual Ray Lights  [Novák et al. 2012a]

- many-light technique for participating media
- use segments of the random walk as light sources
Spreading the Energy

Virtual Ray Lights  [Novák et al. 2012a]

- many-light technique for participating media
- use segments of the random walk as light sources
Spreading the Energy

Virtual Ray Lights  [Novák et al. 2012a]

- many-light technique for participating media
- use segments of the random walk as light sources
Spreading the Energy

Virtual Ray Lights  [Novák et al. 2012a]

- many-light technique for participating media
- use segments of the random walk as light sources
Spreading the Energy

Virtual Ray Lights [Novák et al. 2012a]

- many-light technique for participating media
- use segments of the random walk as light sources

![Images of VRLs and VPLs](images.png)
Spreading the Energy

**Virtual Ray Lights**  [Novák et al. 2012a]

- many-light technique for participating media
- use segments of the random walk as light sources
Spreading the Energy

Virtual Ray Lights  [Novák et al. 2012a]

- many-light technique for participating media
- use segments of the random walk as light sources

Advantages:
- energy is spread along lines, singularity is reduced (not removed)
- unbiased, temporally stable
Spreading the Energy

Virtual Ray Lights [Novák et al. 2012a]

- many-light technique for participating media
- use segments of the random walk as light sources

Advantages:
- energy is spread along lines, singularity is reduced (not removed)
- unbiased, temporally stable

Disadvantages:
- requires 2D integration (along both rays)
Spreading the Energy

Virtual Ray Lights [Novák et al. 2012a]

Multiple scattering

VRLs

VPLs

1200 s

1200 s
Spreading the Energy

Progressive Virtual Beam Lights [Novák et al. 2012b]

- give thickness to VRLs
- use the concept of VSLs to “inflate” points on the ray
Spreading the Energy

Progressive Virtual Beam Lights [Novák et al. 2012b]

- give thickness to VRLs
- use the concept of VSLs to “inflate” points on the ray

Virtual Ray Lights
Spreading the Energy

Progressive Virtual Beam Lights [Novák et al. 2012b]

- give thickness to VRLs
- use the concept of VSLs to “inflate” points on the ray

Virtual Ray Lights

Virtual Beam Lights

- remove singularities completely
- similar to the concept of VSLs
Spreading the Energy

Progressive Virtual Beam Lights [Novák et al. 2012b]

- give thickness to VRLs
- use the concept of VSLs to “inflate” points on the ray
Spreading the Energy

Progressive Virtual Beam Lights  [Novák et al. 2012b]

Buddha Scene
homogeneous
anisotropic (HG g= 0.7)
Spreading the Energy

Progressive Virtual Beam Lights  [Novák et al. 2012b]
Spreading the Energy

Progressive Virtual Beam Lights  [Novák et al. 2012b]
Spreading the Energy

Progressive Virtual Beam Lights [Novák et al. 2012b]

Advantages:

- energy is preserved, distributed over the volume of a beam
- no singularities
- progressively reduces the beam width -> converges to ground truth
Spreading the Energy

Progressive Virtual Beam Lights  [Novák et al. 2012b]

Advantages:
- energy is preserved, distributed over the volume of a beam
- no singularities
- progressively reduces the beam width -> converges to ground truth

Disadvantages:
- requires integration along both rays and over the solid angle
- may over-blur sharp illumination features at the beginning
Lighting with VPLs
# Lighting with VPLs

Comparison of techniques handling surfaces only

<table>
<thead>
<tr>
<th></th>
<th>Speed</th>
<th>Quality</th>
<th>Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Bounding only</strong></td>
<td>✓</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td><strong>Bias Compensation</strong></td>
<td>✗</td>
<td>✓</td>
<td>✓/✗</td>
</tr>
<tr>
<td>[Kollig and Keller 2004]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Local Virtual Lights</strong></td>
<td>✓/✗</td>
<td>✓/✗</td>
<td>✓/✗</td>
</tr>
<tr>
<td>[Davidović et al. 2010]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Screen-space Bias Comp.</strong></td>
<td>✓</td>
<td>✓/✗</td>
<td>✓/✗</td>
</tr>
<tr>
<td>[Novák et al. 2012]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Virtual Spherical Lights</strong></td>
<td>✓/✗</td>
<td>✓/✗</td>
<td>✓/✗</td>
</tr>
<tr>
<td>[Hašan et al. 2009]</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

✓ good, easy
✗ bad, difficult
Lighting with VPLs

Comparison of techniques handling participating media
Lighting with VPLs

Comparison of techniques handling participating media

<table>
<thead>
<tr>
<th>Technique</th>
<th>Speed</th>
<th>Quality</th>
<th>Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bounding only</td>
<td>✓</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>Bias Compensation</td>
<td>✗</td>
<td>✓</td>
<td>✓/✗</td>
</tr>
<tr>
<td>[Raab et al. 2008]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Approximate Bias Comp.</td>
<td>✓</td>
<td>✓/✗</td>
<td>✗</td>
</tr>
<tr>
<td>[Engelhardt et al. 2012]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Virtual Ray Lights</td>
<td>✓/✗</td>
<td>✓/✗</td>
<td>✓/✗</td>
</tr>
<tr>
<td>[Novák et al. 2012]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Virtual Beam Lights</td>
<td>✓/✗</td>
<td>✓</td>
<td>✓/✗</td>
</tr>
<tr>
<td>[Novák et al. 2012]</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

 ✓: good, easy
 ✗: bad, difficult
References


- [Georgiev and Slusallek 2010] - *Simple and Robust Iterative Importance Sampling of Virtual Point Lights*, Eurographics '10 (short papers), 2010


