doi:10.3788/gzxb20134207.0857

基于三层材质模型的面部真实感交互式渲染

杨蒙召[1,2],王宽全[1],左旺孟[1]
(1 哈尔滨工业大学 计算机学院,哈尔滨 150001)
(2 黑龙江科技大学 计算机与信息工程学院,哈尔滨 150027)

摘　要:基于三层材质模型,设计了实时交互式渲染平台,利用基于物理的双向反射皮肤分布函数,模拟了光线在皮肤表面的反射.采用高斯模糊线性之和与纹理坐标扭曲纠正,有效地模拟了光线的次表面散射效果.结合表面反射和次表面散射,得到了最终的面部真实感实时渲染.该平台具有友好的交互式界面,在光线旋转时,可以查看反射和散射的效果,实现实时添加高动态范围背景、调节光照强度或者模糊程度等基本交互操作,更好地辅助用户对虚拟人进行控制和设计.

关键词:面部真实感;交互式渲染;三层材质模型;次表面散射
中图分类号:TP391.41　　**文献标识码**:A　　**文章编号**:1004-4213(2013)07-0857-7

Interactive Rendering for Realistic Face via Three-layer Material Model

YANG Meng-zhao[1,2], WANG Kuan-quan[1], ZUO Wang-meng[1]
(1 *School of Computer Science and Technology, Harbin Institute of Technology, Harbin* 150001, *China*)
(2 *School of Computer and Information Engineering, Heilongjiang University of Science and Technology, Harbin* 150027, *China*)

Abstract: An interactive rendering platform is designed based on three-layer material model. The specular reflection of human skin is approximated based on the bidirectional reflectance distribution function, and the subsurface scattering is effectively simulated using sum-of-Gaussians blur and texture coordinates stretch correction. Combining specular reflection with subsurface scattering, the final real-time rendering for realistic face is obtained. The interactive platform provides a friendly interface for user to operate digital character, views the reflection and scattering effect when the incoming light is rotating, adds high-dynamic range environment map in real-time, and adjusts intensity of illumination and degree of blur, which greatly help user to control and design digital character.

Key words: Realistic face; Interactive rendering; Three-layer material model; Subsurface scattering

0 Introduction

Realistic appearance from digital character can enhance sense of reality in 3D games and give lifelike in film production, computer-aided instruction and other fields, so we should firstly simulate the interaction between incoming light and human skin, and develop efficient method to give a realistic looking with existing GPU pipelines. Human skin is a highly scattering and anisotropic translucent material, so when incoming light arrives at surface of the skin, some small fraction of the incident light is reflected firstly. Most incoming light enters into the skin, brings multiple refraction and Subsurface Scattering (SSS), eventually exits from skin surface. Only

Foundation item: The National Natural Science Foundation of China (Nos. 61173086, 61179009)
First author: YANG Meng-zhao (1980－), male, lecturer, Ph. D. degree candidate, mainly focuses on optical information processing, realistic rendering. Email: yangmengzhao@gmail.com
Responsible author (Corresponding author): WANG Kuan-quan (1964－), male, professor, Ph. D. degree, mainly focuses on virtual reality, optical information processing and bio－computing. Email: wangkq@hit.edu.cn
Received: Apr. 1, 2013; **Accepted**: May 14, 2013

little remainder light in the skin is absorbed by muscle tissue or blood at last. So the scattering properties of the skin are more important than the absorption and reflection. For physical scattering process of light in skin subsurface is very complex, optical model and material model of the skin are difficult to be approximated and established.

Recently, light diffusion theory[1] was firstly presented to simulate multiple scattering to explain SSS effect. Then dipole model[2] was effectively presented to approximate light diffusion based on diffusion theory[1], which was quickly adopted in the practical application. Further, the dipole model was extended by multi-dipoles model[3] in multi-layer skin and which obtained a very realistic rendering. Later the spectral shading model[4] was provided for spectral analysis to interaction between incident light and skin, and shaded the skin color using four parameters. However, these methods have to take a long time to render human face and don't apply to real-time rendering. Nowadays, with the increasing requirements of modern 3D technology, the main challenge of rendering is not only to approximate the complex SSS to give a realistic looking but also develop sufficiently efficient method to allow for real-time rendering and easy to be implemented. In the movie "The Matrix", Borshukov and Lewis[5] used Gaussian function with a customizable kernel to simulate the SSS in the texture space. The technique naturally mapped onto GPUs, but it still failed to capture most complex subtleties of multiple scattering within skin. Gosselin and Green separately used Poisson sampling function[6] and Gaussian blur[7] to approximate the skin in real-time, but their techniques were not based on multi-layer translucent material and were only rough approximation of true scattering. Recently, based on physical structure of skin, NVIDIA Corporation[8] realized a much more realistic rendering method for skin in texture space and which is still a gold standard for user to judge rendering effect. Compared with NVIDIA method, Jimenez et al.[9] provided a solution to simulate SSS in screen space at interactive frame rates, and their technology simplified the rendering process and acquired the effect which approximates to texture space rendering. YANG et al.[10] proposed a 3D morphing target method based on GPU for real-time animation of facial expressions

Our focus differs in that considering three-layer material properties of true human skin, we simulate the specular reflection based on physical Bidirectional Reflectance Distribution Function (BRDF) which scales the relation between incoming light and exiting light at the same point, and approximate subsurface scattering in real-time via six sum-of-Gaussians and texture coordinates (UV coordinates) stretch correction. Also, we design an interactive rendering platform for realistic face of digital character. Using the platform we can view the dynamic effect of reflection and scattering when incident light is rotating around digital character, operate the digital character, enhance reality of scene by adding High-Dynamic Range (HDR) environment map and adjust intensity of illumination and degree of blur, etc. The interaction operation can greatly help user to design and make realistic digital character.

1 Three-layer material model and interaction with incident light

Aim to human skin material, Donner and Jensen[2] firstly pointed out that single layer was not enough to simulate skin appearance, then two-layer material model was presented to shade the skin. Eugene d'Eon et al.[8] introduced three-layer material model by improving two-layer model, which includes oily layer, epidermis layer and dermis layer as shown in Fig. 1. Each layer within the skin actually absorbs and scatters light differently which independently affects the appearance of skin. Reflection and scattering process of incoming light[8] are also displayed in Fig. 1.

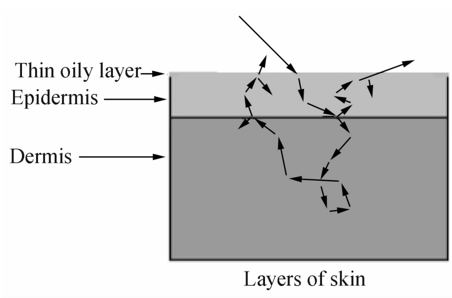

Fig. 1 Three-layer skin model and interaction with incoming light

Oily layer is a thin uppermost layer of the skin surface and is widely distributed in different physical person. When androgen of person secretes more and more, it will cause sebum to give increased and active secretion. At last, sebum exudates along the pores, covers the surface layer of human skin and become a thin oily layer. Oily layer is dielectric and reflects incoming light directly and only, so it doesn't color and shade the skin.

Epidermis layer which is also said epithelium is consisted of keratinization and dead cells. Epidermis contains melanin which truly gives influence to skin color. Melanin is also chromophore which affects the epidermis transparency only in the visible portion of the spectrum. Besides little reflection light on the oily layer, most light enters into the skin and produces single scattering and multiple scattering.

Dermis layer is consisted of collagen fibers, adipose tissue, proteoglycans, blood vessels, lymphatic vessels and nerves[11]. Compared with epidermis, dermis contains little cells, and collagen and elastin of dermis is all protein. In addition, the blood vessels in this layer include heme which affects the reflection ratio of skin. Little remainder light which travels through the oily layer and epidermis layer continues to scatter in this layer and only small fraction of light is absorbed, however most light return to epidermis and finally exits from skin surface.

2 Specular reflection approximation

For the outermost layer of skin is oily layer, small fraction of incident light doesn't enter into the skin and is reflected directly. According to the statistical results of spectrum, the specular light on the oily layer only reflects about 6 percent of the whole light spectrum, and thus reflection light does not dominate the final effect of human face, however it can represent high-frequency information of skin appearance, such as surface details, distribution of high gloss[12-13] etc.

In this paper, we present a physically based BRDF model which is combined Beckmann distribution function[14] with the theory of BRDF pre-computation. The model can approximate Torrance/Sparrow model closely, and using decomposition and pre-computation of BRDF, we can obtain realistic effect of reflection in real-time.

The physically based BRDF model contains a Fresnel term[15], and the Fresnel term describes the amount of reflection when normal incident light arrives at the skin surface given as in

$$\text{fresnel Reflectance}\ (H,V,F_0)=(1-\cos\ (H\cdot V))^5+(1-(1-\cos\ (H\cdot V))^5)\times F_0 \quad (1)$$

where H is the half-angle vector between incoming light direction and view direction, V is view direction, F_0 is the parameter for reflection value of normal incident light, and for skin material the value is 0.028 which has been measured[4].

Beckmann distribution function can be defined as

$$\text{Beckmann}(N,H,m)=\exp\ (-\tan^2(\arccos\ (N\cdot H))/m^2)/(m^2\times(\arccos\ (N\cdot H))^4) \quad (2)$$

where N is normal at the point of incoming light, H is the half-angle vector between incoming light direction and view direction, m is the roughness value over the human face.

Then we compute the 2D texture based on Beckmann distribution function as follows

$$\text{BeckmannTex}=0.5\times\text{Beckmann}(N,H,m)^{0.1} \quad (3)$$

Here, we adjust the value of Beckmann distribution function by using an exponential term to scale and halve it. Thus the value can be mapped into a range[0, 1] and stored in an 8-bit texture. At last the mapped value is inverted and computed in pixel shader.

Now we can evaluate value of BRDF reflection combined with Fresnel term as in

$$\text{frSpec}=\max(\text{BeckmannTex}\times\text{fresnelReflectance}/H\cdot H,0) \quad (4)$$

Based on different direction of view and intensity, we can obtain the specular result on different region of face

$$\text{specularRes}=\text{frSpec}\times\text{rho}\times\text{staturate}(N\cdot H) \quad (5)$$

where rho is different specular intensity over human face. Dot product of N and H scales the attenuation degree when incoming light enter into skin. The function staturate (x) which is in standard function library of CG returns a value between 0 and 1.

Finally, by combining light color with shading shadow, specular reflection of incoming light can be simulated as follows

$$\text{specularLight}=\text{lightColor}\times\text{lightShadow}\times\text{specularRes} \quad (6)$$

where lightColor is the color of illumination light. lightShadow is the shading shadow in our scene.

3 Subsurface scattering approximation

Besides little reflection light, most light enters into epidermis layer and dermis layer of the skin and then causes multiple refraction and scattering[16]. After the scattering light exits from surface of the skin, they arrive at the human eyes. This fraction of the incident light accounts about 90 percent of the whole light spectrum, so they finally dominate the final face effect.

By observing the diffusion profile of light scattering and analyzing the process of scattering attenuation, we find that the diffusion profile is similar to Gaussian curve. Although a single Gaussian doesn′t accurately fit any diffusion profile, we have found that sum of six Gaussians can provide an excellent approximation. Gaussian has also some advantages for GPU realization such as simultaneously separable, radially symmetric and new Gaussian appearance after convoluting two Gaussians, so we can compute Gaussian convolution in real-time using GPU technology.

For three-layer skin, we can use six linear sum-of-Gaussians to approximate diffusion profile[8]

$$R(r)=\sum_{i=1}^{6}\omega_i G(v_i, r) \tag{7}$$

where r is the distance between the position of incoming light arriving at skin surface and the position of scattering light exiting from skin surface, w is the weight to scale proportion of Gaussian convolution, and v is variance to scale deviation degree of light scattering.

Here Gaussian function with variance v can be defined as in

$$G(\nu, r):=(1/2\pi\nu)\times\exp(-r^2/2\nu) \tag{8}$$

Two parameters of weight w and variance v in Eq.(7) to match three-layer skin model has been measured in Ref.[3] and also shown in three color RGB channel in Fig. 2 from GPU gems3[8]. With the Gaussian sums that fit our red, green, and blue diffusion profiles, we use the relative weighting of the Gaussians in each sum, as well as the exact variance of each Gaussian in the sum, in our shading system.

	Variance/(mm²)	Red	Blur weights green	Blue
	0.0064	0.233	0.455	0.649
	0.0484	0.100	0.336	0.344
	0.187	0.118	0.198	0
	0.567	0.113	0.007	0.007
	1.99	0.358	0.004	0
	7.41	0.078	0	0

Fig. 2 Parameters for three-layer skin model

From Fig. 2, we have noticed that the diffusion profiles are strongly color dependent and red light scatters much farther than green and blue . By normalizing these profiles to have a diffuse color of white, we ensure that the result, after scattering incoming light, remains white on average. Then we multiply this result by a photo-based color map to get a skin tone as follows

$$I\times R(r)=I\times\left(\sum_{i=1}^{6}(\omega_i G(\nu_i, r))\right)=\sum_{i=1}^{6}(\omega_i I\times G(\nu_i, r)) \tag{9}$$

where I is initial irradiance map for 3D head mesh.

The convolution of two Gaussians is also a Gaussian given as

$$G(v_1, r)\times G(v_2, r)=\int_0^{\infty}\int_0^{\infty}G(v_1, \sqrt{x'^2+y'^2})\times G(v_2, \sqrt{(x-x')^2+(y-y')^2})\,\mathrm{d}x'\mathrm{d}y'=G(v_1+v_2, r) \tag{10}$$

So this allows us to generate each irradiance texture by convolving the result of the previous one. If the previous irradiance texture contains $I*G(v_1, r)$ and now we wish to compute $I*G(v_2, r)$, we simply convolve with $G(v_2-v_{1.}\, r)$ which greatly reduce the amount of computation.

$$I\times G(v_2, r)=I\times(G(v_1, r)\times G(v_2-v_1, r))=(I\times G(v_1, r))\times G(v_2-v_1, r) \tag{11}$$

In additions, for highly curved regions on the face surface such as brow and nose, distances between two locations in texture space do not correspond directly to distances on the 3D head mesh due to UV distortion, thus the SSS would not be accurate. We need address UV distortion to accurately handle convolution over curved regions. When evaluating each frame, we pre-compute a stretch-correction texture and modulate the space of convolution taps at each point on the surface according to the value in the stretch-correction texture which can correct this problem.

4 Shader Implementation based on GPU

In our GPU rendering pipeline, the vertex shader simply transforms position and normal of each vertex on digital human head from object space to world space, which usually uses the specific model-view matrix and some parameters passed from the C++ program. Then, in the fragment shader each pixel color of human head is evaluated, so all the specular reflection and subsurface scattering are implemented in the fragment shader. At last, we pass the final pixel color of human skin obtained in the fragment shader to the C++ program and use OpenGL rendering function to display rendering result.

Algorithm 1 shows the procedure to compute specular reflection in the fragment shader as follows

Algorithm 1 Specular Reflection in the Fragment Shader
Input: The incoming light direction L and view direction V passed from C++ program. Specular intensity over human face, rho. 2D texture based on Beckmann distribution function, BeckmannTex.
Output: Specular light color of human head, specularLight.
1: compute and normalized the half-angle vector, H=normalize(L +V); 2: evaluate the dot product between half-angle vector and normal, and view direction, ndoth=dot(N, H), cosehv=dot(H, V); 3: evaluate the pixel value of Beckmann texture, BT = pow (2 * tex2D (BeckmannTex, float2 (ndoth,m)),10); 4: compute the Fresnel term, F = fresnelReflection(cosehv, 0.028); 5: evaluate the BRDF reflection, frSpec = max (BT * F/dot(H, H), 0) 6: compute result, specularRes = NdotL * rho * frSpec; 7: obtain the final specular after adding shadow, specularLight += lightColor * lightShadow * specularRes; 8: returnspecularLight;

Algorithm 2 shows the procedure to compute subsurface scattering in the fragment shader as follows

Algorithm 2Subsurface Scattering in the Fragment Shader
Input: The blurred irradiance texture, irradTex_i, obtained through Gaussian convolution and UV correction. The specular reflection, specularLight.
Percentage of diffuse color, s_diffColMix. The sum of six Gaussian weights, normConst.
Output: Final skin color of human head, finalLight.
1: computesix blurred irradiance texture, irradTex_i, using (9) via Gaussian convolution shader and UV stretch-correction shader; 2: evaluate sum-of-Gaussians to obtain the total diffuse light exiting the skin surface, diffuseLight += gaussw * irrad1tap_i.xyz; 3: renormalize diffusion profile to white, diffuseLight/=normConst; 4: Determine skin color from a diffuse color map diffuseLight* = pow(f3tex2D(diffuseColorTex, texCoord), 1.0 -s_diffColMix); 5: Compute final skin color combing specular reflection with subsurface scattering finalLight =float4(diffuseLight+specularLight, 1.0) 6: return finalLight;

5 Design and analysis for interactive rendering platform

With CPU: AMD Athlon II X4 Four Cores, GPU: NVIDIA GeForce GT430 @2G, RAM: 2G, C++ &OpenGL, Shader Language: CG and VS2008, we have achieved final realistic rendering for human face. Also, we have designed an interactive rendering platform for user to friendly debug and operate the 3D digital character.

The scanned 3D head mesh in our paper contains 8 844 vertexes and 17 674 triangular facets, and initial irradiance texture map for head mesh is shown in Fig. 3.

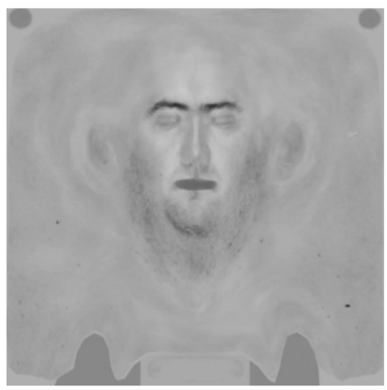

Fig. 3　Irradiance map of human face in texture space

We pre-compute the BRDF model to simulate the specular reflection and store specular reflection into 2D texture in advance. Now we approximate the subsurface scattering in texture space as follows. We firstly obtain five UV stretch-map textures and implement Gaussian blur for initial irradiance map. By adding the stretch-map textures, we can acquire six Gaussian blur textures, then according to Eq. (9) six linear sum-of-Gaussians is obtained to approximate subsurface scattering result. By adding specular reflection with subsurface scattering, we get the final realistic rendering for face shown in Fig. 4.

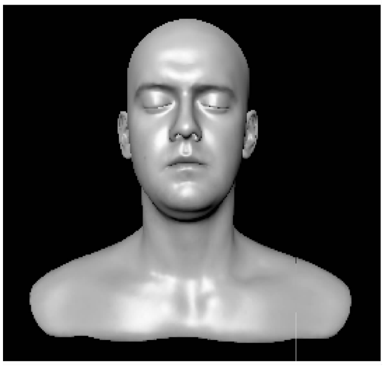

Fig. 4 Final realistic rendering for face in real-time

In order to help user to easily operate and control the digital character in the practical application, we design an interactive interface and can implement some basic interactive operations. Through interaction, we can view the different effects of digital character when the incoming light is changing or fixed.

1) User can easily rotate and control the digital character in real-time using mouse when point light source is fixed as shown in Fig. 5.

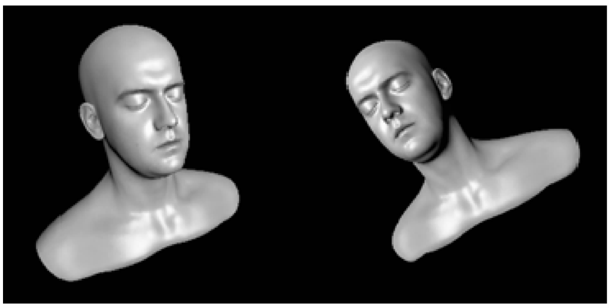

Fig. 5 Controlling and rotating digital character in real-time

2) We can display the specular reflection shown in Fig. 6 and realistic rendering for face as shown in Fig. 7 in real-time when incoming light is rotating around digital character. The outermost oil layer of skin is dielectric material that only reflects light without coloring it. Thus, a physically based skin shader should use a white specular color.

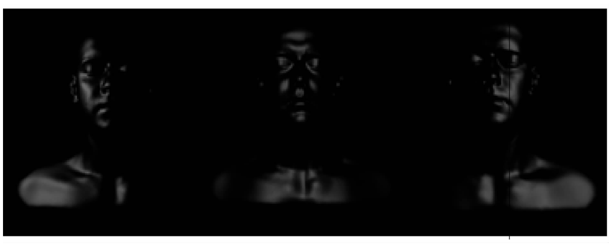

Fig. 6 Specular reflection for face in real-time when light rotation

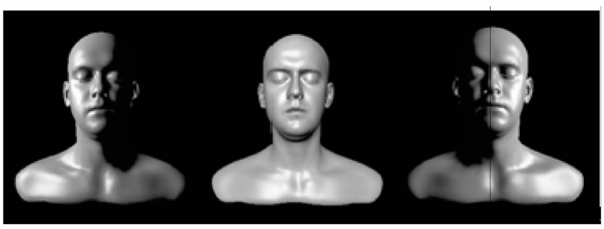

Fig. 7 Realistic face rendering in real-time when light rotation

3) We can achieve the quick shift between 3D head mesh and digital character in real-time shown in Fig. 8 when light is rotating around digital character.

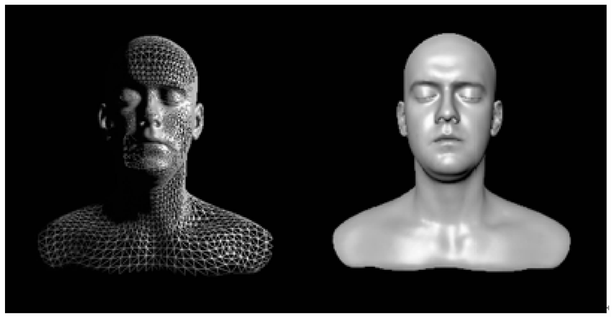

Fig. 8 Shift between 3D head mesh and digital character in real-time when light rotation

4) We can enhance the sense of reality for scene by adding HDR environment map shown in Fig. 9, which can make the digital character has realistic and immersion in 3D game development.

Fig. 9 Enhancing sense of reality by adding HDR environment map in real-time

5) We cannot discuss all the results of each interactive operation for lack of space and finally show the main window for interactive rendering as shown in Fig. 10. For interacting and debugging operation easily, we can select corresponding option to view glow effect, environment lighting when adding the sky box or debugging mode containing the depth map, six results of Gaussian blur pass and five results of UV stretch and so on. Also when we click right button on the rendering window, some menus will appear for us to adjust the exposure, blurring amount and width, and toggle test texture to view different rendering effect of human face.

Fig. 10 Whole interactive rendering platform for realistic face in glow effect

Combing specular reflection with subsurface scattering only, we have obtained approximately 15 Frames Per Second (FPS) to implement realistic rendering. When adding HDR environment map, the rendering speed reduced to about 12 FPS. At last, when implementing the whole rendering after adding light rotation and environment map, the final speed is approximately 6 FPS. Thence, the average speed of our rendering is about 10 FPS.

6 Conclusions

This paper has introduced complex material properties of human skin and discussed the interaction between skin and incoming light. By adding specular reflection approximation with subsurface scattering approximation, we have got the final realistic rendering for face. Also we have designed an interactive rendering window for user to operate the 3D digital character. This can greatly enhance the understanding for interaction between light and skin, and help designers to make virtual character easily which has a great practical value.

Even though we see our work as a step in allowing designers to quickly render and interact with realistic digital character, how the more realistic appearance is approximated is an interesting direction which we are currently working on.

References

[1] STAM J. Multiple scattering as a diffusion process[C]. Proceeding of Eurographics Rendering Workshop, 1995: 41-50,

[2] JENSENH W, STEPHEN R, MARSCHNER M L. A practical model for subsurface light transport[C]. Proceedings of ACM Computer Graphics and Interactive Techniques, 2001: 511-518.

[3] DONNER C, JENSEN H W. Light diffusion in multi-layered translucent materials[J]. *ACM Transactions on Graphics*, 2005, **24**(3): 1032-1039.

[4] DONNER C, JENSEN H W. A spectral BSSRDF for shading human skin[C]. Proceeding of Eurographics Symposium on Rendering, 2006: 409-417.

[5] BORSHUKOVG, LEWIS J. Realistic human face rendering for the matrix reloaded[C]. Proceedings of SIGGRAPH Sketches and Applications, 2003: 1-5.

[6] GOSSELIN. Real time skin rendering[C]. Proceeding of Game Developer Conference, D3D Tutorial, 2004: 120-132.

[7] GREEN S. Real-time approximations to subsurface scattering [M]. GPU Gems, Boston: Addison-Wesley Press, 2004: 263-278.

[8] D'EONE, LUEBKED. Advanced techniques for realistic real-time skin rendering[M]. GPU Gems 3, Boston: Addison-Wesley Press, 2007: 293-347.

[9] JIMENEZ J, SUNDSTEDT V, GUTIERREZ D. Screen-space perceptual rendering of human skin[J]. *ACM Transactions on Applied Perception*, 2009, **23**(6): 1-15.

[10] YANG M Z, WANG K Q, ZHANG L. Realistic real-time facial expressions animation via 3D morphing target[J]. *Journal of Software*, 2013, **8**(2): 418-425.

[11] PAMPLONA D C, CARYALHOC C R. Characterization of human skin through skin expansion [J]. *Journal of Mechanics of Materials and Structures*, 2012, **7**(7): 641-655.

[12] TIAN Dong-wen, ZHANG Yi-xin, WANG Qing-juan. Color precise prediction model of spectral reflectance for recto-verso halftone images[J]. *Acta Photonica Sinica*, 2010, **39**(11): 1982-1987.

[13] LI Xu-yang, YANG Hong-tao, HE Tian-bing, *et al*. Design of a new type on-axis three-mirror-anastigmat optical system [J]. *Acta Photonica Sinica*, 2012, **41**(1): 31-35.

[14] KELEMEN C, SZIRMAY-KALOS L. A microfacet based coupled specular-matte BRDF model with importance sampling[C]. Proceeding of Eurographics, 2001: 34-43.

[15] SCHLICK C. A customizable reflectance model for everyday rendering[C]. Proceeding of Eurographics Workshop on Rendering, 1993: 73-84.

[16] DING Ying, TONG Shou-feng, DONG Ke-yan, *et al*. Study and simulation of atmospheric UV communication performance with vertical transmitter-receiver [J]. *Acta Photonica Sinica*, 2010, **39**(10): 1851-1856.