Submitted to

Workshop on Multimedia Technology

An Efficient Spectral Rendering Technique for Phong Illumination Model

Chung-Ming Wang[1] Jin-Ren Chern[1,2] JiunnShyan Lee[1]

cmwang@cs.nchu.edu.tw jrchern@cs.nchu.edu.tw jslee@cs.nchu.edu.tw

Phone:886-0422840497 ext 915, 904

[1]Institute of Computer Science, National Chung-Hsing University

250 Kuo-Kwant Road Taichung City Taiwan, R.O.C.

[2]Department of Information Management, Chien Kuo Institute of Technology

1, Chieh Shou N. Road Changhua City 500, Taiwan, R.O.C.

Abstract

Light is an electromagnetic wave, for some wavelength dependent phenomena, such as dispersion, interference, diffraction and fluorescence, cannot be simulate by means of the color-based approaches of rendering techniques. So far, spectral rendering is a solution to simulate these wavelength dependent phenomena. Nevertheless, it's a tedious and an inefficient job to construct spectral surface reflectance curves by hand. In this paper, we propose an efficient rendering algorithm to render a sphere object with spectral Phong reflectance with real-time response. The adaptive resolution scheme is employed to find better tabulating entries positions. The purpose of this algorithm is to be combined with an interactive tool of spectral Phong reflectance editor, and to provide users with WYSISWYG(What You See IS What You Get) functionality. The results show that the algorithm achieves the requirement of real-time response and maintaining accuracy.

*Keywords:*spectral rendering, Phong model, wavelength dependent, specular reflection, diffuse reflection

Contact Author: Jin-Ren Chern(陳金仁)

彰化市 500 介壽北路 1 號建國技術學院　資管系

Mobile phone:0919709247

Phone: (04)7224676 ext 3601

Fax:(04)7291952

Email:jrchern@cs.nchu.edu.tw

An Efficient Spectral Rendering Technique for Phong Illumination Model

Chung-Ming Wang[1] Jin-Ren Chern[1,2] JiunnShyan Lee[1]
cmwang@cs.nchu.edu.tw jrchern@cs.nchu.edu.tw jslee@cs.nchu.edu.tw
Phone:886-0422840497 ext 915, 904
[1]Institute of Computer Science, National Chung-Hsing University
250 Kuo-Kwant Road Taichung City Taiwan, R.O.C.
[2]Department of Information Management, Chien Kuo Institute of Technology
1, Chieh Shou N. Road Changhua City 500, Taiwan, R.O.C.

Abstract

Light is an electromagnetic wave, for some wavelength dependent phenomena, such as dispersion, interference, diffraction and fluorescence, cannot be simulate by means of the color-based approaches of rendering techniques. So far, spectral rendering is a solution to simulate these wavelength dependent phenomena. Nevertheless, it's a tedious and an inefficient job to construct spectral surface reflectance curves by hand. In this paper, we propose an efficient rendering algorithm to render a sphere object with spectral Phong reflectance with real-time response. The adaptive resolution scheme is employed to find better tabulating entries positions. The purpose of this algorithm is to be combined with an interactive tool of spectral Phong reflectance editor, and to provide users with WYSISWYG(What You See IS What You Get) functionality. The results show that the algorithm achieves the requirement of real-time response and maintaining accuracy.

Keywords:spectral rendering, Phong model, wavelength dependent, specular

reflection, diffuse reflection

1. Introduction

The ultimate goal for photo-realistic rendering is to produce images identical to the human being perception of real world objects and environments[1, 2]. Great achievements have been made for photo-realistic rendering relative researches through involvement of efforts in developing methods for photo-realistic rendering. Unfortunately, the most researches for photo-realistic rendering are using color-based approaches, for example—RGB triplet, to synthesize the image of real world objects and environments. For some applications, the color-based approaches of rendering techniques can offer fruitful results. However, this cannot always be true for all of the rendering tasks. Since light is an electromagnetic wave, for some wavelength dependent phenomena, such as dispersion, interference, diffraction and fluorescence[3][4], can not be simulate by means of the color based approaches of rendering techniques. For such wavelength dependent phenomena, distinct wavelength of light generates different behaviors and occurrences. So far, spectral rendering is a solution to simulate these wavelength dependent phenomena.

Surface reflectance is an important factor for synthesizing images. Since a reflectance curve for spectrum based approaches consist of plenty numerals for fractions of distinct wavelength reflectance. Nevertheless, someone can’t perceive colors of reflectance at the moment he or she construct surface reflectance curve. Therefore, constructing a surface reflectance curve for spectral rendering is not only a bothersome job, but also a blind work. No one knows how it looks like, when someone ask “What does the surface reflectance curve look like under a white light illuminated?”. Thus we thirst for a tool to help constructing surface reflectance curve for the reason of convenience and accuracy. We expect that this tool is capable of displaying the color of surface reflectance curve real-time. Besides, in order to construct surfaces reflectance curves not only diffuse surfaces but also shiny surfaces, this tool would incorporate the Phong reflectance model[5]. This would be introduce a demand of a algorithm, which is capable of fast rendering spectrum based with Phong model objects, for calculating and displaying surfaces reflectance curves edited in a real time manner. Unfortunately, spectrum based rendering is computationally expensive. Moreover, calculation of Phong model is also time consuming because the specular term of Phong model which involves an exponentiation. Adding these two terms dealt with previously up, increases the burden of efficiency for rendering. Thus how to accelerate the rendering process of spectral Phong model to be capable of rendering the image of surfaces reflectance curves in a real time manner is not only an interested but also a meaningful job. Several methods have been proposed to reduce the computational time of exponentiation required by the Phong model. The method which is computing $e^{n \ln t}$ (n for exponent, t for $\cos\alpha$, refer to figure 1 to see α) , or computing the product of successively multiply t n times for reducing computational time is the most straightforward method but is often too slow. Schlick[6] exploit the expression of rational function to approximate the term -- $\cos^n \alpha$ in the Phong model. The rational function technique, which did not consider properties of spectral computation, it can't meet the requirement of real-time response, although the rational function technique reduce the computational time required by the Phong model. This fact spurred us to develop a novel algorithm for fast spectral Phong model rendering.

In this paper, we propose a novel algorithm for fast spectral Phong model rendering. The purpose of this algorithm is to provide an interactive ability for users to edit surfaces reflectance curves of Phong model with real-time responses. This algorithm uses an importance-driven division method to split the interval for tabulating according to the intensity of specular colors. In addition, we present a parabolic interpolation technique instead of linear interpolation method to improve the accuracy for interpolating the non-tabulated data. We implement a spectral Phong model visualization tool, which incorporates the fast spectral Phong rendering algorithm proposed in this paper, to construct and edit surface reflectance curves. The result shows that the algorithm proposed by us can be not only real-time response but also with no degradation of accuracy.

2. Assumptions for the rendering process

For simplicity, some assumptions for the rendering process are listed below:

(1). The light source for this rendering technique is only one point light source.

(2). The only one object for rendering is a sphere, which reflects the surface reflectance curve of spectral Phong model constructed.

(3). The spectral Phong model involves the following items.

(a). Diffuse coefficients spectral curve: This is a collection of data for fractions of diffuse part of various wavelength.

(b). Specular coefficients spectral curve: This is a collection of data for fractions of specular part of various wavelength.

(c). Specular exponent spectral curve: This is a collection of data for specular exponents of various wavelength.

(4). The spectral curves dealt with previously can be altered dynamically.

(5). The light source can be altered dynamically.

3. An efficient rendering algorithm for spectral Phong model

3.1 **Spectral Phong model**

Specular reflection is an important factor for rendering shiny objects. Specular reflection is when the reflection is stronger in one viewing direction, i.e., there is a bright spot, called a specular highlight. Phong model is an empirical result for modeling light reflection of diffuse, especially specular objects. The intensity of an object containing diffuse and specular light reflection, incorporating Phong model, can be expressed as follows[7]:

$$I = \underbrace{I_a k_a}_{ambient} + \underbrace{f_{att} I_p k_d \cos\theta}_{diffuse} + \underbrace{f_{att} I_p k_s \cos^n \alpha}_{specular} \quad (1)$$

where:

f_{att} : $Attenuation = \max\left(\frac{1}{c_1 + c_2 d_L + c_3 d_L^{\ 2}}, 1\right)$

c_1, c_2 *and* c_3 *are user defined constants associated with the light source*

d_L *the distance the light travels from the point source to the surface*

n : *specular – reflection exponent*

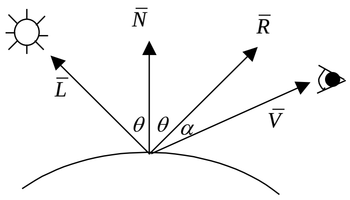
I_a : *ambient light intensity*

I_p : *point light source's intensity*

k_d : *diffuse – reflection coefficient*

k_s : *specular – reflection coefficient*

α : *refers to figure* 1



$\overline{L}$: *light direction*

$\overline{N}$: *surface normal*

$\overline{R}$: *ideal specular reflection*

$\overline{V}$: *viewing direction*

Figure 1 Specular reflection

For simplicity, only diffuse part and specular part of equation 1 would be considered.

Combining Phong model with spectral rendering concept forms the spectral Phong model. Equations of the spectral Phong model need an extra parameter of λ to represent the light intensity for various wavelength separately. Equation (2) is the light intensity expression for spectral Phong model.

$$I(\lambda) = \underbrace{f_{att} I_p(\lambda) k_d(\lambda) \cos\theta}_{diffuse} + \underbrace{f_{att} I_p(\lambda) k_s(\lambda) \cos^{n(\lambda)} \alpha}_{specular} \tag{2}$$

3.2 Importance-driven division method

RGB triplets color system is a most popular system for most of the computer color monitors. Thus the final output of the visualization is also RGB-based system. The result of rendering, which is in spectral model, must be transformed into RGB triplets for displaying. Equation (3) is a formula for translating spectral model into CIEXYZ[8], where k is a normalizing constant, $\phi(\lambda)$ is spectral radiance, $\bar{x}(\lambda), \bar{y}(\lambda)$ and $\bar{z}(\lambda)$ are the so-called color matching functions for X, Y and Z primaries, respectively. Equation (4) is a discrete version of equation (3), where [$\lambda_{\min}$, $\lambda_{\max}$] denotes the electromagnetic wave in the visible range, normally from 380nm to 780 nm. Equation (5), which is a two-matrixes multiplication, is performed to obtain RGB triplets from CIEXYZ primaries[9].

$$\left.\begin{aligned} X &= k\int_{\lambda} \phi(\lambda)\bar{x}(\lambda)d\lambda \\ Y &= k\int_{\lambda} \phi(\lambda)\bar{y}(\lambda)d\lambda \\ Z &= k\int_{\lambda} \phi(\lambda)\bar{y}(\lambda)d\lambda \end{aligned}\right\} \tag{3}$$

$$\left.\begin{aligned} X &= k\sum_{\lambda=\lambda_{\min}}^{\lambda_{\max}} \phi(\lambda)\overline{x}(\lambda) \\ Y &= k\sum_{\lambda=\lambda_{\min}}^{\lambda_{\max}} \phi(\lambda)\overline{y}(\lambda) \\ Z &= k\sum_{\lambda=\lambda_{\min}}^{\lambda_{\max}} \phi(\lambda)\overline{z}(\lambda) \end{aligned}\right\} \tag{4}$$

$$\begin{bmatrix} R & G & B \end{bmatrix} = \begin{bmatrix} X & Y & Z \end{bmatrix} \begin{bmatrix} XYZ \\ TO \\ RGB \end{bmatrix} \tag{5}$$

Base on the previous description, the X, Y and Z value of arbitrary point for spectral Phong model can be calculated as follow:

$$\begin{aligned} X &= k\sum_{\lambda=\lambda_{\min}}^{\lambda_{\max}} (f_{att}I_p(\lambda)k_d(\lambda)\cos\theta + f_{att}I_p(\lambda)k_s(\lambda)\cos^{n(\lambda)}\alpha)\overline{x}(\lambda) \\ Y &= k\sum_{\lambda=\lambda_{\min}}^{\lambda_{\max}} (f_{att}I_p(\lambda)k_d(\lambda)\cos\theta + f_{att}I_p(\lambda)k_s(\lambda)\cos^{n(\lambda)}\alpha)\overline{y}(\lambda) \\ Z &= k\sum_{\lambda=\lambda_{\min}}^{\lambda_{\max}} (f_{att}I_p(\lambda)k_d(\lambda)\cos\theta + f_{att}I_p(\lambda)k_s(\lambda)\cos^{n(\lambda)}\alpha)\overline{z}(\lambda) \end{aligned} \tag{6}$$

The f_{att} and $\cos\theta$ term in the diffuse part of equation (6) is irrelevant to the parameter λ, but is relevant to the geometry of points rendered, in another word the previous two terms is relevant to the distance and direction between points rendered and light source. Thus equation (6) can be rewritten as equation (7):

$$\begin{aligned} X &= k\cdot f_{att}\cdot\cos\theta\sum_{\lambda=\lambda_{\min}}^{\lambda_{\max}} (I_p(\lambda)k_d(\lambda))\overline{x}(\lambda) + k\sum_{\lambda=\lambda_{\min}}^{\lambda_{\max}} (f_{att}I_p(\lambda)k_s(\lambda)\cos^{n(\lambda)}\alpha)\overline{x}(\lambda) \\ Y &= k\cdot f_{att}\cdot\cos\theta\sum_{\lambda=\lambda_{\min}}^{\lambda_{\max}} (I_p(\lambda)k_d(\lambda))\overline{y}(\lambda) + k\sum_{\lambda=\lambda_{\min}}^{\lambda_{\max}} (f_{att}I_p(\lambda)k_s(\lambda)\cos^{n(\lambda)}\alpha)\overline{y}(\lambda) \\ Z &= k\cdot f_{att}\cdot\cos\theta\sum_{\lambda=\lambda_{\min}}^{\lambda_{\max}} (I_p(\lambda)k_d(\lambda))\overline{z}(\lambda) + k\sum_{\lambda=\lambda_{\min}}^{\lambda_{\max}} (f_{att}I_p(\lambda)k_s(\lambda)\cos^{n(\lambda)}\alpha)\overline{z}(\lambda) \end{aligned} \tag{7}$$

Fortunately, the time consuming computation of diffuse part in equation (7) -- $\sum_{\lambda=\lambda_{\min}}^{\lambda_{\max}} (I_p(\lambda)k_d(\lambda))\overline{x}(\lambda)$ is irrelevant to the geometry of point rendered, it is all the same for all point rendered. Hence, pre-calculating the term dealt with previously, would save much time in rendering.

The calculation of specular part is the most time consuming calculation of equation (7). We assume that X_{sp}, Y_{sp} and Z_{sp} ,as shown in equation (8), represent the radiance, which come from the contribution of specular part of equation (7), of CIEXYZ respectively. Beside, R_{sp}, G_{sp} and B_{sp} denote the RGB value for X_{sp}, Y_{sp} and Z_{sp}.

$$
\begin{aligned}
X_{sp} &= k \sum_{\lambda=\lambda_{\min}}^{\lambda_{\max}} (f_{att} I_p(\lambda) k_s(\lambda) \cos^{n(\lambda)} \alpha) \overline{x}(\lambda) \\
Y_{sp} &= k \sum_{\lambda=\lambda_{\min}}^{\lambda_{\max}} (f_{att} I_p(\lambda) k_s(\lambda) \cos^{n(\lambda)} \alpha) \overline{y}(\lambda) \\
Z_{sp} &= k \sum_{\lambda=\lambda_{\min}}^{\lambda_{\max}} (f_{att} I_p(\lambda) k_s(\lambda) \cos^{n(\lambda)} \alpha) \overline{z}(\lambda)
\end{aligned}
\tag{8}
$$

Combining color matching function with XYZ-to-RGB matrix, we obtain a spectra-to-RGB matrix as equation (9).

$$
\begin{bmatrix} \text{Spectra} \\ to \\ RGB \end{bmatrix} = \begin{bmatrix} \overline{x}(\lambda_{\lambda_{\min}}) & \overline{y}(\lambda_{\lambda_{\min}}) & \overline{z}(\lambda_{\lambda_{\min}}) \\ \overline{x}(\lambda_{\lambda_{\min}+1}) & \overline{y}(\lambda_{\lambda_{\min}+1}) & \overline{y}(\lambda_{\lambda_{\min}+1}) \\ . & . & . \\ . & . & . \\ \overline{x}(\lambda_{\lambda_{\max}}) & \overline{y}(\lambda_{\lambda_{\max}}) & \overline{y}(\lambda_{\lambda_{\max}}) \end{bmatrix} \begin{bmatrix} XYZ \\ to \\ RGB \end{bmatrix} \tag{9}
$$

Hence, we can calculate R_{sp}, G_{sp} and B_{sp} as follow:

$$
\begin{bmatrix} R_{sp} & G_{sp} & B_{sp} \end{bmatrix} = k \cdot f_{att} \cdot \left[I_p(\lambda_{\lambda_{\min}}) k_s(\lambda_{\lambda_{\min}}) \cos^{n(\lambda_{\lambda_{\min}})} \alpha \;\; I_p(\lambda_{\lambda_{\min}+1}) k_s(\lambda_{\lambda_{\min}+1}) \cos^{n(\lambda_{\lambda_{\min}+1})} \alpha ... I_p(\lambda_{\lambda_{\max}}) k_s(\lambda_{\lambda_{\max}}) \cos^{n(\lambda_{\lambda_{\max}})} \alpha \right] \cdot \begin{bmatrix} Spectra \\ to \\ RGB \end{bmatrix} \tag{10}
$$

The simplest way to approximate the specular part for RGB intensity value is to use a uniform interval division tabulation method. It is quite simple and fast, but it introduces a significant error. As figure 2 shows, (a) and (c) are rendered according to equation (6). (b) and (d) are rendered by uniform interval tabulating. It is obvious that (b) and (d) are quite different within the highlight specular area. To reduce the error resulted from uniform interval tabulating would consume a lot of memory.

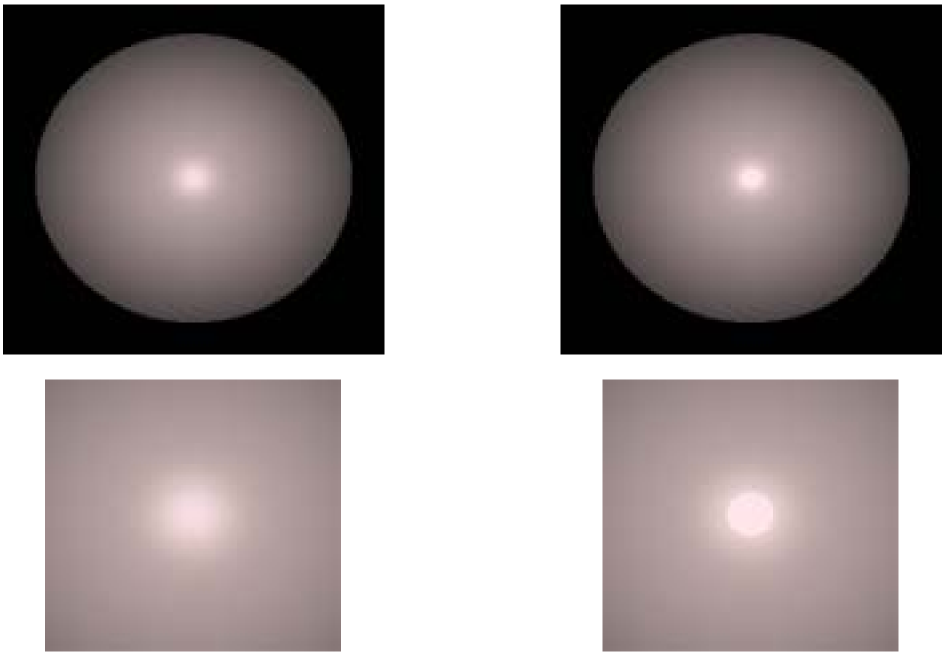


Figure 2 (a). Normal, distant view image, (b). Uniformly tabulating, distance view image, (c). Normal, close view image, and (d). Uniformly tabulating, close view image

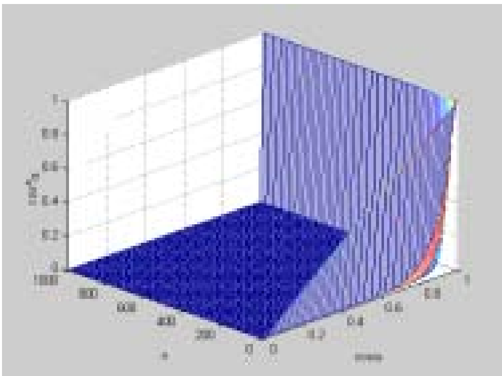


Figure 3 Plot of $\cos^n \alpha$

Figure 3 is a plot for the relationship among n, $\cos\alpha$ and $\cos^n \alpha$. It's obvious that for sufficiently large numbers--n, most of the values of $\cos^n \alpha$ are approaching to 0, but only few values of $\cos^n \alpha$ are greater than 0. To give consideration to computing time and accuracy of results, we develop an adaptive resolution scheme to tabulate the value for R_{sp}, G_{sp} and B_{sp}. As shown in figure 4, the basic ideal of the adaptive resolution scheme is that the density of resolutions or intervals of the tabulation are according to values of the $\cos^n \alpha$ term. The resolution is lower while values of $\cos^n \alpha$ are smaller. On the contrary, the resolution is higher while values of $\cos^n \alpha$ are larger.

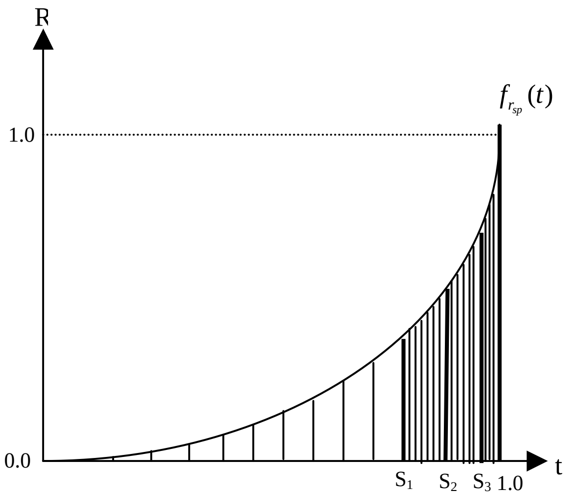


Figure 4 adaptive resolution scheme

For simplicity, $\cos\alpha$ is represented as *t*. We then define three functions $f_{r_{sp}}(t)$, $f_{g_{sp}}(t)$ *and* $f_{b_{sp}}(t)$, which are functions of *t*, as shown in equation (11).

$$\begin{bmatrix} f_{r_{sp}}(t) & f_{g_{sp}}(t) & f_{b_{sp}}(t) \end{bmatrix} = k \cdot f_{att} \cdot \begin{bmatrix} I_p(\lambda_{\lambda_{\min}})k_s(\lambda_{\lambda_{\min}})t^{n(\lambda_{\lambda_{\min}})} & I_p(\lambda_{\lambda_{\min}+1})k_s(\lambda_{\lambda_{\min}+1})t^{n(\lambda_{\lambda_{\min}+1})} \ldots I_p(\lambda_{\lambda_{\max}})k_s(\lambda_{\lambda_{\max}})t^{n(\lambda_{\lambda_{\max}})} \end{bmatrix} \cdot \begin{bmatrix} Spectra \\ to \\ RGB \end{bmatrix} \quad (11)$$

In the following text, we will depict the calculation for R(red) only, the other two colors of G(green) and B(blue) are deduced as R(red). Assuming that the tabulation is divided into m intervals, and the m intervals are grouped into p segments, which are numbered as S_0, S_1, S_2,…,S_{p-1}. For clarity, we restrict that m should be divisible by p. For every segment there is $\frac{1}{p}$ weighting and $\frac{m}{p}$ intervals. For adaptive resolution scheme, we need to find the values for every segment S_q such that W(S_q)= $\frac{1}{p} \cdot q$, where W(t) denotes the weight at t. In other words, we need to find the values of S_0, S_1, S_2,…,S_{p-1}, S_p, to result in that every interval holds the equal weight. Thus we define a function, which is called importance ratio function or weight function, written as *ir(t). ir(t)* denotes the importance ration at point t. Mathematically,

$$ir(t_1) = \frac{\int_0^{t_1} f^{app}(t)dt}{\int_0^1 f^{app}(t)dt} \qquad (12)$$

For ease of computation, the function $f^{app}(t)$ is a function used to approach the original function of $f_{r_{sp}}(t)$. Equation (13) is the expression of $f^{app}(t)$.

$$f^{app}(t) = t^{an} \qquad (13)$$

$$where$$

$$anistheaverageof \ \exp onen$$

$$an = \sum_{\lambda_{\min}}^{\lambda_{\max}} n(\lambda)$$

Thus equation (12) can be further rewritten as:

$$ir(t) = \frac{\int_0^{t_1} t^{an}dt}{\int_0^{1.0} t^{an}dt} = \frac{\frac{t^{an+1}}{an+1}}{\frac{1^{an+1}}{an+1}} = t^{an+1} \qquad (14)$$

In addition, the inverse importance ratio function written as *iir(w),* denotes a mapping from weight to *t* value. Generally, *iir(ir(t))=t*, *ir(iir(w))=w*. Equation (15) solves *t* value for arbitrary *w*, where $0 \le w \le 1.0$.

$$\text{Since } ir(t) = w = t^{an+1}, \quad \text{Thus } t = \sqrt[an+1]{w} \tag{15}$$

After the values of all segment of S_1, S_2,…,S_{p-1}, S_p have been solved, each interval value for table look up could be determined. Since each segment covers the same number of intervals, each interval value for table look up could be determined as equation (16).

$$t_{i\cdot p+j} = \begin{cases} 0, \ for \ i=0, \ j=0 \\ S_i + j\cdot\frac{p}{m}\cdot(S_{i+1}-S_i), \ for \ 0 \le i < p, \ 0 \le j < \frac{m}{p} \\ 1, \ for \ i\cdot p + j = m \end{cases} \tag{16}$$

4. Results

Figure 5, figure 6 and figure 7 are three combinations of reflectance, named R1, R2 and R3 respectively. (g), (h), (i) and (j) of figure 5, figure (6) and figure (7) are spectral curve of point light source, spectral curve of diffuse coefficients of Phong model, spectral curve of specular coefficients of Phong model and spectral curve of exponents of Phong model respectively. In figure 5, figure 6 and figure 7, (a) and (d) are images without tabulating, (b) and (e) are images rendered by using the algorithm proposed in this paper with 200 tabulating entries, (c) and (f) are images rendered by using uniform interval tabulating with 200 tabulating entries, (d), (e) and (f) are the zoom in version images of (a), (b) and (c), respectively. Using (a) and (d) as standard images, compare (e) with (f) and (b) with (c), it is obvious that (c) and (f) in three images described above generate significant errors. RMS error and response time for rendering of three rendering types (without tabulating, our algorithm and uniform interval tabulating) are summarized in Table 1. These data are measured under an AMD Athlon 700MHZ CPU, which is running Micorsoft Windows 2000 OS. According to Table 1, (b) and (e) are with an acceptable real-time response and an acceptable RMS error, regardless of combination of reflectance. Obviously, Table 1 give the information of examining superiority of our algorithm.

Table 1 response time versus RMS error of various rendering methods

Combination	Images	RMS error	Response time
R1	(a)	Standard image	3.658 seconds
	(b)	0.067	0.382 second
	(c)	0.252	0.291 second
	(d)	Standard image	11.777 seconds
	(e)	0.570	0.606 second
	(f)	2.053	0.43 second
R2	(a)	Standard image	5.438 seconds
	(b)	0.177	0.495 second
	(c)	1.198	0.331 second
	(d)	Standard image	11.847 seconds
	(e)	1.570	0.712 second
	(f)	11.524	0.461 second
R3	(a)	Standard image	3.685 seconds
	(b)	0.040	0.490 second
	(c)	0.943	0.311 second
	(d)	Standard image	11.707 seconds

	(e)	0.189	0.610 second
	(f)	7.732	0.401 second

5. Conclusion

We have developed an efficient algorithm by which a sphere object modeled with Phong reflectance model can be rendered in real-time using spectral rendering. This algorithm consists of adaptive resolution scheme and parabolic interpolation techniques. Adaptive resolution scheme diminishes the requirement for tabulating entries with no degradation of accuracy. Parabolic interpolation raises the accuracy without increasing the demand for memories. These techniques can reduce the memory requirement demanded by tabulating entries and reduce time required by rendering in an acceptable range of error, which can be adjusted by alternating the number of tabulating entry.

The functionality of the algorithm proposed in this paper is to edit spectral curve for Phong reflectance model in a WYSISWYG(What You See IS What You Get) manner. This work exhibits the first effort on editing of spectral curve of reflectance. For the integrity of a spectral curve of reflectance editing tool, the following enhancements should be achieved in the near future.

- Phong reflect model is not physically plausible, the modified Phong model[10, 11], which is physically plausible, is to substitute it.
- Use of a non point light source instead of a point light source to render more naturally.

References

[1] Roy Hall and Donald P. Greenberg, "A Testbed for Realistic Image Synthesis," *IEEE Computer Graphics and Applications*, Vol. 3, No. 8, pp. 10-20, November 1983.

[2] Donald P. Greenberg, Kenneth E. Torrance, Peter Shirley, James Arvo, James A. Ferwerda, Sumata Pattanaik, Eric P. F. Lafortune, Bruce Walter, Sing-Choong and Ben Trunbore, "A Framework for Realistic Image Synthesis," In *Proc. of ACM SIGGRAPH 97*, pp. 477-494, 1997.

[3] Yinlong Sun, "A Spectrum-Based Framework for Realistic Image Synthesis," Ph.D thesis, School of Computing Science, Simon Fraser University, 2000.

[4] Steven Collins, "Wavefront Tracking for Global Illumination Solutions," Ph.D thesis, Trinity College Dublin, 1997.

[5] B. T. Phong, "Illumination for Computer Generated Pictures," *Communications of the ACM*, Vol. 18, No. 6, pp. 311-317, June 1975.

[6] Christophe Schlick, "A Fast Alternatives to Phong's Specular Model," *Graphics Gems IV*, pp. 385-387, Academic Press, Boston, 1994.

[7] Foley JD, van Dam A, Feiner SK and Hughes JF, Computer Graphics:Principles and Practice, 2nd ed. Addison-Wesley, Reading Massachusetts,1990.

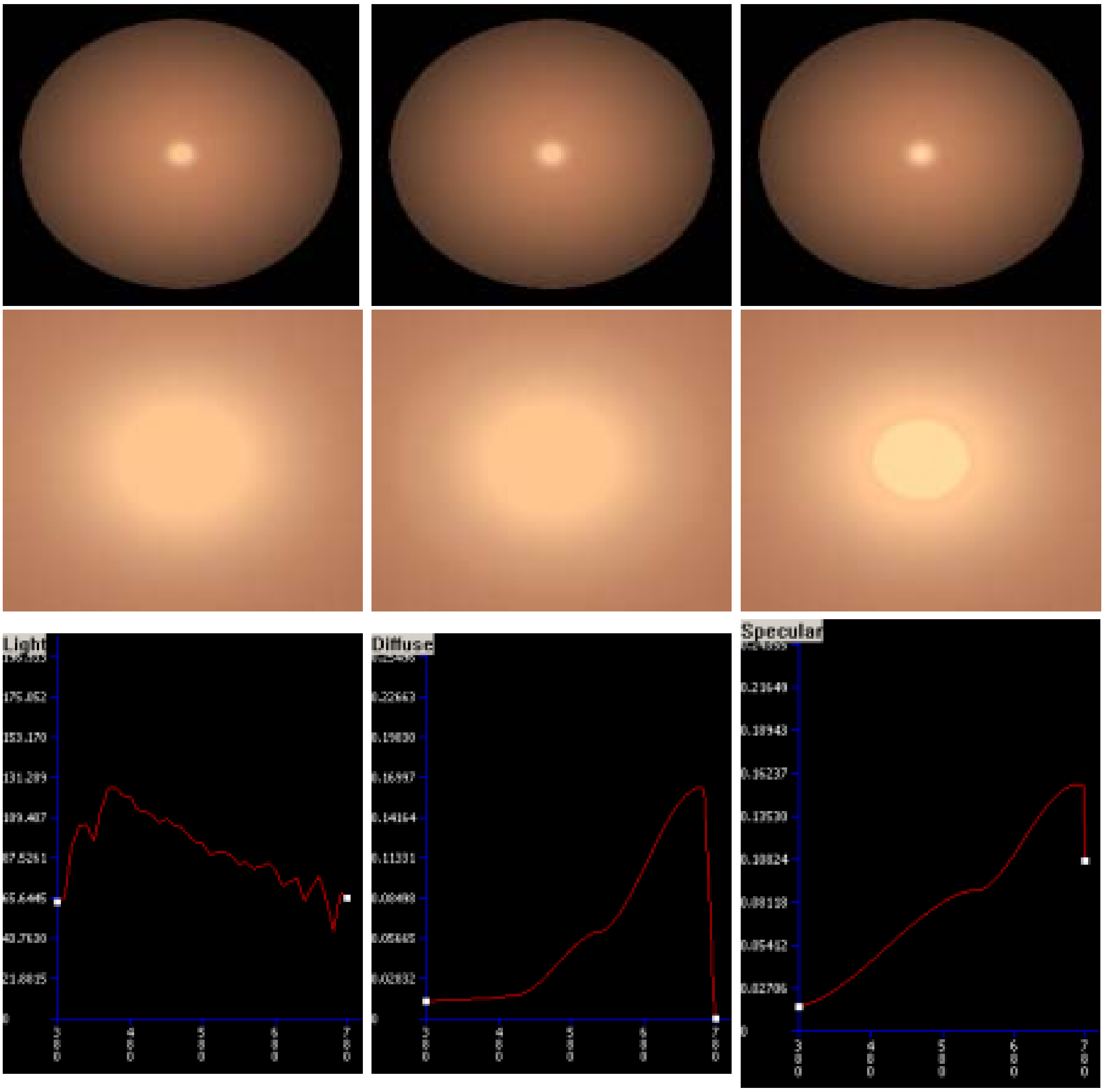
[8] M.D. Fairchild, Color Appearance Models, Addison-Wesley, Reading, Massachusetts., 1998.

[9] Roy Hall, Illumination and Color in Computer Generated Imagery, Springer-Verlag, New

York, 1989.

[10] R.Lewis, “Making Shaders More Physically Plausible,” *Computer Graphics Forum*, Vol. 13, No. 2, pp. 109-120, june 1994.

[11] E. Lafortune and Y. D. Willems, “Using the Modified Phong Reflectance Model for Physically Based Rendering,” Technical Report RP-CW-197, Department of Computer Science, K.U. Leuven, 1994.



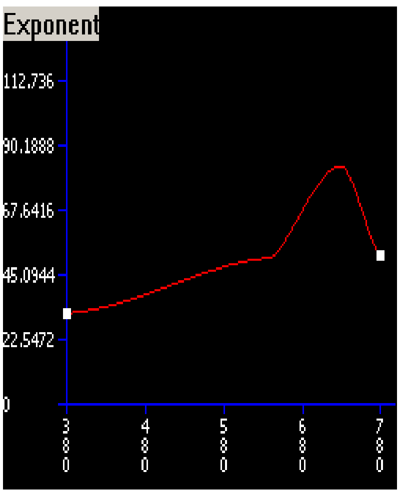
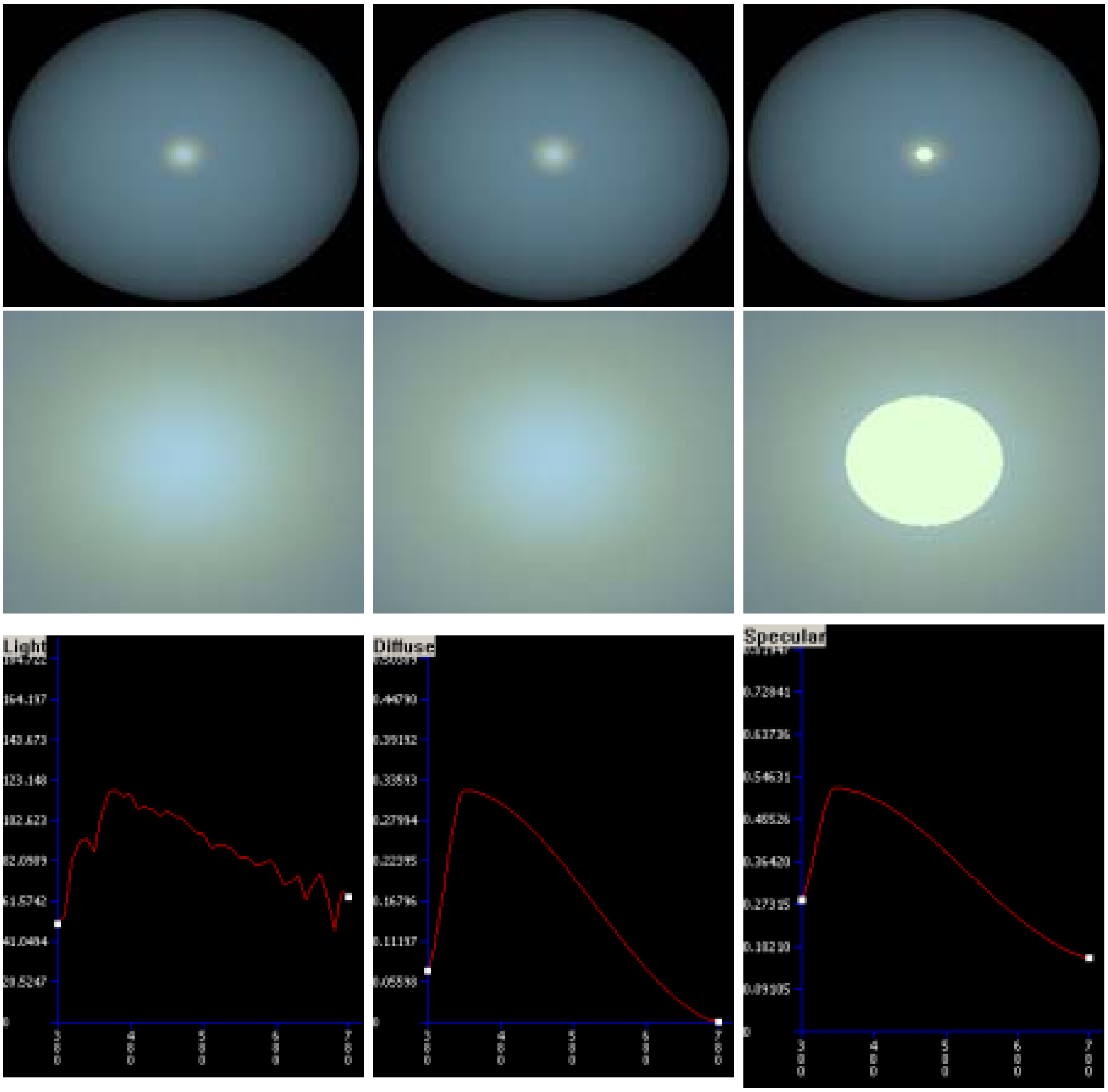


Figure 5 Combination of reflectance R1. (a). Zoom out image without tabulating. (b). Zoom out image rendered by algorithm proposed with 200 tabulating entries. (c). Zoom out image rendered by uniform interval tabulation with 200 tabulating entries. (d). Zoom in image of (a). (e). Zoom in image of (b). (f). Zoom in image of (c). (g). Spectral curve of light (h). Spectral curve of diffuse coefficients. (i). Spectral curve of specular coefficients. (j). Spectral curve of exponents



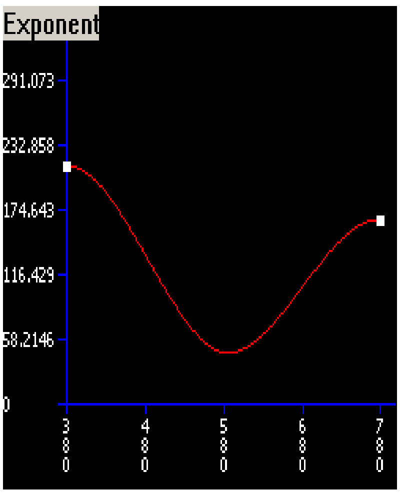
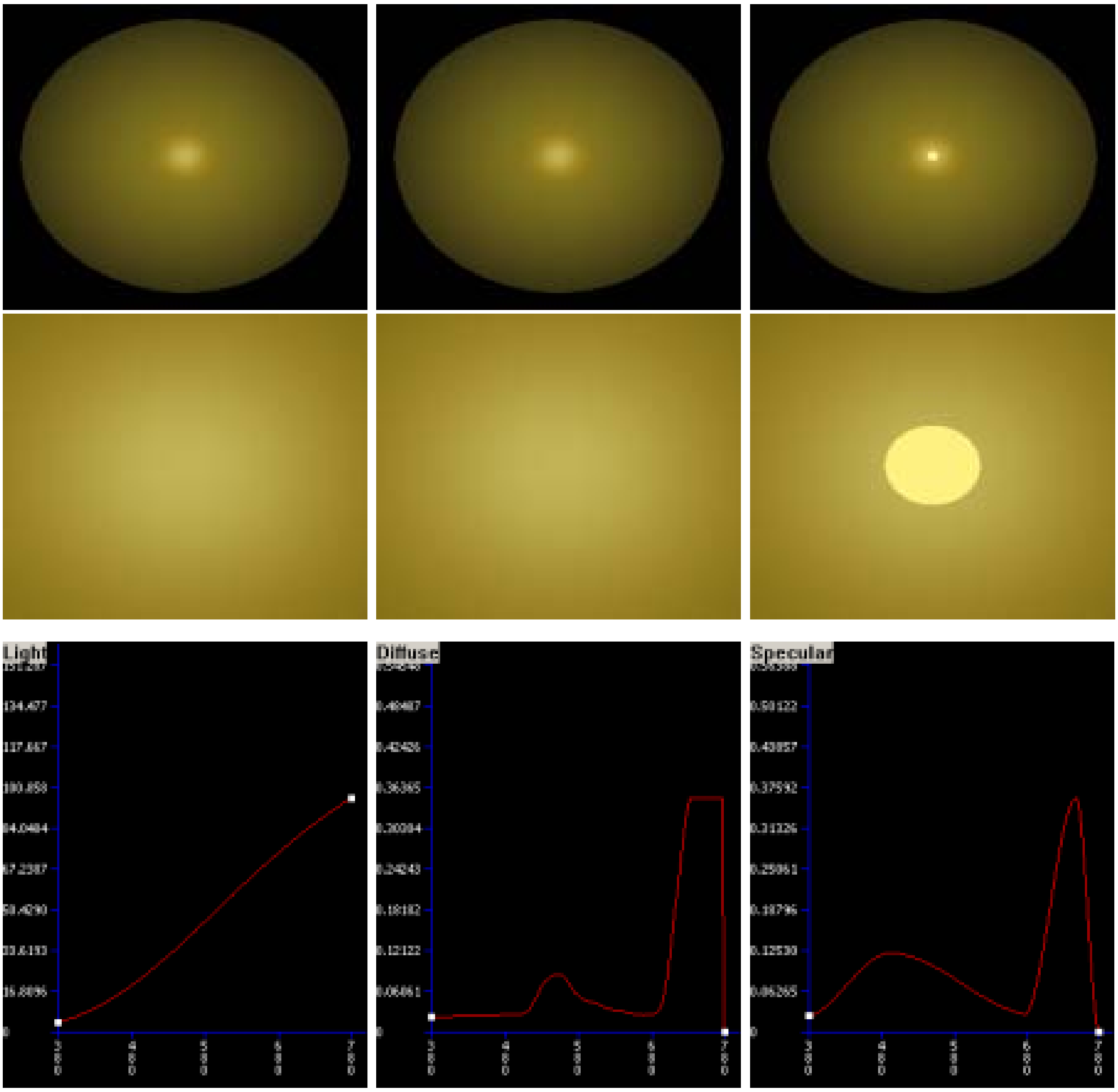


Figure 6 Combination of reflectance R2. (a). Zoom out image without tabulating. (b). Zoom out image rendered by algorithm proposed with 200 tabulating entries. (c). Zoom out image rendered by uniform interval tabulation with 200 tabulating entries. (d). Zoom in image of (a). (e). Zoom in image of (b). (f). Zoom in image of (c). (g). Spectral curve of light (h). Spectral curve of diffuse coefficients. (i). Spectral curve of specular coefficients. (j). Spectral curve of exponents



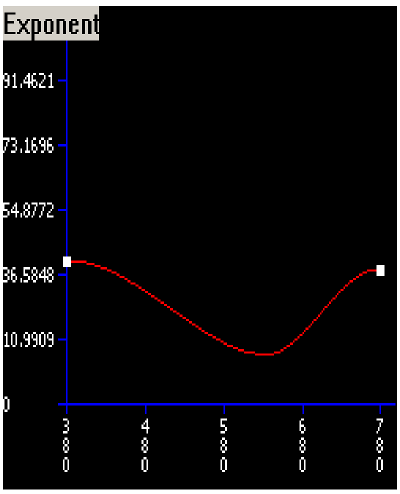


Figure 7 Combination of reflectance R3. (a). Zoom out image without tabulating. (b). Zoom out image rendered by algorithm proposed with 200 tabulating entries. (c). Zoom out image rendered by uniform interval tabulation with 200 tabulating entries. (d). Zoom in image of (a). (e). Zoom in image of (b). (f). Zoom in image of (c). (g). Spectral curve of light (h). Spectral curve of diffuse coefficients. (i). Spectral curve of specular coefficients. (j). Spectral curve of exponents