Transient Rendering

Adam Smith James Skorupski James Davis
University of California Santa Cruz
Department of Computer Science
{amsmith,jskorups,davis}@cs.ucsc.edu

Technical Report UCSC-SOE-08-26
School of Engineering, University of California, Santa Cruz
February 2008

Abstract

The assumption that the speed of light is infinite underlies many established models in computer graphics and vision. Researchers exploring time-of-flight based sensors are moving into a domain that implicitly requires relaxation of this assumption. The classic rendering equation provides a rigorous foundation for understanding light transport, but fails to encompass the transient effects of light propagation at finite speeds. In this paper, we will introduce a physically-relevant generalization of the rendering equation and a method for approximating this equation, and define a summary measure of transient light patterns, which is used as a basis for general sensor model.

Keywords: Computational light science, time-of-flight, rendering equation, light transport, sensor modeling

1 Introduction

There is a growing interest in time-of-flight (TOF) based computer vision applications [Haker 2007, Oprisescu 2007]. In this paper, we seek a general, physical explanation of the measurements made in this field. This is our central motivation. The key observation that drives our work is that models outside of TOF applications assume steady-state light transport. However, to date, there is no established theoretical framework to motivate reasoning about transient light transport that employs computer graphics or vision relevant assumptions and models. At a high level, we aim to use basic physics in combination with core graphics theory to assemble our framework based on a novel formulation of global illumination that does not assume an infinite speed of light.

Critical to our work is the distinction between steady-state and transient light transport. Steady-state transport corresponds to the familiar case in computer graphics or vision, in which the speed of light is conventionally assumed to be infinite (or takes no time to cross any distance). We interpret the value of a pixel as the amount of light received at that pixel. This value is not a function of time. Videos may be interpreted as a sequence of images of different but static worlds. Fundamentally, steady-state light transport describes an amount of energy, a number of photons, or the irradiance at a pixel.

In transient light transport, we assume that the speed of light is some finite value. As light scatters around a scene, it takes different paths, and longer paths take a longer time to traverse. Even a single pulse of light can evolve into a complex pattern in time. Fundamentally, transient transport describes power, a rate of incoming photons, or irradiant flux at a pixel, which, importantly, is measured as a function of time.

In this paper, we will introduce a physically-relevant generalization of the rendering equation, called the *transient rendering equation*, and a method for approximating this equation, called the *cellular approximation procedure*. Next, we will define a summary measure of transient light patterns, called the *transient photometric response function*, and used this function as a basis for general sensor model called the *TPRF sensor*.

2 Related Work

To situate our work with respect to existing research, we looked for research areas that displayed detailed mathematical analysis, light-specific models, and a concern for transient effects. Figure 1 summarizes the overlap of those research areas with our work. We will look at two research areas based on sensing the real world, and one about generating images of synthetic worlds.

	Rigorous Analysis	Light	Transient
SONAR	✓		✓
LIDAR		✓	✓
Rendering Equation	✓	✓	
Transient Rendering	✓	✓	✓

Figure 1. The overlap of related research areas with the properties that our work requires.

2.1 SONAR

SONAR (SOund Navigation And Ranging), is a technique that uses sound propagation in a medium such as air or water to detect and locate remote objects. The speed of sound is six orders of magnitude slower than the speed of light, and therefore easier to detect. Work in SONAR has produced intricate models of the effects of many surfaces with complicated scattering properties [Russell 1996]. These models yield the ability to recover detailed information about the world from samples that describe functions of time. Russell et al have a pipelined analytical framework including emission, propagation, scattering, medium effects, sensors, and data interpretation [1996].

Unfortunately for us, SONAR models are specialized for sound propagation. With sound, diffraction is pervasive, and there are no simple ray sensors or projectors, whereas the opposite is true in computer graphics or vision. The tie-in to our work is that SONAR applications are powerful but cannot tell us about the properties of light propagation.

2.2 LIDAR

LIDAR (LIght Detection And Ranging), is roughly the light analog of SONAR. The speed of light, while exceedingly fast, becomes noticeable over long distances. Short pulses of laser light from an emitter can be used to trigger time-delayed reflections from remote objects in a scene [Kamermann 1993]. Compared with SONAR, LIDAR models are extremely simple. Many assume just a single bounce in the light path and a small number of distinct paths [Jutzi 2006].

Unfortunately for us, LIDAR models cannot handle scenes of arbitrary complexity. The tie-in to our work is that while LIDAR applications do reason about light as a function of time, even taking into account transient effects, they are too simple and specialized for our purposes.

2.3 Rendering equation

In computer graphics, the "rendering equation" refers to a description of steady-state light transport in a scene [Kajiya 1986]. Solutions to this equation are what is called global illumination, and give a physical explanation for observed light. The rendering equation can be seen as the theoretical basis for a vast array of light transport models. It adapts the radiative transport equations from physics with graphics-relevant approximations, such as the existence of a bidirectional scattering function, an infinite speed of light, and a world without diffraction. The rendering equation usually comes in one of two forms, the vacuum rendering equation or volume rendering equation, depending on the representation of the world. The introduction of the rendering equation suggested a new way computing images, by simply evaluating an integral without, leaving the physics to the model.

In linear operator form, the rendering equation is stated as follows: $R = R_0 + GR$, where R is the total radiance, R_0 is the radiance due to light emission, and G is the global light transport operator. Note that this is a recursive definition. Here, G includes geometry and visibility terms. Kajiya presented the original rendering equation in detailed integral form, however we present it operator form here to parallel our definition of the transient rendering equation [1986].

Unfortunately for us, while the rendering equation gives us a very detailed and rigorous analysis of light transport, it does not take into account the effects of transient light propagation. The tie-in to our work is that the rendering equations gives a common story to a huge space of rendering applications, but needs just a small tweak to account for propagation delay.

3 Transient Rendering Equation

In order to rigorously describe light as it scatters around a scene, we must provide a solution for global illumination. Recall that the traditional rendering equation gives an exact description of the light at every point and in every direction within a scene, subject to an infinite speed of light. In this section, we will adapt it to describe a function of time that also depends on the speed of light.

To begin, the radiant flux at a point (in a direction, at a time) is due to the light that is emitted at that point (in that direction, at that time) plus the light that scatters through it from other points a distance away. In operator form, this is written as: $R = R_0 + GR$, where R is the total radiant flux, R_0 is the locally emitted flux, and G is the global light transport operator. Specifically, $R = f(X, \omega, t)$ is a function of a point, direction, and time. Our relation looks identical to the traditional rendering equation, but, importantly, it will describe power instead of energy and take into account the speed of light using the method we will show next.

Global light transport G is the composition of two physical processes, propagation and scattering. Propagation turns radiant flux into irradiant flux, taking light from one surface and across a distance to another. Light from one point arrives at another delayed by a time proportional to its distance. On the other hand, scattering turns irradiant flux into radiant flux, taking incoming light and bouncing it back out. Because scattering takes place around a single point, and no distance is covered, no time delay is incurred. We model scattering using a bidirectional scattering distribution function (BSDF) [Heckbert 1991]. Figure 2 illustrates this process.

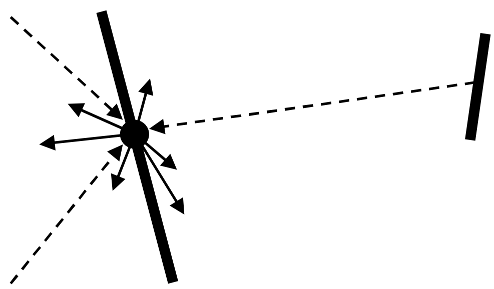

Figure 2. Global light transport is the composition of propagation and scattering. Propagation moves light between surfaces, and scattering redirects light at the point of incidence.

Now we state our transient rendering equation.

$$\boxed{R = R_0 + SPR}$$

We take P to include the geometry and visibility terms, and S to be analogous to scattering in traditional rendering (except that it operates on flux). This relation tells us the power of light at every point, in every direction, at every time. This is indeed the description of global illumination in terms of flux that we required. This relation can be expanded into integral form, but we need to commit to additional details about the world to write it. Recall that Kajiya's original statement of the rendering equation corresponded to what is now known as the vacuum rendering equation, which assumes that the world is populated by infinitely thin surfaces separated by a transparent vacuum.

The transient rendering equation echoes the structure of the traditional rendering equation because both were derived from physics, and are readily specialized into vacuum and volume variants. We have chosen the term “transient” because the key distinction between ours and the traditional rendering equation is that our relation describes the short-term (and necessarily time-varying) effects of the propagation of light. We now have a physically-motivated description of global illumination for arbitrary scenes.

4 Cellular Approximation Procedure

The transient rendering equation is not in a form suitable for direct evaluation. The definition is recursive, and, as such, we need an expression in terms of only given values before we can write out its solutions.

4.1 Derivation

We will begin the derivation with the operator form of the transient rendering equation.

$$R = R_0 + GR$$

Next, we move GR to the left hand side, isolating R_0.

$$R - GR = R_0$$

We can collect $I - G$ as a single operator to apply to R.

$$(I - G)R = R_0$$

We apply the inverse of $I - G$ to both sides, yielding a non-recursive definition.

$$R = (I - G)^{-1}R_0$$

We expand $(I - G)^{-1}$ using the Neumann series.

$$R = (I + G + GG + GGG + \cdots)R_0$$

Finally, distributing this operator with R_0 yields the following expression:

$$\boxed{R = R_0 + GR_0 + GGR_0 + GGGR_0 + \cdots}$$

This expression says that the final flux is the initial flux plus the once-scattered initial flux, plus the twice-scattered initial flux, etc corresponding to intuition.

4.2 Model

Now that we have a non-recursive definition for R, we next need to develop a model that gives a form to operators G and R_0. Again, we adopt a similar model to that of the presentation of the traditional rendering equation. That is, we assume the world is populated by a collection of thin surfaces in a vacuum. Specifically, we will model surfaces as collections of flat interfaces with given geometry and scattering properties (in the form of a BSDF). Note that using a BSDF allows power to be gained or lost at each point on an interface. The specification of interfaces means that this procedure will generate approximations to the transient *vacuum* rendering equation. Because most of the space we model is empty, light propagations freely in a straight line. For generality, we will allow the speed of light to vary in distinct regions of free space. Finally, we say that all interfaces have a known light emission pattern over time. This initial emission function could easily be computed from a simple model of point lights floating in free space.

Next, we will define some notation that is necessary to formally describe our model.

$R_k(X, \omega, t)$	R_k is the radiant flux after k scattering events, X is a point, n is a direction, t is a time.
$I_k(X, \omega, t)$	I_k is the irradiant flux after k scattering events
$R(X, \omega, t), I(X, \omega, t)$	R, I are the sum of R_k, I_k, for all indices $k \geq 0$.
$R^*(X, \omega, t), I^*(X, \omega, t)$	R^*, I^* are R, I extended to be defined in free space.
$R_0(X, \omega, t), I_0(X, \omega, t)$	R_0, I_0 are given for all interfaces, describing the light originating at X.
$K(X, \omega, \omega')$	K is the spatially-varying scattering kernel (BSDF) defined at every point on interfaces. This should include effects of refraction.
$Y(X, \omega)$	Y is the first point encountered heading in direction ω from X.
$E(X, Y)$	E is the geometry term: the dot product of the interface normal at X with direction $Y - X$
$D(X, Y)$	D is the propagation time for light going between X and Y

Recall that we stated $G = SP$. With the above notation, we can now define the form of these operators. P tells us that the k-scattered irradiant flux is the visible k-scattered radiant flux, attenuated by the geometry term, delayed by the propagation time.

$$I_k = PR_k$$

$$I_k(X, \omega, t) = G(X, Y(X, \omega))R_k\left(Y, n, t - D(X, Y(X, \omega))\right)$$

S tells us that the $(k + 1)$-scattered radiant flux is the sum of k-scattered irradiant flux distributed by the scattering kernel.

$$R_{k+1} = SI_k$$

$$R_{k+1}(X, \omega, t) = \int_{\omega'} K(X, \omega', \omega) I_k(X, \omega', t) d\omega'$$

Using these relations, we can build up I_k and R_k inductively for arbitrary large k from I_0 and R_0 alone, which are givens.

4.3 Procedure

Now we present and algorithm for approximating I^* and R^*. To begin, partition the world by adding virtual interfaces so that light may propagate unobstructed in straight lines within each cell.

Note that this implies that these cells are convex. We distinguish three types of interfaces: real, virtual and boundary, illustrated in Figure 3. Real interfaces correspond to actual surfaces in the world and may have arbitrary scattering kernel K. Virtual interfaces are like real interfaces, but have $K(X,\omega,\omega') = \delta(\omega - \omega')$, where δ is the Dirac delta function. That is to say, they are transparent and scattering through them does not affect the direction of light. Boundary interfaces are those not touching another cell of interest, and may be considered to have $K(X,\omega,\omega') = 0$ for simplicity.

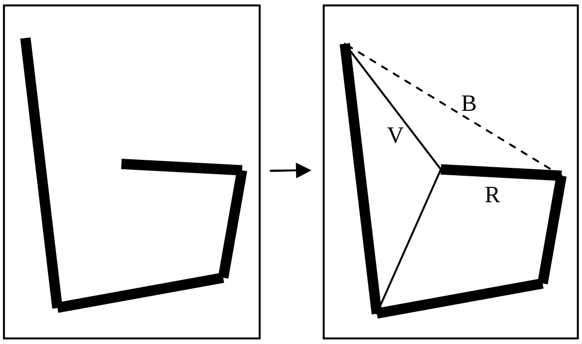

Figure 3. Above illustrates the one possible partitioning of an abstract 2-D world, where segment R is a *real* (physical) interface, V is a *virtual* (transparent) interface, and B is a *boundary* (ignored) interface. Note that all cells are convex.

The core of the procedure is the following. Use P and S defined above to build R_k and I_k in an alternating manner. When the flux represented is as small as desired (due to power loss in scattering), sum all R_k and I_k to form an approximate R and I, respectively. Note that R and I are only defined on interfaces. We can define an I^* for X in free space by collecting (with appropriate delay) light from the final R. Note that $R^* = I^*$ by the assumption of the transparency of free space.

4.4 Discussion

There are a few interesting things to note about our model and procedure. First, adding virtual interfaces may seem to inflate the scattering index. However, this process ensures that the visibility term normally considered in rendering is always 1, and thus ignored in our model. This avoids explicit shadow calculation and does not otherwise change the results. Next, keep in mind that as the flux functions are functions of a point, direction, and time, they need a suitable representation in practice to allow their updates to be computed. The accuracy of these representations will dominate the overall accuracy of the procedure because the procedure itself is derived from an analytical framework, which, in some sense, provides the *correct* answer.

5 Transient Photometric Response Function

Recall that our original goal was to explain sensor measurements. The transient rendering equation, by itself, tells us far more than we need to know for TOF-based computer vision. We can imagine practical sensors that report flux over time at a specific point and in a specific direction in response to a specific light source. In this section we will introduce a function which will allow us to model exactly what a sensor measures.

If we look at a restriction of the total irradiant flux function, I^*, to a fixed eye point E and a viewing direction ω, and assume a scene is lit only by a single, delta function impulse emission R_0, we have the following definition for the *transient photometric response function (TPRF)*:

$$\boxed{\mathrm{TPRF}(t) = I^*(E,\omega,t) = R\Big(Y(E,\omega),\omega,t - D\big(E,Y(E,\omega)\big)\Big)}$$

We chose this name carefully. The term *transient* refers to the transient rendering equation, from which this function is derived. The term *photometric* refers to the emphasis on the intensity of light, as opposed to electromagnetic radiation with specific wavelengths. *Response* indicates that we are concerned with the results of an impulse. Finally, *function* emphasizes that the transient photometric response is not just a single value, but a function of one variable: time.

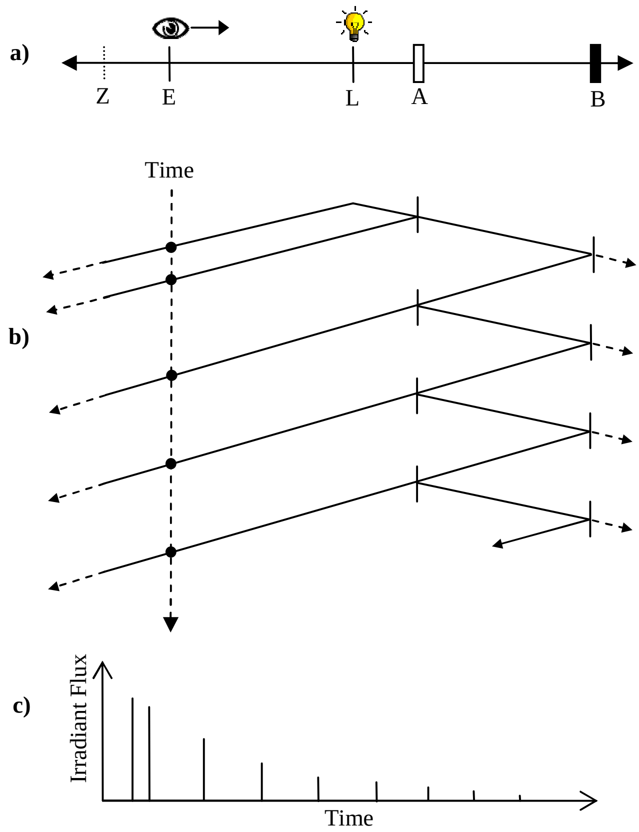

Figure 4. Here we show the derivation of a TPRF in a simple scene. (a) 1-D world with real interfaces A and B, eye point E, light source L, and boundary Z (b) Result of transient rendering (c) TPRF at E looking to the right

To see how a TPRF can be derived from a description of the world, we will work through a simple example, illustrated in Figure 4. We begin with a simple one dimensional world populated by two interfaces (points) A and B. Suppose that A is partially transparent, B is opaque, and both interfaces are partially reflective. Next, we identify an eye point E and a light point L. Additionally, we have to identify another point Z so that we may form a cell enclosing E and L. Z is an example of a boundary interface, while A and B are real.

From this world we would like to derive an I^* so that we can define the TPRF at E, say, looking to the right (in direction $\hat{n}$). Per our approximation procedure, we have a way of defining I^* in terms of givens, such as the initial radiant flux at every interface. A delta function impulse of light begins at L, stated as $R_0(E,\omega,t) = \delta(t-0)$, for $\omega \in \{-\hat{n},+\hat{n}\}$ (where "left" and "right" are the only directions in a 1-D world). We can use this to define the irradiant flux I_0 at A by simply delaying the light from E, giving us $R_0(A,\hat{n},t) = \delta\big(t - D(L,A)\big)$, with I_0 and R_0 constant zero elsewhere. Now, using the transient rendering equation, we can build up I_k and R_k at A and B for arbitrary k to assemble a suitable I. Note that boundary interface Z can be ignored here because it does not participate in scattering. Finally, we can define I^* from R. At this point, we know the total flux at every point in the cells Z-A and A-B, in both directions, at every time. For such a simple world, this evolution produces a simple ray tree, illustrated in section b of Figure 4. Now we are ready to form the TPRF. The TPRF at E is simply the following:

$$\mathrm{TPRF}(t) = I^*(E,-\hat{n},t) = R(A,-\hat{n},t - D(E,A)).$$

Even in this simple example, considerable information about the scene is encoded in the TPRF, particularly some that any single measurement could miss. Note in section c in Figure 4 that the distance A-B is evident in the separation of pulses in the TPRF.

The TPRF is (almost) directly sensed by many existing LIDAR applications. However, usually only simple properties of the function are examined, such as the time delay before the first peak. As presented, the TPRF model corresponds to a highly idealized sensor, but has important properties we will see in the next section that allow us to form a much more realistic sensor model.

6 TPRF Sensor Model

When we began this work, we sought to explain sensor measurements. Realistically, sensors are man-made physical devices subject to engineering limitations. The trigger light sources in sensors we would like to model do not (and physically cannot) pulse for infinitely short periods of time. Instead, their output is governed by some envelope in time. Additionally, when collecting light to form a sample, sensors integrate flux over a period of time. Furthermore, light travels in discrete photons, not as idealized continuous flux. Finally, real devices have myriad internal sources of electronic noise. Taken together, these circumstances form a distinct departure from the assumptions of transient rendering.

We address the above concerns in the design of our sensor model. We allow the trigger light source to have an arbitrary envelope in time, given by some function $\mathrm{Light}(t)$. The sensitivity to flux over time for the collection of a single sample is also allowed an arbitrary envelope $\mathrm{Exposure}_i(t)$, for sample i. We ignore the discrete nature of light because, in practice, we expect a vast number of photons to be received for each measurement. Finally, we address noise sources with an all-encompassing additive Gaussian white noise term.

Now we can state our full sensor model, which we call the *TPRF sensor*. In the following expression, M_I is the *i*'th measurement in a sequence, and $*$ denotes convolution:

$$\boxed{\mathrm{M}_i = \max\left(0, \mathrm{Noise} + \int_t (\mathrm{TPRF} * \mathrm{Light})(t)\ \mathrm{Exposure}_\mathrm{i}(t)\ dt\right)}$$

To understand this definition, let us examine its components. First consider the function $f(t) = \mathrm{TPRF}(t) * \mathrm{Light}(t)$. This corresponds to the idealized observed total flux resulting from an arbitrary emission pattern at the trigger light source. We can do this because the TPRF was formulated in terms of a single impulse, and convolution allows us to synthesize any waveform from a superposition of impulses. This is equivalent to feeding the arbitrary emission pattern into the original transient rendering process. Next, this result is accumulated in the sensor over time. Simply multiplying $f(t)$ with the exposure sensitivity gives us an effective measured flux function. Now, all that remains is for us to integrate over time to find the area under this curve. The result, a value measured in terms of energy, need only be perturbed by a noise value before representing a hypothetical measurement from our model sensor.

It is interesting to note that the example presented in the previous section (illustrated in Figure 4) could be interpreted as a simple LIDAR experiment. In this case, E and L could be the locations of the photo sensor and pulse-emitting laser on an airborne platform, respectively, and A and B could represent a partially transparent foliage layer and opaque ground layer, respectively. Knowing the geometry of this experiment, transient rendering could produce a TPRF. In Figures 5, 6 and 7, we illustrate the process of generating hypothetical measurements from a given TPRF. The first shows a representative TPRF. The second shows the result of convolving the TPRF with the light emission envelope. In this case, we chose a Gaussian curve. Finally, the last shows a sequence of simulated measurements in the case of a box exposure envelope. These measurements could be compared with real sensor measurements in a physical LIDAR experiment.

This is a very general but physically motivated sensor model that can be used to produce reference measurements by simulation. The TPRF encapsulates all of the world dependence of the model, and thus a single simulated TPRF can be used with several different sensor profiles. We wanted some general, physical explanation of the measurements we make, and now we have it.

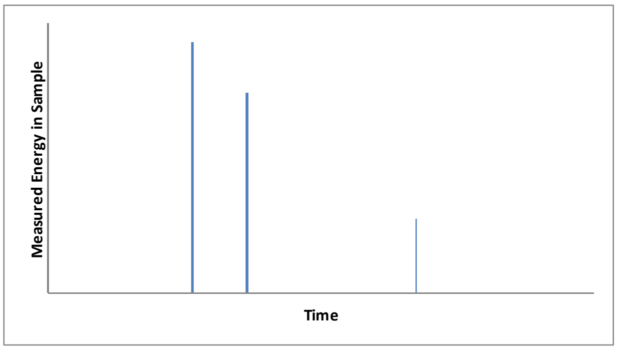

Figure 5. A representative TPRF derived from the scene in Figure 4 (spikes represent weighted delta function impulses)

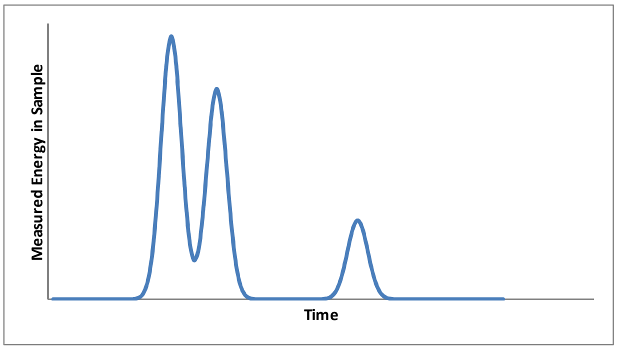

Figure 6. Result of convolving the TPRF from Figure 5 with a light emission envelope

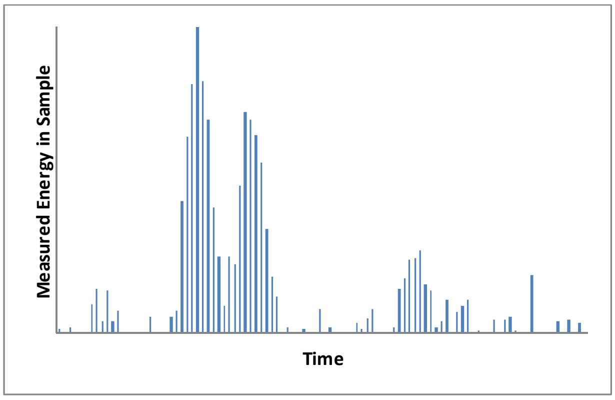

Figure 7. A sequence of hypothetical measurements derived from the measured flux function in Figure 6.

7 Future Research Directions

Our transient rendering framework is a first step in rigorously exploring the transient effects of light propagation in a vision/graphics setting. Four new directions of research that are opened up from this work are generalizing this model, computationally implementing it, building a sensor, and exploring new applications.

To generalize transient rendering, effects such as subsurface scattering could be modeled by moving directly to transient *volume* rendering, or generating appropriate geometric details (myriad small interfaces in the interior of objects) and using the cellular approximation procedure presented in this paper. The derivation of our approximation procedure assumes that interfaces cannot story energy over time. If we allow an excitation at each point on interfaces, we may take into account phosphorescence. Furthermore, wavelength could be properly treated by taking into account dispersion at interfaces. A solution to a transient rendering equation that included proper treatment of wavelengths would form a complete plenoptic function.

Before computationally implementing our framework, it would be necessary to develop additional details. First, it would be necessary to decide on representations for the flux functions used in the approximation (such as R_k, I_k). The functions could be represented as analytic expressions or collections of point samples which correspond to photons recorded at a specific point, direction, and time. Next, our approximation procedure limits local light transport to individual cells. Presumably, some procedure exists for telling which exact regions are needed to calculate a specific TPRF. This development of a method for computing dependencies would prove invaluable in a practical simulator.

Next, sensors could be built that attempt to directly measure the TPRF, to support TPRF-based applications. This is a realistic assumption, given that existing LIDAR systems measure data similar to what would be needed.

Finally, this work opens up new application areas. Transient rendering may allow 3.0D range finding (shape recovery including hidden surfaces), or uncover implicit assumptions in traditional 2.5D range finding. If modeled, subsurface scattering parameters may be recoverable from samples in time, instead of space. Additionally, decomposing the TPRF into single-scatter layers may reveal interesting structures of a scene.

8 Conclusion

Using only basic physics in combination with core graphics theory, we have taken initial steps into exploring the effects of taking propagation delays for light into account and called this *transient rendering*. In doing this, we have defined a physically-relevant generalization of the rendering equation by modifying the propagation operator to account for the speed of light and called this generalization the *transient rendering equation*. We have also defined a method for approximating this equation in terms of parameters for quite general worlds based on a convex partitioning of the scene and called this the *cellular approximation procedure*. Next, we defined a summary measure of transient light patterns comparable to the output of a highly idealized sensor and called this the *transient photometric response function*. We then utilized this function in the definition of a sensor model incorporating several physical phenomena, called the *TPRF sensor*. Finally, we proposed a wide array of new research directions, including previously unreachable applications.

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

References

HECKBERT, P. 1991. *Simulating Global Illumination Using Adaptive Meshing.* PhD thesis, University of California, Berkeley. 26.

JUTZI, B., STILLA, U. 2006. Precise range estimation on known surfaces by analysis of full-waveform laser. *In Proceedings of Photogrammetric Computer Vision PCV 2006*. International Archives of Photogrammetry and Remote Sensing Vol. 36.

HAKER, M., BÖHME, M., MARTINETZ, T., BARTH, E. 2007. Geometric invariants for facial feature tracking with 3D TOF cameras. In *IEEE Sym. on Signals Circuits & Systems (ISSCS*), session on Alg. for 3D ToF-cameras, pages 109–112.

KAJIYA, J. T. 1986. The Rendering Equation. In *Proceedings of the 13th Annual Conference on Computer Graphics and Interactive Techniques.* Evans, D. C. and Athay, R. J., Eds. ACM SIGGRAPH 1986. 143-150.

KAMERMANN, G. W. 1993. Laser Radar, *Active Electrical Optical Systems, The Infrared & Electro-Optical Systems Handbook.*

OPRISESCU, S., FALIE, D.,CIUC, M., BUZULOIU, V. 2007. Measurements with ToF cameras and their necessary corrections. In *IEEE Sym. on Signals Circuits & Systems (ISSCS), session on Alg. for 3D ToF-cameras,* pages 221–224.

RUSSELL, G. T., BELL, J. M., HOLT, P. O., CLARKE, S. J. 1996. Sonar Image Interpretation and Modeling. In *Proceedings of the 1996 Symposium on Autonomous Underwater Vehicle Technology, 1996. AUV '96.* 317-324.