

An Inverse Rendering Approach for Heterogeneous Translucent Materials

Jingjie Yang & Shuangjiu Xiao
Shanghai Jiao Tong University

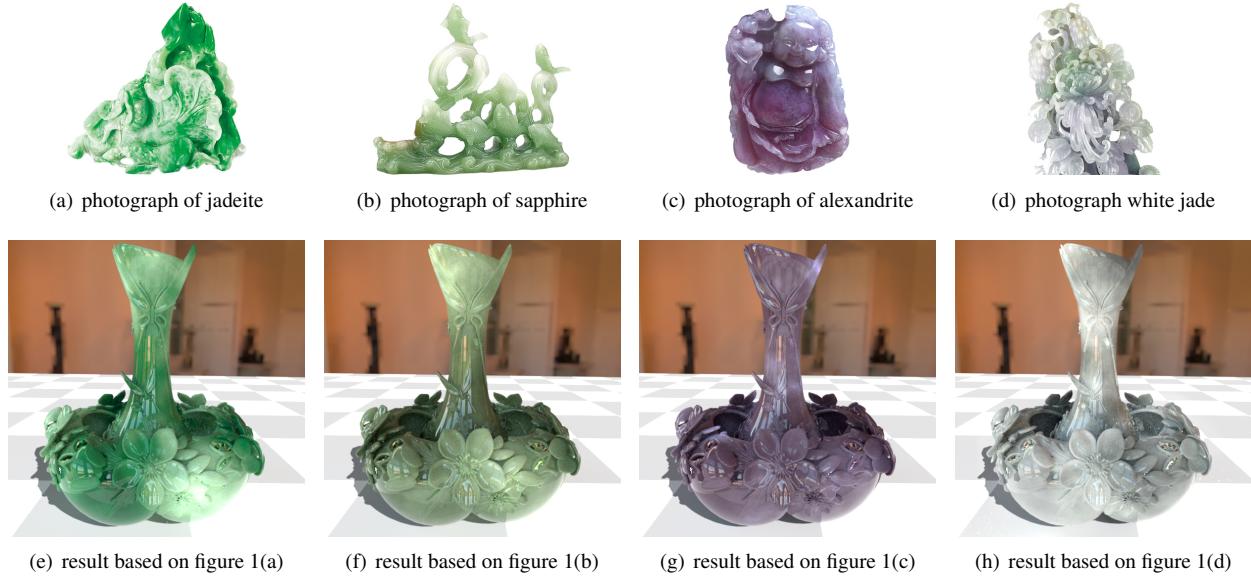


Figure 1: Inverse rendering results of a vase model. The first row shows four input photographs of different heterogeneous translucent materials without irrelevant backgrounds. The second row shows the rendering results of our approach, which generates volume material data and estimate the heterogeneous optical parameters of our volume light transport model from a single input photograph.

Abstract

Since heterogeneous translucent materials, such as natural jades and marble, are complex hybrids of different materials, it is difficult to set precise optical parameters for subsurface scattering model as the material really has. In this paper, an inverse rendering approach is presented for heterogeneous translucent materials from a single input photograph. Given one photograph with an object of a certain heterogeneous translucent material, our approach can generate material distribution and estimate heterogeneous optical parameters to render images that look similar to the input photograph. We initialize material distribution using 3D Simplex Noise combined with Fractal Brownian Motion, and set color pattern of the noise using histogram matching method. The volume data with heterogeneous optical parameters is initialized based on the value of color pattern matched noise, and it is rendered in a certain lighting condition using Monte Carlo ray marching method. An iteration process is designed to approximate optical parameters to minimize the difference between rendering result and input photograph. Then the volume data with optimal heterogeneous optical parameters is obtained, which can be used for rendering any geometry model in different lighting conditions. Experimental results show that heterogeneous translucent objects can be rendered precisely similar to the material in the photograph with our approach.

Keywords: heterogeneous translucent material, inverse rendering, histogram matching, subsurface scattering

Concepts: •Computing methodologies → Computer graphics;

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not

1 Introduction

The importance of rendering volumetric scattering materials has kept increasing in recent years. A lot of efforts have been devoted to modeling and simulating translucent materials in order to give an efficient and precise result. The appearances of many real-world heterogeneous translucent materials including marble and jade arise from complex interactions between light and the material volume with spatially variant optical properties. Light penetrates into heterogeneous translucent materials and scatter multiple times before re-emerging towards the observer. Due to the complexity of heterogeneous translucent materials, it is challenging to measure and simulate this phenomenon efficiently and accurately in computer graphics.

Volume optical parameters are important in rendering heterogeneous translucent materials with complex composition of matters. In order to render a high-quality image, a series of parameters are required as input.

For homogeneous materials, the scattering parameters can be faithfully measured from real-world materials [Mukaigawa et al. 2009] [Gkioulekas et al. 2013]. In these studies, a laser and a s-

made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org. © 2016 ACM.

VRCAI '16, December 03-04, 2016, Zhuhai, China

ISBN: 978-1-4503-4692-4/16/12

DOI: <http://dx.doi.org/10.1145/3013971.3013973>

scanner are used to collect brightness data of materials with fixed shapes, and physically accurate optical parameters are calculate using the brightness data.

However, for heterogeneous translucent materials, it is difficult to get completely precise parameters from real world using scanning method. Volume distributions of heterogeneous translucent materials are complex. At a particular position in the volume, the scattering phenomenon of light can be influenced by different scattering and absorption events that cannot be separated. [Wang et al. 2008] cut heterogeneous translucent materials into sheets and use scanning method to obtain material distribution and optical parameters, but the method is costly for rare materials. In order to render some specified materials, the optical parameters are still need to be adjusted manually, and the computational cost of repetitive trial-and-error process is quite expensive.

[Zhao et al. 2014] makes some experiments and shows that different optical parameters can lead to identical rendering results, which gives us some inspiration. Instead of using complex equipment to get parameters, we present a method that can generate volume material distribution and estimate optical parameters based on one photograph. The optical parameters are not physically accurate, but the material in the rendering result is visually similar to the material in the source photograph.

In our approach, we initialize 3D noise data using Simplex Noise combined with Fractal Brownian Motion, and use some image processing techniques to map values in 3D noise to optical parameters *scattering coefficient* σ_s , *absorption coefficient* σ_a , and *phase function* f_p . After that, we generate our 3D volume data with these optical parameters. In our approach, we set the volume data 256^3 resolution with σ_s , σ_a and f_p stored in each point.

The benefits of our approach are:

- (1) The input data of our approach is simple and inexpensive. The traditional scanning method costs a lot because the method needs to cut real-world materials into a particular shapes and use some special equipments. Our approach does not use expensive and complex equipments. The input data of our approach is just one photograph of a translucent material object, which is very easy to get.
- (2) When our approach generates volume data with optical parameters that make the rendering result similar to the photograph, we can render the same volume data with arbitrary geometry models, viewpoints and lighting conditions.

Our contributions include:

- A framework that take one photograph of a certain heterogeneous translucent material as input, generate material distribution and estimate optical parameters to render images that look similar to the material in the photograph.
- An iteration process that can approximate optical parameters including scattering coefficients, absorption coefficients and phase function parameters based on the photograph.
- A 3D volume model with heterogeneous volume optical parameters to express a certain type of heterogeneous translucent material than can be rendered using Monte Carlo ray marching method.

In addition to our visual appearance of our rendering results, our approach is also validated by comparing color histograms of rendering result and input photograph. **Figure 1** shows an example of our method.

2 Related Work

Our approach covers a wide range of studies including image processing, modeling and rendering. In this section we review prior works in several parts as below.

Participating Medium. Participating Medium describes a kind of medium type. When radiation travels through a participating medium, it undergoes three kinds of phenomena: absorption, scattering, and emission. [Siegel et al. 1992] use mass coefficients to build volume model for participating media. If the material is homogeneous, the medium has a constant density [Blinn 1982] [Max 1986] [Klassen 1987] [Nishita 1987]. If the media is heterogeneous, mediums are expressed using heuristic functions [Gardner 1985] [Perlin 1985] [Ebert and Parent 1990], texturing functions, fractal algorithms [Sakas 1990] [Sakas 1993], particle systems [Yaeger et al. 2011] and blobs [Stam 1996].

Radiative Transfer Equation Radiative Transfer Equation(RTE) describes the transfer of energy in participating media. In the early years, radiative transfer was used in many areas including astrophysics [Hulst and H. 1958] [Mishchenko et al. 2006], wave propagation [Ishimaru and Ishimaru 1978], etc. [Blinn 1982] introduced radiative transfer to the field of computer graphics. [Jakob et al. 2010] proposes a new volume scattering model to better handle scattering media with oriented structures.

There are a lot of methods that can solve radiative transfer equation. The methods can be mainly divided into two categories: Monte Carlo Method and Diffusion Equation.

Monte Carlo Method. Monte Carlo methods solve the full RTE directly that trace random rays within the environment. One approach of the method is getting sampling points by using a constant step distance [Philippe et al. 1993] [Blasi et al. 1998]. Another approach is getting sampling a cumulative density function at points with random distances [Pattanaik and Mudur 1993] [?]. Some faster methods are used in recent years. [Hachisuka et al. 2012] introduced volumetric photon mapping, and [Dachsbacher et al. 2014] developed many-lights methods.

Diffusion Equation. [Ishimaru and Ishimaru 1978] presents diffusion equation by applying first-order approximation do direct radiance. [Jensen et al. 2002] presents dipole model for subsurface scattering. [D'Eon and Irving 2011] [Frisvad et al. 2014] optimize the dipole model to render more realistic results. Besides dipole model, finite element method [Stam 1995] is also widely used. [Wang et al. 2008] [Li et al. 2013] use finite element method to render heterogeneous materials.

Inverse Rendering. Forward rendering method render the scene using given parameters, while inverse rendering method recover material properties from the real world. [Schoeneman et al. 1993] [Pellacini et al. 2007] [Kawai et al. 2010] proposed methods for solving inverse lighting problem. Recent years, many other methods have been proposed to render different kind of materials. [Wang et al. 2008] [Adolfo et al. 2011] [Dobashi et al. 2012] [Gkioulekas et al. 2013] [Papas et al. 2013] used inverse rendering methods for rendering translucent scattering materials.

Noise Function. Since most materials are mixtures of different matters, we consider different material volumes as different distribution of heterogeneous noises. Perlin noise was introduced by [Perlin 1985]. After that, a more efficient algorithm called Simplex Noise [Perlin 2001] was introduced in 2001. Noises are applied for generating terrain, textures, sea surface, etc. [Acosta et al. 2014] presents an image-processing model to simulate rusted textures using Perlin Noise.

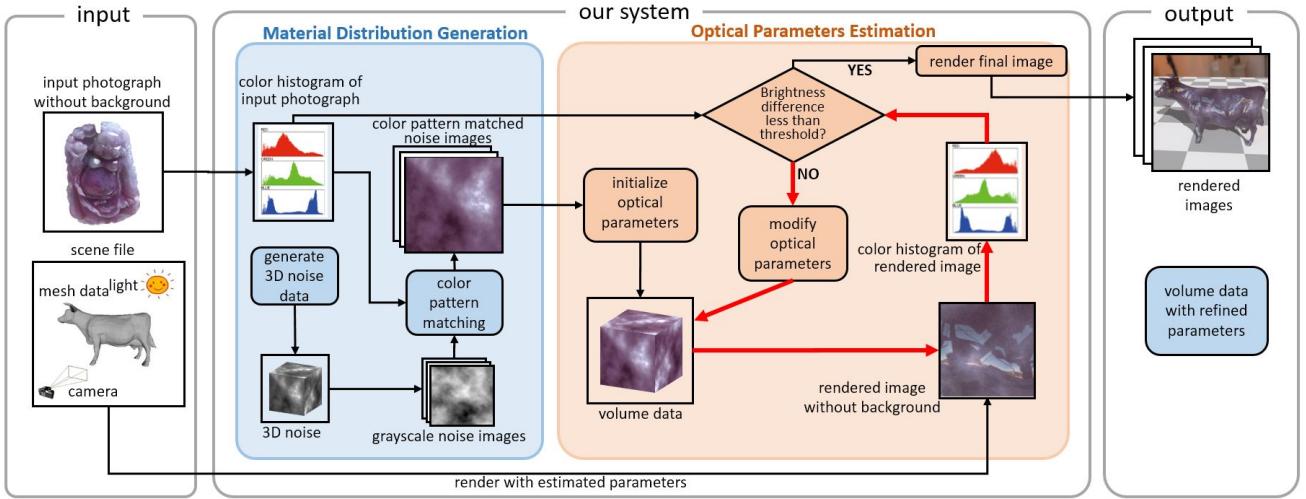


Figure 2: The framework of our approach. The input data of our approach contains a photograph without irrelevant background, and a scene file containing geometry model, light and camera. There are mainly two stages: material distribution generation stage and optical parameters estimation stage. In material distribution generation stage, we initialize 3d noise data using Simplex Noise combined with Fractal Brownian Motion, and we generate color pattern matched noise images using histogram matching method based on the histogram of input photograph. In optical parameters estimation stage, the heterogeneous optical parameters are initialized based on the RGB value of color pattern matched noise images. The red arrows indicate our iteration process of approximating parameters to make rendering result more similar to the photograph. In each iteration, we compare histograms of input photograph and rendering result, and update optical parameters of volume data based on the histogram differences. After iteration, our approach outputs the rendering result and volume data with optimal optical parameters.

3 Framework

The framework of our approach is illustrated by Figure 2. There are mainly two processing stages. The first stage of our approach is generating volume material distribution, and the second stage is estimating optical parameters.

The input data of our approach contains two parts. The first part is one photograph of heterogeneous translucent material object. As a preprocessing step, the irrelevant background is removed from the photograph. The second part of the input is 3D scene file including geometry model, light and view options, etc. The goal of our approach is to generate volume data with heterogeneous optical parameters for rendering photorealistic images.

In our approach, color histograms for each RGB channel of the input photograph are firstly generated. The color histograms are not only used for initializing volume data but also for optimizing optical parameters and validating the result images. In order to generate heterogeneous volume data, 3D Simplex Noise combined with FBM (Fractal Brownian Motion) is used to represent the volume material distribution of the object. The 3D noise data is separated into grayscale noise images, and we use histogram matching method to generate color pattern matched noise images in order to set color gradients of the noise.

After color pattern matched noise images are generated according to the input photograph, the initial optical parameters in each point are estimated in the volume data. The parameters include scattering coefficient σ_s , absorption coefficient σ_a , and phase function f_p . In our approach, an iteration process is designed for optimizing optical parameters in order to make rendering result look more similar to the input photograph.

4 Material Distribution Generation

In order to render images that look similar to the input photograph, an effective volume data generation method is necessary. In this section, we present our method for generating material distribution based on the photograph.

4.1 Initialization of Noise Images

In this part, we initialize three dimensional noise using Simplex Noise algorithm, in combination with the FBM for generation of underlying octaves. Simplex Noise [Perlin 2001] [Gustavson 2005] construct gradients using simplest and most compact shape that can be repeated to fill the entire space. The simplex shape looks like a slightly irregular tetrahedron. At one point in 3D space, the noise value is contributed from four simplex corners.

FBM is the sum of multiple Simplex Noise functions where each Simplex Noise function contains octaves with increasing frequencies and decreasing amplitudes. At iteration i , Simplex Noise with frequency Fr_i and amplitude Amp_i is added to the result of previous iterations. We set the volume shape to a cube with side length s , and other parameters we estimate are showed in Table 1. Fr_0 represents the frequency in the first iteration.

In our approach, we set $Fr_0 = 1/s$ and $Fr_{i+1} = Fr_i * 2$. Amplitude determines the detail of the noise. Amp_0 represents the amplitude in the first iteration. Value k is set to control amplitudes in each iteration. We estimate the value $Amp_0 = k (0 < k < 1)$ and $Amp_{i+1} = Amp_i * k$. In the process of iteration, the frequency increases and the amplitude decreases.

Given a point $P(x, y, z)$, the process of FBM can be described as follows. The noise value of point P is the sum of simplex noise values in the n iterations. All of the noise values are saved in a

parameter	meaning
s	The side length of volume.
n	Iteration num, we set $n = \log_2 a$
Fr_i	The frequency of FBM at i th iteration.
Amp_i	The amplitude of FBM at i th iteration.
k	The control number of the amplitude.

Table 1: Parameters to generate 3d noise

three dimensional array.

$$Noise(x, y, z) = \sum_0^{n-1} SimplexNoise(x*Fr_i, y*Fr_i, z*Fr_i)*Amp_i$$

We color 3D noise data based on the pixels of input photograph. We split the $s*s*s$ 3d volume into s images, each image has width s and height s . We use a histogram matching method to color the separated images, and use them to construct 3d volume data with estimated optical parameters.

In order to apply histogram matching method to set color of 3D noise data more conveniently, we cut the 3D noise data into images. Figure 3 shows the noise images with different amplitudes.

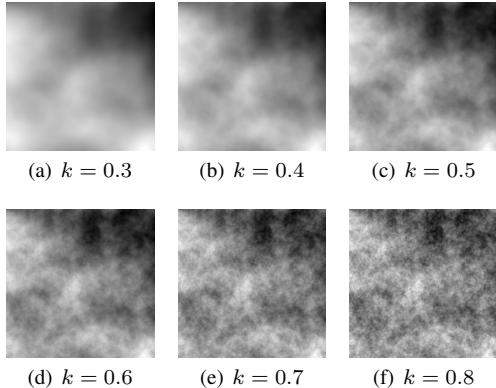


Figure 3: Noise images with different amplitudes. The images have same size 256*256, same iteration number of 8, same frequency and same random number seed. Their amplitude control numbers are different.

4.2 Generation of Color Pattern Matched Noise Images

The initialized grayscale noise images should be modified to match the color pattern of the input photograph. For visual sense, color pattern indicates the heterogeneous material distribution of the object. We use histogram matching method to achieve it. Getting the input photograph as reference image and a noise image as target image, histograms of the two images are computed respectively in RGB channels. The alpha value of pixels in reference image is considered, so that the background of reference image will not be calculated.

Then we calculate the cumulative distribution functions of the two histograms: $F_1()$ for the reference image and $F_2()$ for the target image. Then for each gray level $G_1 \in [0, 255]$, we find the gray level G_2 , for which $F_1(G_1) = F_2(G_2)$ and here is the result of histogram matching function $M(G_1) = G_2$.

Finally, we apply the function $M()$ on each pixel of the reference image.

After matching histograms in three channels, the color pattern matched 3D noise images are obtained. Figure 4 shows an example of coloring the noise image. Figure 4(a) shows the input photograph with histograms in RGB channels. Figure 4(b) indicates one of the grayscale noise images with histograms in RGB channels. The histograms in the three channels are same in 4(b). Figure 4(c) shows the color pattern matched noise image with RGB channels.

Before histogram matching, there is great difference between the source photograph and noise images. By comparing Figure 4(a) and 4(b) we can see that the color patterns and histograms look very different. After histogram matching, Figure 4(a) and 4(c) show that the color patterns and histograms in RGB channels look similar to each other.

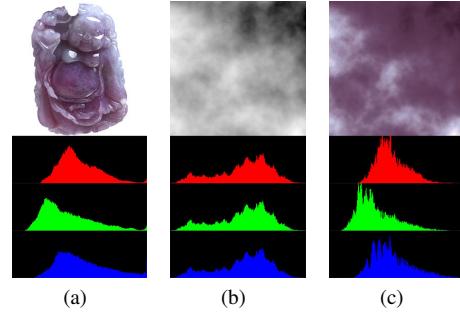


Figure 4: Comparation of colors and histograms. (a)Input photograph and histograms. (b)Grayscale noise image and histograms. The image is selected from Figure 3(c). (c)Color pattern matched noise image and histograms.

Our approach repeats the histogram matching method for all of the noise images. For each $s * s * s$ 3D noise data, s noise images with size $s * s$ are processed. The noise images all have similar colors and histograms to the source photograph. These color pattern matched noise images will be used to construct 3D volume data with estimated optical parameters. Figure 4 shows more examples of histogram matching method.

Figure 5 shows some other examples of color pattern matching of noise images. We download a photograph of heterogeneous translucent material from the Internet. Figure 5(a) contains three kind of jade materials. The shape of the stones are very simple and do not have bumps on the surfaces. We extract the image for each kind of jade and generate color pattern matched noise images respectively. The results are showed in Figure 5(b), 5(c) and 5(d).

More examples comparing input images and color pattern matched noise images are showed in Figure 6. Many photographs of carved jade are downloaded from the Internet. We remove the irrelevant backgrounds and put the remaining pixels into our system. The shapes in the photographs are irregular, and there are more details and bumps than in the photographs in Figure 5.

5 Optical Parameters Estimation

In this section, we present an iterative method of generating participating media volume data by estimating optical parameters from the color pattern matched noise images. Rendering result is used to evaluate the confidence of the optical parameters.

If the material is isotropic, the form of radiative transfer equation

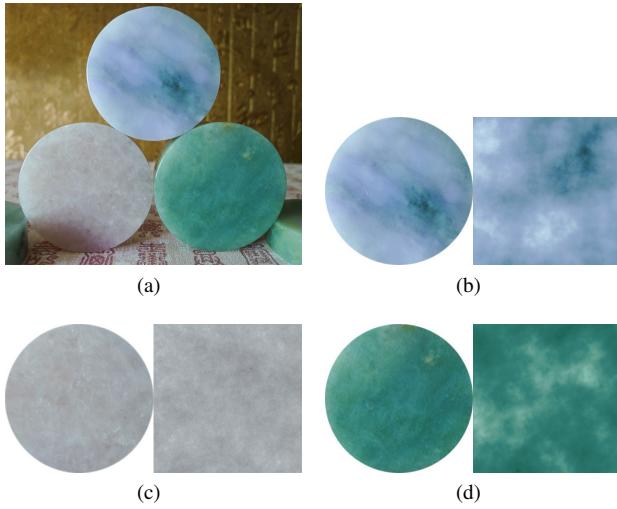


Figure 5: (a) is the input photograph of jade. (b)(c)(d) show the comparation between input photographs and noise images. The left round images are different material parts of (a), and the right square images are the color pattern matched noise images generated by our method.

is:

$$(\omega \cdot \nabla) L(x, \omega) = -\sigma_t L(x, \omega) + \sigma_s \int_{S^2} f_p(\omega' \cdot \omega) L(x, \omega') d\omega' + Q(x, \omega)$$

[Jakob et al. 2010] presents anisotropic radiative transfer framework. The anisotropic form of radiative transfer equation is:

$$(\omega \cdot \nabla) L(x, \omega) = -\sigma_t(x, \omega) L(x, \omega) + \sigma_s(x, \omega) \int_{S^2} f_p(\omega' \rightarrow \omega) L(x, \omega') d\omega' + Q(x, \omega)$$

In these equations above, $x \in R^3$ is a point in the volume. ω is direction of incident light and ω' is direction of emergent light. $L(x, \omega)$ is intensity of the incident light at position x and orientation ω . $Q(x, \omega)$ is self-luminous parameter of the material. Scattering coefficient σ_s determines the amount of light that is scattered. Absorption coefficient σ_a determines the light that is absorbed. Extinction coefficient σ_t is the sum of σ_s and σ_a , where $\sigma_t = \sigma_s + \sigma_a$. The phase function f_p determines the proportion of light that is scattered from direction ω to direction ω' . The phase function is normalized, that $\int_{4\pi} f_p(\omega' \cdot \omega) d\omega' = 1$. In isotropic materials, the amount of scattered light in all directions are equal. In anisotropic materials, the proportion of scattered light in different directions varies when the direction of incident light changes.

We do not consider self-illumination in our approach. The optical parameters influence the color of the rendering result. We will present our method of estimating the relevant parameters separately. In our approach, we initialize scattering coefficient σ_s and absorption coefficient σ_t based on the value of color pattern matched noise. There is a small gap between our rendering result using initial parameters and input photograph. In order to reduce the difference, we use an iterative method to approximate these parameters by comparing histograms of the rendering result and input photograph.

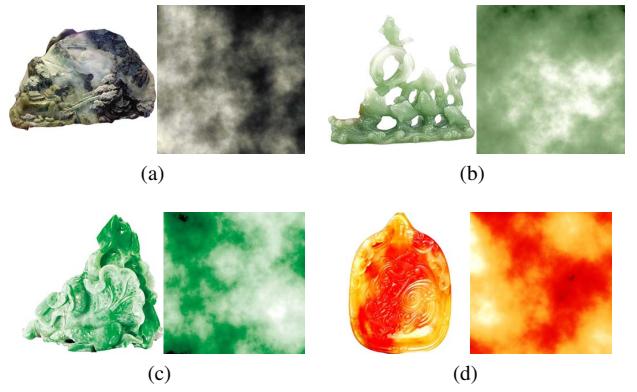


Figure 6: Other examples of comparation between photographs and color pattern matched noise images. The left images are photographs of different types of jades. The right images are color patterned matched images.

5.1 Scattering Coefficient and Absorption Coefficient Initialization

Scattering coefficient σ_s and absorption coefficient σ_a are estimated according to the RGB value of color pattern matched noise images. In our approach, we present a method of mapping the color of noise images to these two optical parameters.

The scattering coefficients in RGB channels determine the intensities that the light is scattered in these three channels. There is a positive correlation between scattering coefficients and rendered color. The proportion of RGB values in scattering coefficient is close to the proportion of RGB values in the rendering result result. Absorption and scattering are opposite phenomena, so there is a negative correlation between absorption coefficient and render color. Hence, at each point in the volume, we initialize σ_s and σ_a using noise colors and optimize them in the process of iterations.

Table 2 gives some intervention variables that we use to estimate the parameters. For each point $x(i, j, k)$ in the volume, we get the RGB value from the k th noise image at position (i, j) . The RGB values are three integers having range $[0, 255]$. We divide the values by 256 to set them to set the range to $[0, 1)$. We estimate the parameters $\sigma_s(x) = v_c(x) * d_s + k_s$ and $\sigma_a(x) = (1 - v_c) * d_s * d_a$. d_s influences the scattering intensity and d_a influences the material translucency.

parameter	meaning
$v_c(x)$	color of noise data at point x
d_s	scale value to multiply with $v_c(x)$.
$k_s(r, g, b)$	parameters to adjust brightness in RGB channels
d_a	scale value for generating σ_a .

Table 2: Parameters to estimate.

We estimate the initial values $d_s = 1.0$ and $d_a = 0.1$, which means the material has a tendency that the scattering contribute more than the absorption. We initialize $k_s = (0, 0, 0)$, and adjust the k_s during iterations.

5.2 Phase Function Estimation

The input image does not have enough information for estimating the proportion of light that is scattered in different directions. We

set an isotropic phase function to $f_p(\omega' \cdot \omega) = \frac{1}{4\pi}$, which means the value is constant in all directions.

5.3 Rendering Method

In our approach, the initial volume data is used to render the target geometric model with Monte Carlo ray marching method. The initial volume data is put in to the scene file with a geometry model. Other data include a background image, a directional light and a camera. The color of the directional light is white and the intensity of light in the scene is fixed. Since most of the materials in the photograph has very smooth surface, we use specular reflection coefficient of 0.5. That means half of the light is reflected and half of the light is refracted on the surface. We set the refractive index to 1.6, which can represent many kind of jades.

5.4 Iteration for Parameter Approximation

In our approach, we use histograms to evaluate the similarity between the input photograph and the rendering result. We compare histograms in RGB channels separately and adjust σ_s and σ_t according to some variables that are designed to calculate histogram differences. These variables are showed in Table 3

parameter	meaning
$H_{input}(i, \kappa)$	The proportion of pixels with gray value i and channel κ in input photograph.
$H_{result}(i, \kappa)$	The proportion of pixels with gray value i and channel κ in rendering result.
i	The gray value.
κ	The RGB channel.
$D(\kappa)$	The variance of two histograms in channel κ .
$A_{input}(\kappa)$	The average histogram value of input photograph in channel κ .
$A_{output}(\kappa)$	The average histogram value of rendering result in channel κ .

Table 3: Histogram parameters.

The difference between two histograms in channel κ is:

$$D(\kappa) = \sum_{i=0}^{255} (|H_{input}(i, \kappa) - H_{result}(i, \kappa)|^2)$$

$$i \in [0, 255], \kappa \in \{R, G, B\}$$

The average values of histograms in channel κ are showed as below, the average values are used to compare brightness of two images.

$$A_{input}(\kappa) = \sum_{i=0}^{255} i * H_{input}(i, \kappa)$$

$$A_{output}(\kappa) = \sum_{i=0}^{255} i * H_{output}(i, \kappa)$$

Algorithm 1 compares histograms of two images in one of RGB channels. The output data is used to determine the difference between two images.

Algorithm 2 shows our approximation method of optical parameters. The average values of two histograms determine the search direction. If the rendering result is brighter, our approach reduces the scattering coefficient. If the rendering result is darker, our approach increases the scattering coefficient. Δk influences the rate

Algorithm 1 Histogram Compare Algorithm

Input: Histograms of Two Images

Output: Variance and Brightness Difference Between Histograms

```

1: function HISTDIFF( $H_{input}, H_{output}, \kappa$ )
2:    $D(\kappa) \leftarrow 0$ 
3:   for  $i \leftarrow 0$  to 256 do
4:      $D(\kappa) += [H_{input}(i, \kappa) - H_{result}(i, \kappa)]^2$ 
5:   end for
6:   return  $D(\kappa)$ 
7: end function
8: function HISTAVGDIFF( $H_{input}, H_{output}, \kappa$ )
9:    $A_{input}(\kappa) \leftarrow 0, A_{output}(\kappa) \leftarrow 0$ 
10:  for  $i \leftarrow 0$  to 256 do
11:     $A_{input}(\kappa) += i * H_{input}(i, \kappa)$ 
12:     $A_{output}(\kappa) += i * H_{output}(i, \kappa)$ 
13:  end for
14:   $A_{input}(\kappa) /= 256, A_{output}(\kappa) /= 256$ 
15:   $result \leftarrow A_{output}(\kappa) - A_{input}(\kappa)$ 
16:  return  $result$ 
17: end function

```

of convergence. We set $\Delta k = 0.01$ and $\Delta d = 1.0$, which means we increase or decrease σ_s by 0.01 until the brightness difference is less than 1.0 in each channel.

In the parameter approximation process, the background of the scene is not rendered. Once final version of parameters are acquired, our approach will render the complete scene.

Algorithm 2 Parameter Approximation Algorithm

Input: Photograph and Volume Data

Output: Adjusted Optical Parameters

```

1: function ADJUSTPARAMS
2:   Render Volume Data With Initial Parameters
3:   Compute  $H_{input}$  and  $H_{output}$ 
4:    $histAvgDiff \leftarrow HISTAVGDIFF(H_{input}, H_{output}, \kappa)$ 
5:   if  $histAvgDiff > 0$  then
6:      $sign \leftarrow 1$ 
7:   else
8:      $sign \leftarrow -1$ 
9:   end if
10:  while  $histAvgDiff * sign > \Delta d$  do
11:     $k_s(\kappa) += sign * \Delta k$ 
12:    Update  $\sigma_s$  and  $\sigma_t$  in the volume
13:    Render new volume data and compute  $H_{output}$ 
14:     $histAvgDiff \leftarrow HISTAVGDIFF(H_{input}, H_{output}, \kappa)$ 
15:  end while
16: end function

```

Figure 7 shows our process of iteration. Figure 7(a) is the input photograph. Figure 7(b) is the rendering result using initial parameters generated by our approach.

Figure 7(d) shows the variance and brightness difference of RGB channels in each iteration. The top row shows the rendering result using parameters generated in each iteration. Compared with the photograph, the initial rendering result has some difference with input photograph. During the process, there is almost no change in variances, the differences in brightness become smaller. In the last iteration, the brightness differences in three channels are close to zero.

Figure 7(c) shows the rendering result using approximated parameters. After iteration, the rendering result looks more similar to the input photograph.

We integrate the approximated optical parameters into our volume data for rendering images with different geometry models and different lighting conditions.

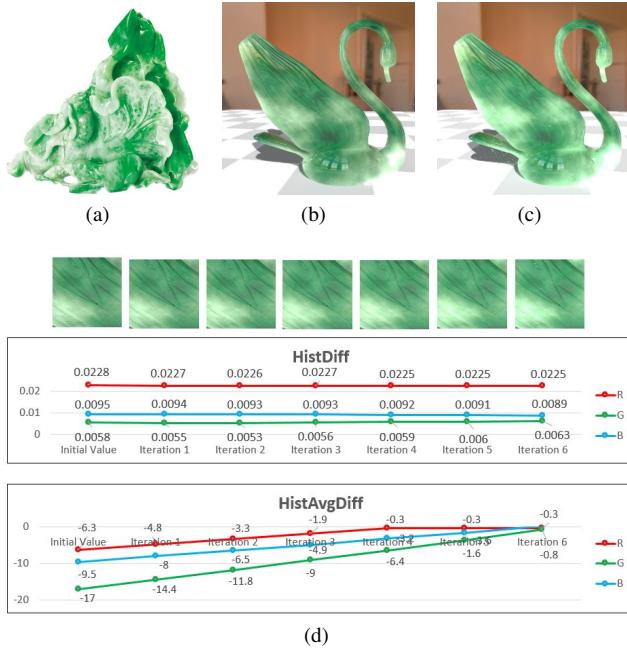


Figure 7: Results and evaluation of optical parameters approximation. (a) Input photograph. (b) Rendering result using initial parameters. (c) Rendering result using approximated parameters. (d) Variance and brightness difference of RGB channels between input photograph and rendering result in each iteration.

6 Experimental Results

We performed tests on an Intel Core i7-4770K 3.5GHz CPU with 8GB memory, and an NVIDIA GeForce GTX770 GPU.

Our approach is tested using different kinds of input photographs with different color patterns and different shapes. Some of the objects in the photographs look simple without many bumps and details. Some of the objects in the photographs look complicated and finely crafted.

After generating volume data using the photograph, we also render the volume in different view points and different lighting conditions. Mitsuba physically-based renderer [Jakob 2010] is used in our rendering process.

We introduce our main results in the following parts.

Simple Shape Input

The photographs are raw materials of jades. In the photographs, the shape of the stones are very simple and do not have bumps on the surfaces. Figure 5 shows some intermediate results of the photographs. Since the photographs do not have any highlight or self-shadow, our approach can extract the colors precisely.

Figure 8 shows some of our results. There are 4 photographs in our test. We generate 256*256*256 volume data for every photograph, and render each volume data with two geometry models. One geometry model does not have many details, and the other one has complex geometry. The number of vertices and faces of the meshes

are showed in Table 4. All of the images are rendered under the same lighting condition.

mesh	number of vertices	number of faces
vase	203134	401738
swans	16136	32192
deer	15874	31744

Table 4: Mesh data

In our approach, the execution time of generating initial parameters is mainly affected by the size of the volume. For a 256*256*256 volume, it takes about 1 minute to estimate initial parameters. In the parameters approximation process, the execution time is affected by volume size, mesh data and resolution of image. And the execution time of rendering mainly depends on the resolution of image, the resolution of volume data and the complexity of mesh data. It takes about 2 hours to render the vase model in Figure 8 with image resolution 600*600 and 2048 samples for each pixel.

Complicated Shape Input

For complicated input, some of the results are showed in Figure 1. We download some photographs of delegate jade carving from the Internet and remove the backgrounds carefully. We generate corresponding volume data and render the data using a high polygon mesh. The four rendering results showed in Figure 1 have same volume size and they are rendered under same lighting condition. The mesh data are also showed in Table 4.

More results with complicated inputs are showed in Figure 9. Our approach render one image for each photograph and compare rendering results with input photographs. From the results we can see that the materials rendered by our approach look similar to the materials in the photographs.

Rendering Results in Different Lighting Conditions

After estimating optical parameters, the volume data can be rendered in different lighting conditions. Figure 10 shows a group of examples. The volume is generated in a specific lighting condition, which is showed in Figure 8(b). Then we can change the direction and brightness of the light. There is a directional light and global illumination in the scene. Figure 8(b) shows an initial state of light. Figure 10(a) changes the perspective of camera to let the light shine from behind. Figure 10(b) reduces the brightness of global illumination. Figure 10(c) keeps the perspective unchanged and change the direction of lights. Figure 10(d) and 10(e) increase the brightness of lighting and change the perspective of camera.

7 Discussion and Limitation

Our approach takes one photograph of heterogeneous translucent material as input. The background of the photograph should be removed carefully in order to avoid interrupting the process of estimating optical parameters. If the background is not removed completely, the parameter estimation process will be affected and the final rendering result will be mixed with background color.

Our approach can estimate parameters of a wide range of materials, which is showed in Figure 9. However, in order to get good results, there are some limitations of the input photograph. The brightness of light in the photograph should be in a reasonable range. If the there are excessive specular areas in the input photograph, our approach may overestimate scattering parameters, which is showed in Figure 11(a).

The surface of material in the photograph can be uneven. However, if there are too many holes and too much self-shadow, the



Figure 8: Rendering result using photographs of simple shapes. The left round images are input photographs of different jades. From each photograph, two images are rendered using our method. The middle images are our rendering results using mesh "deer" and the right images are our rendering results using mesh "vase".

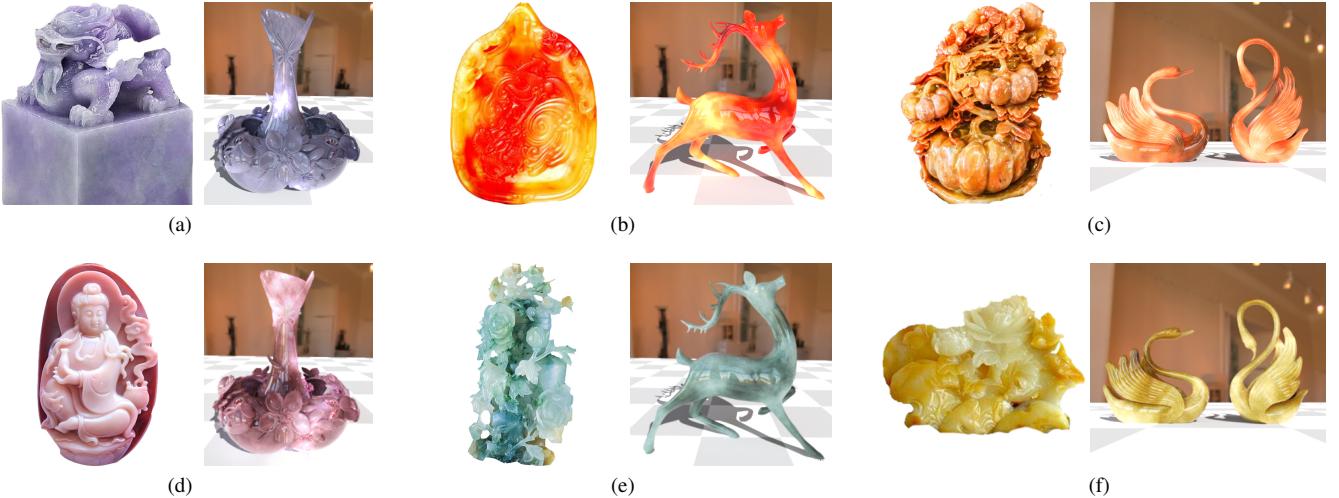


Figure 9: Rendering results based on photographs of complicated shapes. There are 6 photographs which are showed in the left of each subfigure. From each photograph, we render one image using our method. The rendering results are showed in the right of each subfigure.

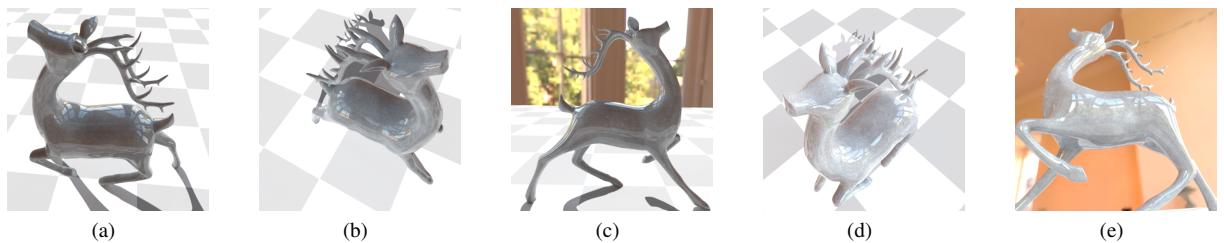


Figure 10: Rendering results in different lighting conditions. (a)Change perspective. (b)Reduce brightness of global illumination. (c)Change directions of lights. (d)Increase brightness of lights. (e)Increase brightness of lights.

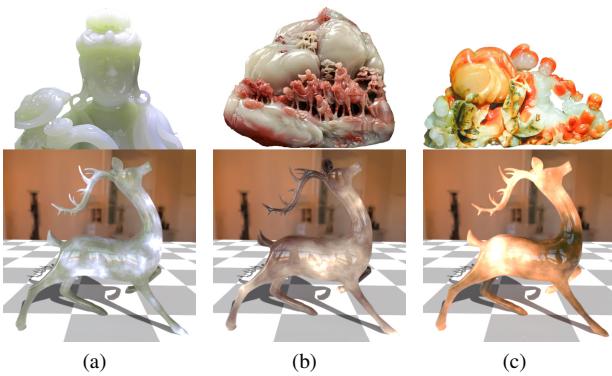


Figure 11: Some failure examples. (a)Excessive specular areas. (b)Too much self shadow. (c)Too many colors.

parameter estimation process will be affected, which is showed in Figure 11(b).

If the material in the photograph is very colorful, our approach may ignore some of the colors. Figure 11(c) shows an example, the photograph contains green but our rendering result ignores that color.

8 Conclusion

We propose an inverse volume rendering approach of estimating optical parameters of heterogeneous materials based on one photograph. We initialize material distribution using 3d Simplex Noise combined with Fractal Brownian Motion, and use histogram matching method to generate color pattern matched 3D noise. We initialize optical parameters based on the RGB value of color pattern matched noise, and use an iteration process to approximate these optical parameters in order to minimize the histogram difference between rendering result and photograph. After optimizing the parameters in a certain lighting condition, the volume data can be rendered with different geometry models and in different lighting conditions. In order to prove the usefulness of our approach, we download different kinds of photographs, and render the volume data generated based on these photographs.

References

- ACOSTA, M. R. G., DIAZ, J. C. V., AND CASTRO, N. S. 2014. An innovative image-processing model for rust detection using perlin noise to simulate oxide textures. *Corrosion Science* 88, 6, 141–151.
- ADOLFO, M., ECHEVARRIA, J. I., SERON, F. J., JORGE, L., MASHHUDA, G., AND DIEGO, G. 2011. Bssrdf estimation from single images. In *Computer Graphics Forum*, 455–464(10).
- ARBREE, A., WALTER, B., AND BALA, K. 2010. Heterogeneous subsurface scattering using the finite element method. *IEEE Transactions on Visualization and Computer Graphics* 17, 7, 956–969.
- BLASI, P., SAC, B. L., AND SCHLICK, C. 1998. *An Importance Driven Monte-Carlo Solution to the Global Illumination Problem*. Springer Berlin Heidelberg.
- BLINN, J. F. 1982. Light reflection functions for simulation of clouds and dusty surfaces. *Acm Siggraph Computer Graphics* 16, 16, 21–29.
- DACHSBACHER, C., KRIVANEK, J., HASAN, M., ARBREE, A., WALTER, B., AND NOVAK, J. 2014. Scalable realistic rendering with many-light methods. *Computer Graphics Forum* 33, 1, 88–104.
- D'EON, E., AND IRVING, G. 2011. A quantized-diffusion model for rendering translucent materials. *Acm Transactions on Graphics* 30, 4, 1–14.
- DOBASHI, Y., IWASAKI, W., ONO, A., YAMAMOTO, T., YUE, Y., AND NISHITA, T. 2012. An inverse problem approach for automatically adjusting the parameters for rendering clouds using photographs. *Acm Transactions on Graphics* 31, 6, 439–445.
- EBERT, D. S., AND PARENT, R. E. 1990. Rendering and animation of gaseous phenomena by combining fast volume and scanline a-buffer techniques. *Acm Siggraph Computer Graphics* 24, 4, 357–366.
- FRISVAD, J. R., HACHISUKA, T., AND KJELDSEN, T. K. 2014. Directional dipole model for subsurface scattering. *Acm Transactions on Graphics* 34, 1, 1–12.
- GARDNER, G. Y. 1985. Visual simulation of clouds. *Acm Siggraph Computer Graphics* 19, 3, 297–304.
- GKIOULEKAS, I., ZHAO, S., BALA, K., ZICKLER, T., AND LEVIN, A. 2013. Inverse volume rendering with material dictionaries. *Acm Transactions on Graphics* 32, 6, 1210–1214.
- GKIOULEKAS, I., XIAO, B., ZHAO, S., ADELSON, E. H., ZICKLER, T., AND BALA, K. 2014. Understanding the role of phase function in translucent appearance. *Acm Transactions on Graphics* 32, 32, 13–15.
- GUSTAVSON, S. 2005. Simplex noise demystified. *Tne027 Digital Communication Electronics Lab2*.
- HACHISUKA, T., JAROSZ, W., GEORGIEV, I., KAPLANYAN, A., NOWROUZEZAHRAI, D., AND SPENCER, B. 2012. State of the art in photon density estimation. In *SIGGRAPH Asia*, 1–562.
- HEITZ, E., DUPUY, J., CRASSIN, C., AND DACHSBACHER, C. 2015. The sgx microflake distribution. *Acm Transactions on Graphics* 34, 4, 1–11.
- HULST, V. D., AND H., C. 1958. Book reviews: Light scattering by small particles. *Science* 127.
- ISHIMARU, A., AND ISHIMARU, A. 1978. Wave propagation and scattering in random media. repr. of the 1978 orig. *Wave Propagation and Scattering in Random Media* 1, 6, 407–460.
- JAKOB, W., ARBREE, A., MOON, J. T., BALA, K., AND MARSCHNER, S. 2010. A radiative transfer framework for rendering materials with anisotropic structure. *Acm Transactions on Graphics* 29, 4, 157–166.
- JAKOB, W., 2010. Mitsuba renderer.
- JENSEN, H. W., MARSCHNER, S. R., LEVOY, M., AND HANRAHAN, P. 2002. A practical model for subsurface light transport. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, 511–518.
- KAWAI, J. K., PAINTER, J. S., AND COHEN, M. F. 2010. Radiotimization - goal based rendering. In *Computer Graphics, Conference Series*, 147–154.
- KLASSEN, R. V. 1987. Modeling the effect of the atmosphere on light. *Acm Transactions on Graphics* 6, 3, 215–237.

- LAFORTUNE, E. P., AND WILLEMS, Y. D. 1996. Rendering participating media with bidirectional path tracing. *Eurographics* 96, 91–100.
- LI, D., SUN, X., REN, Z., LIN, S., TONG, Y., GUO, B., AND ZHOU, K. 2013. Transcut: Interactive rendering of translucent cutouts. *IEEE Transactions on Visualization and Computer Graphics* 19, 3, 484–494.
- MAX, N. L. 1986. Atmospheric illumination and shadows. *Acm Siggraph Computer Graphics* 20, 4, 117–124.
- MISHCHENKO, M. L., TRAVIS, L. D., AND LACIS, A. A. 2006. Multiple scattering of light by particles. In *Multiple Scattering of Light by Particles*.
- MUKAIGAWA, Y., SUZUKI, K., AND YAGI, Y. 2009. Analysis of subsurface scattering based on dipole approximation. *Ipsj Transactions on Computer Vision and Applications* 1, 1, 128–138.
- NISHITA, T. 1987. A shading model for atmospheric scattering considering luminous intensity distribution of light sources. *Acm Siggraph Computer Graphics* 21, 4, 303–310.
- PAPAS, M., REGG, C., JAROSZ, W., BICKEL, B., JACKSON, P., MATUSIK, W., MARSCHNER, S., AND GROSS, M. 2013. Fabricating translucent materials using continuous pigment mixtures. *Acm Transactions on Graphics* 32, 4, 96–96.
- PATTANAIK, S. N., AND MUDUR, S. P. 1993. Computation of global illumination in a participating medium by monte carlo simulation. *Journal of Visualization and Computer Animation* 4, 3, 133–152.
- PELLACINI, F., BATTAGLIA, F., MORLEY, R. K., AND FINKELSTEIN, A. 2007. Lighting with paint. In *AMC Transaction on Graphics*, 9.
- PERLIN, K. 1985. An image synthesizer. *Acm Siggraph Computer Graphics* 19, 3, 287–296.
- PERLIN, K., 2001. Noise hardware.
- PHILIPPE, B., BERTR, S., AND CHRISTOPHE, S. 1993. A rendering algorithm for discrete volume density objects. In *Computer Graphics Forum*, 201–210.
- SAKAS, G. 1990. Fast rendering of arbitrary distributed volume densities. In *Eurographics*.
- SAKAS, G. 1993. Modeling and animating turbulent gaseous phenomena using spectral synthesis. *Visual Computer* 9, 9, 200–212.
- SCHOENEMAN, C., DORSEY, J., SMITS, B., ARVO, J., AND GREENBERG, D. 1993. Painting with light. In *Conference on Computer Graphics and Interactive Techniques, SIGGRAPH*, 143–146.
- SIEGEL, R., HOWELL, J. R., AND SIEGEL, R. 1992. Thermal radiation heat transfer - third edition. *Bristol, PA (United States); Hemisphere Publishing*.
- STAM, J. 1995. *Multiple scattering as a diffusion process*. Springer Vienna.
- STAM, J. 1996. Multi-scale stochastic modelling of complex natural phenomena. *Thesis Department of Computer Science University of Toronto*.
- WANG, J., ZHAO, S., TONG, X., LIN, S., LIN, Z., DONG, Y., GUO, B., AND SHUM, H. Y. 2008. Modeling and rendering of heterogeneous translucent materials using the diffusion equation. *Acm Transactions on Graphics* 27, 1, 329–339.
- WANG, J., TONG, X., LIN, S. S., GUO, B., SHUM, H. Y., AND LIN, Z., 2012. Modeling and rendering of heterogeneous translucent materials using the diffusion equation.
- YAEGER, L., UPSON, C., AND MYERS, R. 2011. R.: Combining physical and visual simulation-creation of the planet jupiter for the film 2010. *Acm Siggraph Computer Graphics* 150, 6, 347–8.
- ZHAO, S., RAMAMOORTHI, R., AND BALA, K. 2014. High-order similarity relations in radiative transfer. *Acm Transactions on Graphics* 33, 4, 1–12.