GPU-based 3D Oriental Color-Ink Rendering

Crystal S.Oh, Yang-Hee Nam
Division of Digital Media, Ewha Womans University
11-1 Daehyun-dong, Seodaemun-gu, Seoul, South Korea
osjung8@hotmail.com, yanghee@ewha.ac.kr

Abstract

Automated rendering technique in oriental painting style is demanding in the entertainment industry such as game, animation since this makes their contents more unique. To simulate the oriental painting more vivid, intrinsic color expression is critical. Generally actual artists put 3 colors on the different part of the brush at the same time and they express the volume and diffusion effect with this brush using peculiar color pigments. However most existing works have only focused on black ink painting and they rarely discuss about how to simulate the effect of actual pigments and their layered application. This paper presents a novel algorithm which can express the volume and diffusion of 3D objects using oriental color-ink model constructed from the real artists' standpoint. This model consists of 3 layers according to different color tones. They are combined using Kubelka-Munk (KM) composition model where optical parameters are extracted from the real painting media. We implemented our model on a GPU and the results show real-time rendering performance in arbitrarily given 3D scenes.

Key words: NPR, Oriental Painting, Kubelka-Munk, GPU.

1. Introduction

Non-photorealistic rendering researches have investigated aesthetic and stylized element into computer graphics by simulating various artistic styles such as cartoon, watercolor, oil painting. In particular, automated rendering techniques in oriental painting style are demanding in the entertainment industry such as game, animation since they makes the contents more unique.

Among several factors contributing oriental painting style, coloring method is very special. Generally traditional artists put 3 colors on the different parts of the brush at the same time and they express the tone and diffusion effect affected by correct orders and the way of compound application of these 3 colors. Moreover, the oriental color pigments have unique optical properties on account of using natural materials and their resulting form in solution. Most existing works, however, have only focused on the black ink diffusion and the representation of western color pigments. Therefore, their ink models and rendering frameworks are inappropriate for oriental color-ink style painting.

This paper proposes an oriental color-ink model based rendering method which provides an automated way of rendering arbitrary 3D models in oriental style which is also suitable for real-time application such as games and virtual environments. In our approach, we define a novel rendering framework that consists of 4 components for rendering arbitrary 3D models in oriental style : Color Composition, Volume & Color Diffusion expression, Silhouette expression and Depth & Paper expression. To simulate the rich oriental color, we focus on the following two main algorithms :

First, for realistic Color Composition effect, we provide how we could simulate the optical effect of superimposed 3 color layers using Kubelka-Munk (KM) compositing model by finding its coefficients from the true representation of real paint media.

Secondly, we propose a color-ink model consisting of 3-layers according to the different color tones (Base, Mid and Vivid tone) from the real artists' standpoints. To meet the requirements of the correct order and methods of applying color layers in oriental painting with KM composition colors, a new algorithm is suggested for expressing Volume & Diffusion using this 3-layered color model.

We implemented our oriental color-ink model based framework on a GPU, and we finally show the achievement of real-time rendering results where we can render a 212,452 faces at 21.7 fps on a 3.2Ghz, Intel 4P CPU with NVIDIA GeForce 6800 GPU.

2. Related Works

We will discuss some previous works related to oriental color ink rendering on two topics, which are 3D oriental black ink painting and color representation.

3D Model based Oriental Black Ink Rendering

The importance of colors in oriental painting is no less than black ink effects but most previous researches in this area focused on black ink painting. Kang suggested a real time 3D oriental black ink painting using a hardware accelerated rendering algorithm[1]. They represent 5 features of oriental painting as tone, diffusion, brush stroke, depth and paper. Zhang presented diffusion effect for 3D model using a simple behavioral model of ink[2]. It is based on a cellular automaton computation but they can not achieve real-time rendering and their experiment use only tree models so they have limitations to express whole features of oriental painting.

Way presented a methods to automatically draw trees in Chinese ink painting style from 3D polygon models[3]. They define outline rendering and texture generation for oriental painting but they only generate the textures for various trees so they also have limitations. To provide an automated way of rendering 3D models in oriental color ink style, we need to define the novel rendering frame-work that construct the whole process for oriental color ink painting that focus on the color expressions.

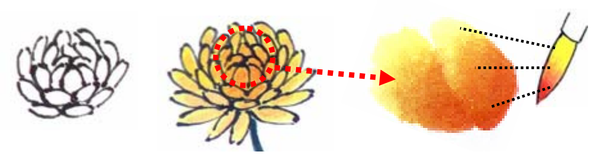
Color Composition Model

Recently, most NPR researches which express the color composition use the Kubelka-Munk(KM) model to compute a color mixtures of pigments since this model can display the color of the mixed pigments realistically. Curtis used KM model for optically compositing thin glazes of paint in their watercolor simulation[4]. Baxter also used this model in interactive painting system for oils or acrylics styles[5]. But these works were focused on the representation of western color pigments.

Lin simulate the oriental color ink diffusion effects based on the flows of water, ink, and pigment on paper. Chu also presents a physically-based method for simulating ink dispersion in absorbent paper based on the lattice Boltzmann equation[6,7]. However, these researches didn't consider the unique factors of oriental pigments. Since they can not reflect the optical effect of superimposed pigment layers and color composition method, we need to found the KM coefficients and the composition method which is suitable for oriental painting.

3. Artist's Real Process of Oriental Color-ink Painting

To render arbitrary 3D models in oriental color ink style automatically, it is important to study the characteristics that make oriental painting so unique.



(a) Silhouette (b) Volume (c) Color Diffusion

Fig. 1 Real oriental color-ink painting process

In the rest of this section, we'll review some of the most important characteristic expressions of oriental color ink painting from the artists' standpoint [8,9]. Fig.1 shows the general steps of the real oriental color-ink painting.

Silhouette Expression The silhouette expression of the oriental painting explains the object shape and the painter's emotion according to the strength and weakness of stroke.

Volume Expression In the traditional oriental color painting, artists put 3 colors on the different part of the brush at the same time and they express the volume and diffusion expression with this brush. These colors are applied to objects in the order from dark to light tone. Also they are painted from the center of object to outside in the oriental painting and there's some blank area very close to the outside This empty area is one of the unique factors in oriental style. Since such a coloring method is different from western painting, it is required to develop a novel Volume expression algorithm.

Color Diffusion Expression This is the irregular spreading effect that reveals the color ink diffusion over the oriental paper which is composed of fibers in random position and direction. In particular, the color diffusion effect also occurs among the adjacent color tone layers.

4. 3D Rendering Framework using Oriental Color-ink Model

Based on the observation on real artist's painting process in the previous section, we define the oriental color-ink model as consisting of 3 color layers: Base, Mid and Vivid. According to this color-ink model, we propose the novel rendering algorithm automatically generating oriental painting style for arbitrary 3D models. Using this framework, users can easily create the oriental scene by just giving 3D input models and choosing 3 colors for each object.

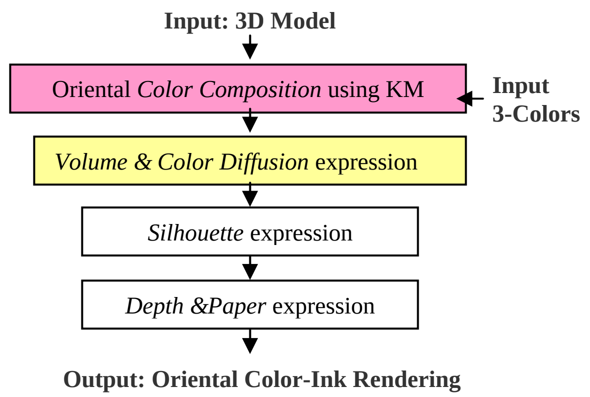


Fig. 2 Overview of our rendering framework

Fig. 2 shows the overview of our framework and Fig. 3 shows our color-ink model. We'll explain the main components in the following two subsections.

4.1. Oriental Color Composition using Kubelka-Munk(KM) Model

We use the KM compositing model for simulating the optical effect of superimposed 3-color layers and creating realistic Color Composition that reflects the characteristics of traditional oriental pigments. The KM model is a two-flux radiative transfer approach which has been suitable for describing reflectance properties of light scattering and absorbing materials. In particular, this model can display the color of mixed pigments realistically [10,11].

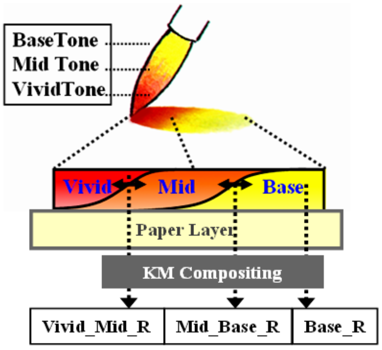


Fig. 3 Color-ink model and KM composition

Fig.4 shows the steps in our color composition using KM model. In brief, we begin with user selected two RGB colors(RGBw, RGBb), and stage 2 compute the absorption and scattering coefficients(S&K) and stage 3 uses this values to compute the reflectance and transmittance(R&T) of the each layers. Stage 4 composites the overall reflectance with the single layer's reflectance, and finally we can render the final pixel colors. We'll explain our algorithms for these steps in the following two subsections.

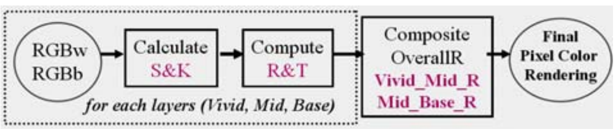


Fig. 5 Steps in Color Composition using KM model.

Image-based Acquisition of the Oriental Pigment Coefficients

Oriental pigment has unique colors on account of using natural materials and solutions. But in previous works using KM theory have only focused on the representation of western color pigments. In our method, we use the image-based acquisition method to reflect the optical parameters of oriental pigments to our KM model. In typical applications of KM theory, each pigment is assigned a set of absorption coefficients K and scattering coefficients S. Curtis proposed an interactive method to determine these coefficients choosing two RGB colors of the pigment over both a white and a black background [4].

In our application, we gain S & K from a true representation of real painting media. We create thick and flat samples of standard natural pigments that use common to artists. Our sample is painted on the Han-ji (oriental paper) as a white background and we used black ink as a black background like Fig.5. From these experimental set-up, we can acquire two colors, RGBw and RGBb respectively, then the K and S values can be computed by the equation (1) and (2).

Fig.5 Real oriental color pigment samples

$$S = \frac{1}{b} \times \coth^{-1}\left(\frac{b^2 - (a - RGB_w)(a-1)}{b(1 - RGB_w)}\right) \quad (1)$$

$$K = S(a-1) \quad (2)$$

where, $a = \frac{1}{2}\left(RGB_w + \frac{RGB_b - RGB_W + 1}{RGB_b}\right)$

$$b = \sqrt{a^2 - 1}$$

Overall Reflectance Computation using 3-Color Layers.

Given S and K for the pigmented 3 layers of given thickness x, the KM model allows us to compute reflectance R and transmittance T through the each layer by the equation (3). We can then determine the overall reflectance of two abutting layers with R1, R2 and T1, T2 respectively by the equation (4). We assume that user can select a desired thickness for each color layer.

$$R = \sinh bSx / c\,,\ T = b / c \quad (3)$$

where, $c = a \sinh bSx + b \cosh bSx$

$$OverallR = R_1 + \frac{T_1^2 R_2}{1 - R_1 R_2} \quad (4)$$

In oriental color-ink model, we have two pairs of adjacent pigmented color layers : *Vivid-Mid* and *Mid-Base*. So we need to compute two overall reflectances, *VividMid_R* and *MidBase_R*. To render final pixel color, we use the 3-KM reflectance values like Fig.3. We use the *Base* layer's R value as it is because it does not have a pigment composition.

4.2. Volume & Color Diffusion Expression

First, to represent the *Volume* expression using 3-KM reflectance colors, we compute the desired *tonelevel* of each vertex using diffuse lighting computation like Fig.6 . According to this value, we can decide the 4-tone levels for the KM reflectance colors from dark (VividMid_R) to light tone(Blank) like Fig.6. These colors are applied in the order from the center of object to the outside(silhouette) area following the real oriental volume expression which emphasize the center of the object as we described in the previous section.

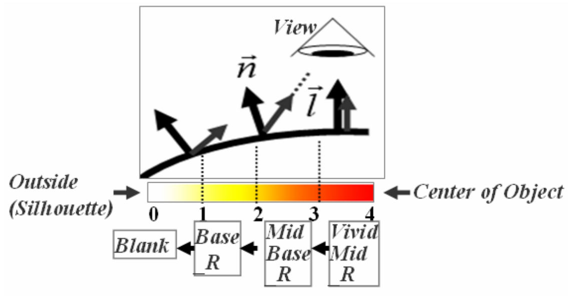


Fig. 6 Generation of *tonelevles*

Then to render final pixel color, we set the *tonelevels* from 0 to 4 and interpolate each pixel colors with this value. For example, if the *tonelevel* is 2.8, the final pixel color is decided from the following code. (We use the papercolor for the blank area close to the silhouette.)

```
  Final_Pixel_Color = Vivid_Mid_R * 0.8 + Mid_Base_R * 0.2;
```

Second, to simulate the *Color Diffusion* expression, we use the predefined diffusion *valuemap* that represent an irregular spreading pattern (that reveals the color ink diffusion in the oriental paper). This map is pre-rendered by Perlin noise function(like Fig.7 left). Otherwise, we also use the real ink-diffusion image (like Fig.7 right). At runtime, we create the color diffusion expression by blending the final pixel color using the diffusion *valuemap* as their blending rates. We can express the various styles using these predefined diffusion *valuemaps.*

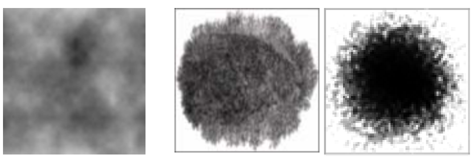


Fig.7 Diffusion valuemap samples

5. GPU implementation and experimental result

We achieve real-time rendering performance using GPU based implementation to calculate all the processes of our rendering algorithm. Proposed method is implemented using OpenGL and GLSL and we have tested on a 3.2Ghz, Intel 4P CPU with NVIDIA GeForce 6800 GPU. Our frame rates for various models are reported in Table1.

Table 1. Frame rates and sizes of representative models

Model	#of Faces	Fps
Cow	6088	80.7
Flower	10027	71.1
Game Character	11175	69.1
Korean Temple	212452	21.7

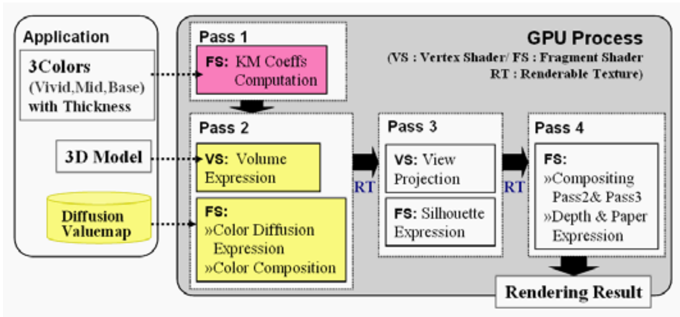


Fig. 8 Implementation architecture of oriental color-ink rendering

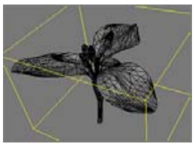
Our rendering framework on the GPU follows the stages of Fig. 8 and our pipeline has four GPU rendering passes. First, when user selects the 3-Colors and each thickness through the application, the fragment shader of pass 1 computes the KM values. Secondly, by importing a 3D input model, the vertex shader of pass 2 computes the *tonelevel* of each vertex, then the fragment shader (of pass 2) blends the multiple diffusion textures and composites the oriental KM color.

Then rendering result of pass 2 transforms into view projection space and we use this image as a reference for pass 3 via render-to-texture extension. In the pass 3, we use the adjusted *Sobel* mask for stylized silhouette expression. In the 3D space, it is required the detection of features and the silhouette of the input 3D model, but these usually take long computation times. So we use the image processing method. Our convolution filter is defined as Fig.9.

1	0	-1
2	0	-2
1	0	-1

Fig.9 Convolution filter for the Silhouette expression

In the pass 4, exponential fogging is used for atmospheric depth effect. Since recent graphics hardware has advantages about the per-pixel computation, most of our algorithms are implemented by the fragment shader.



(a) 3D mesh model

(b) Volume & Diffusion using KM Colors (c) Shilhouette (d) Depth & Paper

Fig.10 Oriental color-ink rendering process

Fig.10 ~ Fig.13 shows some of the experimental results using our oriental color-ink rendering algorithm. Fig.10 (a) shows the 3D input model and (b)~(c) show the rendering result of each rendering pass, (b) shows the KM color compositing result after tone & diffusion effect, (c) shows our silhouette rendering result and (d) shows the final result after the depth & paper effect.

(a) Diffusion value map (b) Purple flower (c) Detail image

Fig.11 Purple tone flower with minute diffusion texture

(a) Diffusion value map (b) Purple flower (c) Detail image

Fig.12 Orange tone flower with smooth diffusion texture

In the Fig.11 and Fig.12, we observe two different color tone variation and different diffusion texture style. For the Fig.11, we use the 3 purple-tone colors for KM compositing and apply minute diffusion texture. In the case of 12, we use the 3 orange-tone colors and apply more smooth diffusion texture.

Fig.13 Three example frames selected from continuous navigation, rendered in oriental style

Fig.13 shows our rendering result for a scene consisting of a 3D game character with a 3D temple model in front of the 2D background image, so that we can see how orientally stylized scene looks like in 3D games or in VR contents.

6. Conclusions and Further Remarks

Our method proposed an automated way of rendering arbitrary 3D models in oriental color ink painting style. Our novel framework of realtime GPU based oriental painting is different from other existing works in the aspect that this can automatically render 3D models in real-time and our color-ink model is established by reflecting the properties of real pigments and their way of blending. Since this framework is applied to 3D scene that can interactively changes, it is also suitable for real-time application such as animations, games, virtual environments.

As our future work, we can further extend color diffusion expression model using physically based simplified simulation in real time instead of using just a pre-computed diffusion textures.

References

1. Kang, S.J., Kim, C.H.: Real-Time 3D Sumi-e Painting. ACM SIGGRAPH Technical Sketch (2003)

2. Zhang, Q., Sato, Y.: Simple Cellular Automaton-based Simulation of Ink Behaviour and Its Application to Suibokuga-like 3D Rendering of Trees. The Journal of Visualization and Computer Animation (1999), 27-37

3. Way, D.L., Lin, Y.R.: The Synthesis of Trees in Chinese Landscape Painting using Silhouette and Texture Strokes. In Proceeding of Eurographics (2001), 123-131

4. Curtis, C.J., Anderson, S.E., Seims, J.E., Fleischer, K.W., Salesin, D.H.: Computer-Generated Watercolor. In Proceeding of SIGGRAPH (1997), 421-430

5. Baxter, W., Wendt, J., Lin, M.C. : IMPaSTo: A Realistic, Interactive Model for Paint. In Proceeding of NPAR(2004) 45-56

6. Lin, W. J., Shih, Z.C.: Computer-Generated Chinese Painting with Physically-Based Ink and Color Diffusion. In Proceeding of CGW (2004)

7. Chu, N.S: Moxi: Real-Time Ink Dispersion in Absorbent Paper. ACM Transactions on Graphics, Vol.24, No.3 (2005)

8. Sea, J.S: Oriental Color-ink Painting. Hyung-sul publishing co. (1992)

9. An, D.S: Traditional Oriental Painting. Sea-ye publishing co. (1991)

10. Haase, C.S (ed.): Modeling Pigmented Materials for Realistic Image Synthesis. ACM Transaction of Graphics (1992), Vol.11, No.4, 305-335

11. Kortum, G.: Reflectance Spectroscopy. Springer-Verlag (1969)