Transient Rendering

Adam Smith
James Skorupski

University of California, Santa Cruz
December 11, 2007
CMPS 290B
{amsmith, jskorups}@cs.ucsc.edu
Motivation

There is growing interest in time-of-flight based computer vision applications and we want some general, physical explanation of measurements we make.

Our contribution:
a formal model that let’s us do just that
Background

• We want
  ▫ Rigorous analysis
  ▫ Specific to light
  ▫ Transient effects

• What’s out there
  ▫ LIDAR
  ▫ SONAR
  ▫ Rendering Equation
LIDAR

- **NO**: Rigorous analysis
- **YES**: Specific to light
- **YES**: Transient effects

(UC Santa Cruz)

(UC Davis)
SONAR

• YES: Rigorous analysis
• NO: Specific to light
• YES: Transient effects

Height field of two sunken ships

(USGS)

Overview (Transponder in yellow)

(NOAA)
Rendering Equation

- YES: Rigorous analysis
- YES: Specific to light
- NO: Transient effects

\[ L = L_0 + G \circ L \]

where

- L is total light
- \( L_0 \) is emitted light
- G is global transport (single bounce)

The rendering equation is

\[ I(x, x') = g(x, x') \left[ e(x, x') + \int g(x, x'', x') \rho(x, x', x'') \right] \]

where:
- \( I(x, x') \) is the related to the intensity of light passing from point \( x' \) to point \( x \)
- \( g(x, x') \) is a “geometry” term
- \( e(x, x') \) is related to the intensity of emitted light from \( x' \) to \( x \)
- \( \rho(x, x'x'') \) is related to the intensity of light scattered from \( x'' \) to \( x \) by a patch of surface at \( x' \)

(Kajiya, 1986)
The Important Distinction

Steady state vs. transient light transport
Visualization

- Steady state: Where the light comes out
Visualization

- Transient: *When* the light comes out
Visualization
Visualization
Visualization
Visualization
Visualization
Visualization

- Note: Top pulse wins the race!
## Energy vs. Power

<table>
<thead>
<tr>
<th>Steady State</th>
<th>Transient</th>
</tr>
</thead>
<tbody>
<tr>
<td>Energy (Joules)</td>
<td>Power (Watts)</td>
</tr>
<tr>
<td>Number of photons received</td>
<td>Rate of photons received</td>
</tr>
<tr>
<td>Radiance</td>
<td>Radiant flux</td>
</tr>
</tbody>
</table>
Infinite vs. Finite

<table>
<thead>
<tr>
<th>Steady State</th>
<th>Transient</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{Delay}(X, Y) = 0$</td>
<td>$\text{Delay}(X, Y) = \frac{</td>
</tr>
</tbody>
</table>

- $X, Y$ are points
- $c$ is the speed of light
## Functions

<table>
<thead>
<tr>
<th>Steady State</th>
<th>Transient</th>
</tr>
</thead>
<tbody>
<tr>
<td>( L = f(X, \omega) )</td>
<td>( F = g(X, \omega, t) )</td>
</tr>
</tbody>
</table>

- \( X \) is a point
- \( \omega \) is a direction
- \( t \) is a time
Transient Rendering Equation

(our contribution)

\[ F(t) = F_0(t) + G \circ F(t) \]
Transient Rendering Equation

- Global light transport $G$ is the composition of two physical processes
  - propagation, $P$
    - delays light over distances
  - scattering, $S$
    - same as traditional rendering

$$G \circ F(t) = S \circ P \circ F(t)$$
Example

a) 1-d world with two surfaces A and B, eye E and light L
b) result of transient rendering
c) light seen at E over time

- Input: positions, scattering kernels, initial light emission
- Output: received light power at every point, every direction, and every time
Example

a) 1-d world with two surfaces A and B, eye E and light L
b) result of transient rendering
c) light seen at E over time

- Input: positions, scattering kernels, initial light emission
- Output: received light power at every point, every direction, and every time
Example

a) 1-d world with two surfaces A and B, eye E and light L
b) result of transient rendering
c) light seen at E over time

- Input: positions, scattering kernels, initial light emission
- Output: received light power at every point, every direction, and every time
a) 1-d world with two surfaces A and B, eye E and light L
b) result of transient rendering
c) light seen at E over time

- Input: positions, scattering kernels, initial light emission
- Output: received light power at every point, every direction, and every time
Example

a) 1-d world with two surfaces A and B, eye E and light L
b) result of transient rendering
c) light seen at E over time

- Input: positions, scattering kernels, initial light emission
- Output: received light power at every point, every direction, and every time
Sensor Model

- Turns ideal worlds into ground truth sensor readings
- Takes into account:
  - Sampled function of time
  - Integration over shutter window
  - Light pulse envelope
  - Discrete photons
- Produces: sequence of energy measurements
New Research Directions

• Applications: do things we could not do before
• Building sensors: capture transient patterns directly
• Theory: generalize and compute
Some Applications

- 3.0D range finding (hidden surfaces)
- Subsurface scattering estimation from time instead of space samples
- Model-based LIDAR applications
Building Sensors

• Existing LIDAR hardware measures the data we need, but throws most of it away
Theory

- **Generalize**
  - Wavelength
  - Subsurface scattering
  - Phosphorescence

- **Compute**
  - Dependency calculation
  - Function representations
  - Augment a common raytracer
Theory

- **Generalize**
  - Wavelength
  - Subsurface scattering
  - Phosphorescence

- **Compute**
  - Dependency calculation
  - Function representations
  - Augment a common raytracer
Conclusion

We have taken initial steps into exploring the effects of the light propagation delay, and called this *Transient Rendering*.

We hope that transient rendering can serve as a principled foundation for future time-of-flight based computer vision techniques.