Research on the application of real time rendering technology in Chinese meticulous painting style

Yan Zhang[1, a *] Aijin Li[2,b]

[1]Art and Media School, [2]Department of Computer Science and Technology

[1,2]Century College,Beijing University of Posts and Telecommunications , 102101

Yanqing County, Beijing, China

[a]zhangyanlaj@hotmail.com, [b]liaijingzy@hotmail.com

Keywords: Non realistic graphics Real-time rendering Game design

Abstract. The research of real sense graphics has made great progress in recent years.Not only the game screen has been able to achieve a very real effect， in other fields such as virtual reality, 3D animation film and other fields have been fully applied with this technique.The development of non real sense graphics bring the new user the visual experience. In the non real sense rendering effect, it can achieve a lot of personalized visual effect,bring a new visual experience to the user, and it can be integrated into the national and regional elements in the game design.

Introduction

The non realistic graphics have been proposed in a long time ago.A french researcher called Decaudin Philippe wrote a lecture "Rendering of 3D-Scenes Cartoon-Looking" which was published in 1996. Since then, a lot of scientists are working on this research and made considerable development.Then this technology had being applied to the game design(As shown in Figure 01and Figure02).

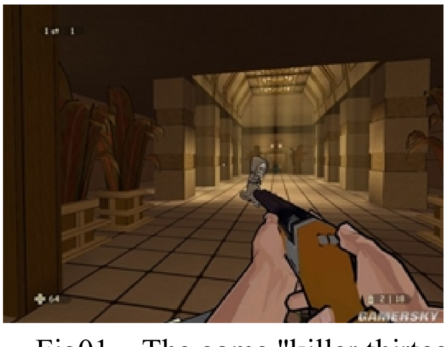

Fig01 The game "killer thirteen"

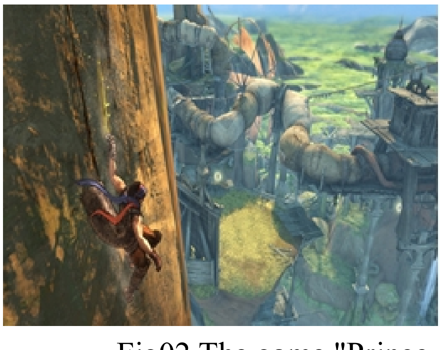

Fig02 The game "Prince of Persia"

The study started later in China.Animation director Xu Yi’s work which named "Oder to Summer"was selected as the excellent works in SIGGRAPH 2003, he had becoming the first person in the world's top CG summit in China.Xu Yi uses the traditional Maya modeling method to achieve the traditional Chinese ink painting style effect(As shown in Figure 03).

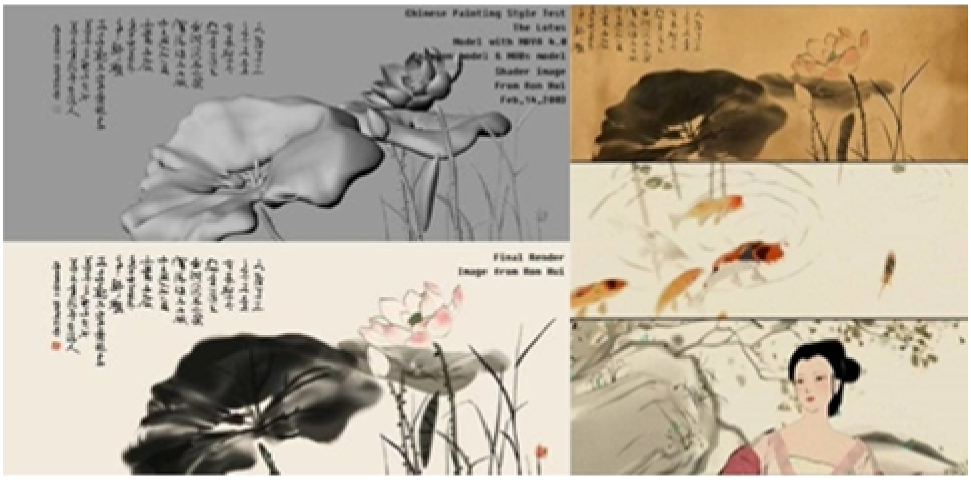

Fig03 Xu Yi's work "Oder to Summer"

This technology gradually popular with Chinese game companies in recent years, like the "The Scar in the Sky"and"Xuanyuan Sword 6".Famous Chinese director John woo had supervised the manufacture of the production of Chinese ink style action hand game which named "Bloody angel"(As shown in Figure 04 and Figure 05).

Fig04 The game"The Scar in the Sky"

Fig05 The game"Bloody angel"

The simulation of Chinese traditional meticulous painting is still blank.In this paper, we will study the proposition that how to realize it.

Research method:

We will take the tree aspects to solve this problem,they are the the rendering of meticulous painting style, extraction of outline, and the definition of two-dimensional texture.

The rendering of meticulous painting style

BRDF (bidirectional reflectance distribution function) is a relatively simple diffuse illumination model.The reflection formula as shown below:

$$L_o(p, w_o) = \int_{2\pi^+} f_r(p, w_i, w_o) L_i(p, w_i) \cos\theta_i \mathrm{d}w_i$$

This formula simulated the process of the light reflection of point p in object surface, fr(p,w_i,w_0)be used as the ratio of bidirectional reflectance distribution function (BRDF), incident radiation of the point p expressed as L_i(p,w_i), w_i and w_0 respectively expressed as direction of incident light and the direction of the reflected light. d_{wi} used as differential cone and w_i is its axis, w_i and $w_0$form the Angle with normal vector of point p respectively expressed as θ_i and θ_0.

Completely diffuse (BRDF) was modified into a kind of important reflection type, namely the Lambertian reflection.Form the Lambertian reflection formula,the reflection radiation formula Lo = Lr, d (p) has nothing to do with w_o.When the w_i and w_o have none effect to the BRDF, we can define fr, d (p) and extracted elements from the formula of reflection, then form a new formula of Lambertian reflection:

$$L_{r,d}(p) = f_{r,d}\int_{\Omega i} \mathrm{Li}(\mathrm{p}, \mathrm{w_i})\cos\theta_i \mathrm{dw_i} = f_{r,\mathrm{d}}(\mathrm{p})\mathrm{E_i}(\mathrm{p})$$

(formula 4)

The following formula can be obtained by the above formula:

$$f_{r,d}(p) = L_{r,d}(p) / E_i(p)$$

（formula 5）

The formula reflects the ratio of the surface flux of entire hemisphere above the mesh and all internal the incident flux of reflection.

Reflectance that p (p, Ωi, Ωo) is defined as:

$$p(p,\Omega\, i,\Omega\, o) = d\Phi\mathrm{o}/\mathrm{d}\Phi\mathrm{i} = \frac{\int_{\Omega\, \mathrm{o}}\int_{\Omega\, \mathrm{i}} fr(p, w_i, w_o)Li(p, w_i)\cos q_i q_o dw_i dw_o}{\int_{\Omega\, i} Li(p, w_i)\cos q_i w_i}$$

（formula 6）

Simplified the formula 6 we can get the new reflectivity formula as follows:

$$p_d(p) = df_o / df_i = L_{r,d}(p)p \,/\, E_i(p) = f_{r,d}(p)p$$

（formula7）

Combined with the following formula 8:

$$f_{r,d}(p) = p_o(p)/p$$

（formula8）

We adopted double hemisphere reflectivity coefficient to p_{hh} mark the completely diffuse state.For completely diffuse reflection, we can regarded $L_i(p,w_i)=L_i$ as a constant, So we can extracted $L_i(p)$ from formula 6 and Simplify it:

$$p_{hh}(p) = \frac{fr,d(p)}{p}\int_{2p^+}\int_{2p^+}\cos q_i \cos q_o dw_i dw_o = pf_{r,d}(p) = p_d(p)$$

（formula 9）

We can find that in completely diffuse reflection model, existing relationships that $P_{hh}(p)=p_d(p)$. The scope of the l_s of $l_s\in$ [0, ∞], and it can reflect the light transmission of the radiation is proportional to the state of the $l_s c_i$. Only by using the ambient light to render the object only by a single color value, but the material contains the diffuse color will be mixed in the light of the color of the environment.

The incident radiation intensity of the ambient light can be expressed as:

$$L_i = l_s c_l$$

（formula10）

In the implementation of Lambertian, we use p_d to represent the RGB color values, there is:

$$p_d = k_d c_d$$

（formula11）

In the formula that the $k_d\in$[0, 1], It represents the coefficient of diffuse formula; c_d is the value of diffuse color . According to the formula 9 we can get the following formula:

$$p_{hh} = k_d c_d$$

（formula12）

The environmental illumination can be obtained by the formula 12 and the p_{hh} of the formula 10:

$$L_o(p,w_o) = p_{hh} * Li(p,w_i) = k_a c_d * (l_s c_l)$$
（formula 13）

The k_a of this formula represents the coefficient of ambient light reflection of Specific material , the value range is that $k_a \in [0, 1]$ .In order to convenient for the controlling, we believe that ambient light is diffuse reflection, so $k_a=k_d$.

The total reflection radiation of the point p can be obtained by adding the radiation intensity of the environment:

$$L_o(p,w_o) = k_a c_d * (l_s c_l) + \sum_{j=1}^{n} (k_d c_d / p) * (l_{s,j} c_{l,j})(n \cdot l_j)$$
（formula14）

In the above formula:

$$n \cdot l_j = \cos q\ ,\ \ q \in [0, p/2]$$

The extraction of outline:

Assuming that vector of the viewpoint is V, the two sides of the contour lines are Nback and Nfront. If Nfront · V > 0 and Nback · V < 0 it's not meet the requirements, and N · V= 0 satisfy the condition of the patch itself is tangent with the line of sight(As shown in Figure 06).

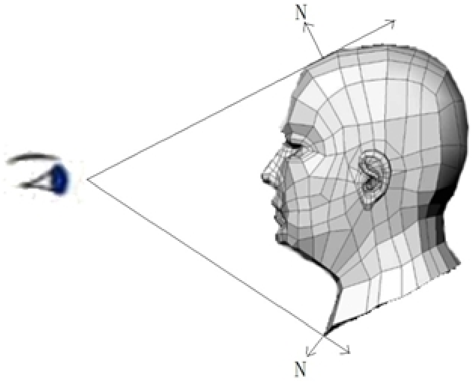

Fig06 Extraction of contour lines

We can judge the contour of 3D mesh model by using the conditions that（Nfront · V）×（Nback · V） < 0. At the same time, we need to determine the contour of 3D model surface which was depression, and it has been confirmed that when the angle which formed by two concave facets less than 75 degrees it will formed concave visually.

Assuming that the normal vector of the line of sight is n_f, the normal vector of the line of sight is n_b, then the detection method of the contour of grid object is:

$$(n_f \cdot l) \times (n_b \cdot l) < 0$$

The material object detects the the value of n*l that corresponding vertexes of the corresponding point at each point of the collision, only then n*l<0 became using the ambient light to render the corresponding vertexes.

We could obtained the vertex curvature by specific method that $k_v=k_i*k_{vvi}$, now we can improve the formula 14:

$$L_o(p,w_o) = k_a c_d * (l_s c_l) + \sum_{j=1}^{n} (k_d c_d / p) * (l_{s,j} c_{l,j}) * D_m k_v(j)$$

The color of mesh object has different effect when the value of k_a and k_d changed.According to the conservation of energy that $k_a+k_d<1.0$, we can get more rich color change effect. In accordance with the experience if we use$(k_a,k_d) \in [0,1]^2$ we can get the more details of the color effect.

Define two-dimensional texture:

The diffuse reflectance of the 3D model is defined to be 0.0-1.0, and suppose that the value of the color is 0 that means the material is totally not affected by the illumination.In the 3D modeling software, the general method is adopted the value that $k_a=k_d$ .After the treatment, the level of the five steps are 0.1, 0.3, 0.6, 0.8, 1, they are the most suitable values for the performance of fine brushwork style.The process can be expressed as follows:

$$k_s（k_a）= \begin{cases} 0.1 & k_a <= 0.18 \\ 0.3 & 0.18 < k_a <= 0.46 \\ 0.6 & 0.46 < k_a <= 0.71 \\ 0.8 & 0.71 < k_a <= 0.84 \\ 1.0 & k_a > 0.84 \end{cases}$$

The value k_s can be obtained by k_a after the steps processing.We put the k_s that into the formula:

$$L_o(p,w_o) = k_s C_d * (l_s c_d) + \sum_{j=1}^{n} (k_s c_d / \Pi) * (l_{s,j} c_{l,j}) * D_m k_v(j)$$ (formula 15)

Finally, we can get relatively satisfactory results.

Research result：

We used the Unity3d as the platform to achieve the above ideas. Unity 3d is equipped with a powerful shadow and material language tool called ShaderLab. In programming language, it is similar to the syntax of CgFX and Direct3D,but it not only records the basic endpoint or image point information, it also describes all the information necessary for the material.In the process of achieving the above algorithm, we use ShadeLab to achieve the control of the color effect. The following is a brief section of the source code:

```
 if ((0.0 == _WorldSpaceLightPos0.w)) {
     lowp vec3 tmpvar_8;
     tmpvar_8 = normalize (_WorldSpaceLightPos0.xyz);
     lightDirection = tmpvar_8;
  } else {
     lightDirection = normalize ((_WorldSpaceLightPos0 - xlv_TEXCOORD0).xyz);
  };
  highp vec3 tmpvar_9;
  tmpvar_9 = (gl_LightModel.ambient.xyz * _Color.xyz);
  highp vec3 tmpvar_10;
  tmpvar_10 = ((_LightColor0.xyz * _Color.xyz) * max (0.0, dot (tmpvar_2, lightDirection)));
  highp vec3 tmpvar_11;
  tmpvar_11 = cross (xlv_TEXCOORD1, xlv_TEXCOORD6);
  highp float tmpvar_12;
  tmpvar_12 = dot (tmpvar_2, lightDirection);
  if ((tmpvar_12 < 0.0)) {
    specularReflection = vec3(0.0, 0.0, 0.0);
  } else {
    specularReflection = ((_LightColor0.xyz * _SpecColor.xyz) * pow (max (0.0, dot (reflect (-(lightDirection), tmpvar_2), tmpvar_3)), _Shininess));
  };
```

The final effect is shown in Figure(As shown in Figure 08, Fig09):

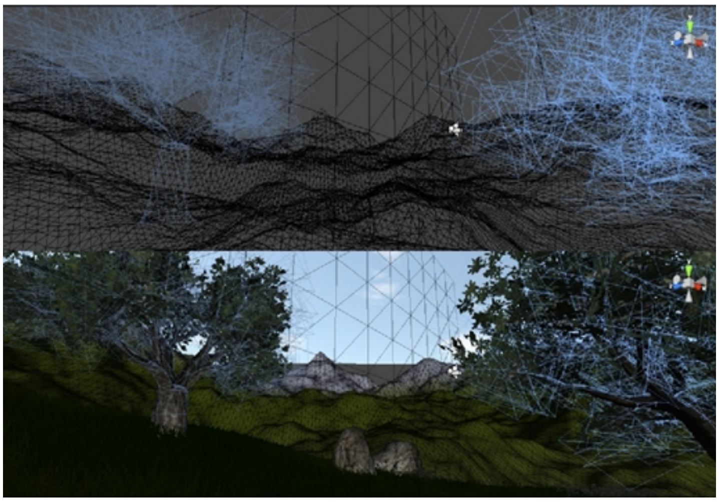

Fig 7 The transient process of mesh model to texture mapping

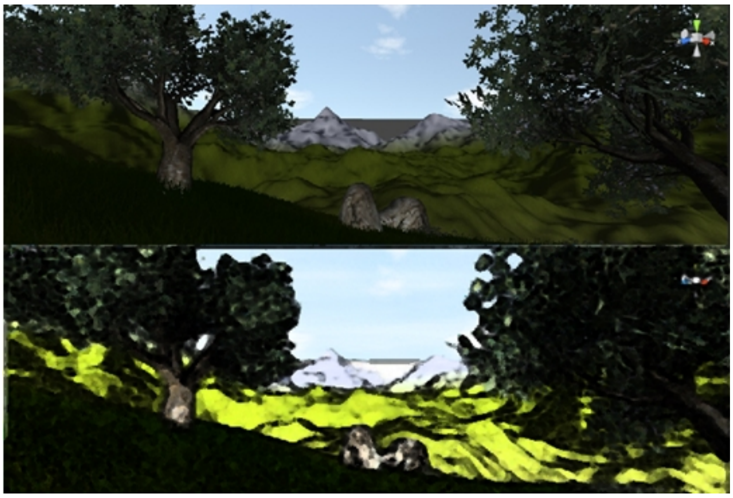

Fig 8 The transient process of real sense effect to meticulous painting style

After realization, we verify the correctness of the previous point of view.

conclusion:

In this paper, we propose a method that how to based on the features of surface to achieve the Chinese traditional meticulous painting style.The realization of the Chinese traditional meticulous painting style have the better effect and the process is relatively simple. It’s suitable for general situation that three-dimensional game design,and is a new attempt for the real-time rendering of 3D games.The realization process and drawing process is relatively simple, and retain more details of 3D model. It's the beneficial enrichment to the researching of real-time rendering and non-realistic computer graphics.

References

[1]Jame F.Blinn.Simulation of wrinkled surface.Computer Graphics(SIGGRAPH 78Processings), pp.286-292, August 1978.

[2]Jerry O.Talton, Darren Fitzpatrick.Teaching Graphics with the OpenGL Shading Language.Proceedings of the 38th SIGCSE Technical Symposium on Computer Science Education, March 7-10,2007.

[3]M.Kilgard.NVIDIA Graphics, Cg, and Transparency.SIGGRAPH 2006 Course Notes, 2006.

[4]Ian Stroud, Gabor Renner,Paul Xirouchakis, A divide and conquer algorithm for media surface calculation of planar polyhedra, Computer-Aided Design,2007,39(9):794-817.

[5]Forrester Cole, Doug De Carlo,Adam Finkelstein, Line Drawing from 3D models, SIGGRAPH 2008 Class, Aug.2008.