

基于柏林噪声的中国水墨画仿真算法

罗燕¹ 吴中福¹ 郭选昌² 周尚波¹

(重庆大学计算机学院 重庆 400044)¹ (重庆大学艺术学院 重庆 400044)²

摘要 为了实现中国水墨画的计算机仿真,基于对水墨画感性艺术效果的分析,提出了基于柏林噪声的中国水墨画仿真算法。该算法运用柏林噪声的基本理论,通过构造随机噪声函数、插值干扰函数等方法来实现水墨画的特征效果。仿真结果证明该算法较成功地模拟了中国水墨画的典型艺术效果。

关键词 非真实感绘制,中国水墨画,柏林噪声,仿真算法

Graphical Simulation Algorithm for Chinese Ink and Wash Painting Based on Berlin Noise

LUO Yan¹ WU Zhong-fu¹ GUO Xuan-chang² ZHOU Shang-bo¹

(Department of Computer, Chongqing University, Chongqing 400044, China)¹

(Department of Art, Chongqing University, Chongqing 400044, China)²

Abstract Based on analyzing the art effects of Chinese ink and wash drawing, a novel graphical simulation algorithm for Chinese ink and wash painting based on Perlin noise was proposed. We used the theory of Perlin noise, by constructing random noise functions, interpolation functions and different frequencies of the Perlin noise function to get the art effects of Chinese ink and wash painting. The experiment results indicate that the proposed algorithm can simulate the typical art effects of Chinese ink and wash painting successfully.

Keywords Non-photorealistic rendering, Chineses ink and wash painting, Perlin noise, Simulation algorithm

1 引言

自20世纪90年代中期开始,非真实感绘制(Non-Photorealistic Rendering, NPR)技术^[1]成为计算机图形学的研究热点,其主要内容是用计算机软硬件手段模拟具有不同绘画效果的画面和动画,中国水墨画的绘制效果就是该领域的一个研究分支。水墨画是中国画的一种,指纯用水墨所作之画,相传始于唐代,成于五代,盛于宋元,明清及近代以来续有发展。水墨画以笔法为主导,充分发挥墨法的功能。“墨即是色”,指墨的浓淡变化就是色的层次变化,“墨分五彩”,指色彩缤纷可以用多层次的水墨色度代替。长期以来水墨画在中国绘画史上占有重要的地位,是我国民族文化的瑰宝,它有着朴素抽象、富有暗示的画风,独特而鲜明的艺术特点,其最大的特色便是笔墨的运用与意境的表达。中国水墨画的基本要素有三个:单纯性、象征性、自然性。

中国水墨画通过长期的演变和发展,不仅形成了诸多风格派系,而且技法繁多,又因为艺术创作过程中偶然因素和感性因素的大量存在^[2],使得这种极为随意挥洒的艺术形式很难与逻辑严谨规范的计算机技术相融合,因此对中国水墨画进行仿真研究是计算机艺术领域的一个具有挑战性的课题。研究者应首先深入研究水墨画的特征,进而从繁多的技法中抽取体现中国水墨画特征的典型艺术效果,如浓淡、渗透、飞白等特效,并对感性的艺术效果进行图像分析,实现其数字化

描述,以便根据其特征选择合适的仿真手段来构建水墨画仿真模型体系^[3]。现在对中国水墨画的仿真研究方法基本上可分为两大类:基于物理建模的方法和面向艺术效果的仿真方法。

本文所研究的是采用面向艺术效果的仿真方法对水墨画进行模拟。将利用柏林噪声理论来模拟水墨画的效果,通过构建柏林噪声函数,并在一定范围内添加不同比例的柏林噪声函数,可以较真实地模拟水墨画效果。实验表明,所提算法较成功地实现了对水墨画的仿真,从而对中国水墨画的艺术效果仿真提供了一个有价值的新思路。

2 相关工作

近年来许多研究者就水墨画的计算机仿真方法进行了探讨,Strassmann^[4]提出了生成水墨画效果的毛笔模型;Lee^[5]等人提出了一个水墨与宣纸融合模型,由于他们所提的纸结构都以像素矩阵为基础,提出的所有扩散算法也都是以像素为单位进行的,使得扩散区域十分规则,扩散效果不真实,也不能模拟不同种类及不同纹理的宣纸。在渲染方面,Way^[6]等人通过绘制纹理笔画来模拟中国山水画,提出了一种计算机生成水墨画岩石纹理的方法,但这种方法较为繁琐,在水墨渲染时,用户很难找出合适的密度、节点分布、纹理长度、交角等参数组合,合成的效果缺少水墨韵味。石永鑫等人^[7]提出了基于粒子系统的水墨画绘制模型,借鉴粒子系统

到稿日期:2009-07-01 返修日期:2009-09-14

罗燕(1980-),女,博士生,主要研究方向为数字艺术等,E-mail:relax_luoyan@yahoo.com.cn;吴中福 教授,博士生导师;郭选昌 教授,博士生导师;周尚波 教授,博士生导师。

的基本概念,使用“伪布朗运动”作为水墨粒子流动的推动形式,对给定的输入笔迹进行边缘提取和检出,通过层次对象混合运算来实现基本的多笔次叠加效果,在一定程度上模拟了水墨画的部分典型效果,但该模型的算法实现起来有一定的难度,且系统的输入参数缺少灵活性,在单笔仿真方面不能有效地模拟水墨形态的自然分布效果,缺乏真实感效果。

本文针对以上情况,提出了一种基于柏林噪声函数的水墨画仿真算法,将柏林噪声理论应用于水墨画仿真中,通过构造随机噪声函数和插值函数来实现水墨画效果,并试验了实现水墨画的实时动态特性。

3 基于柏林噪声的仿真算法

水墨画的水墨主要是在纸的内部进行扩散和渗透的,因此水墨的运动不单单靠重力作用,更主要是纸纤维的吸水作用和墨粒子的粘附作用,进而产生扩散纹理^[8]。而这些效果的显现除了与墨的浓度(即颜色的深浅)有关之外,还与纸纤维的吸水性、用水量多少、用笔的角度以及纸纹理的情况有关,有时甚至完全是随机出现的效果,且水墨的形态是由不同变化级别的图案组合而成的。

我们发现可以通过构建柏林噪声函数来模拟绘制水墨的扩散渗透、浓淡、飞白等艺术效果。本文利用在一定范围内添加不同频率的噪声函数来实现一些特殊的水墨效果。先对柏林噪声函数进行计算并将其存入柏林噪声函数值分布表,为加快计算速度,不需每次对柏林噪声进行计算,只需一次性计算出所需已知范围内的分布。再在水墨模拟程序中加入柏林噪声函数,以影响水墨的浓淡、渗透、飞白等效果。

3.1 柏林噪声的基本原理

柏林噪声是由 Berlin 在 20 世纪 80 年代提出的^[9,10],它可以在某一空间生成连续的噪声。噪声函数是一系列随机噪音叠加起来而产生的一个连续函数,已经被成功应用于大理石、火焰云和云彩的模拟,本文运用柏林噪声模拟水墨画的效果是一个新的尝试。

3.2 构建柏林噪声函数

3.2.1 噪声函数

一个噪声函数从本质来说就是一个有种子的随机数生成器。它有一个参数,并基于这个参数返回一个随机数。如果用同一个参数输入两次,则噪声函数两次产生相同的返回值,这一特性非常重要,便于控制噪声函数值域。

图 1 是由一个噪声函数生成的随机值,X 轴上的每个点都对应于一个介于 0 和 1 之间的随机数。

图 2 是通过在这些随机数之间平滑插值后的效果图,给出了一个非整型的连续函数参数。接下来讨论各种不同的插值方法。

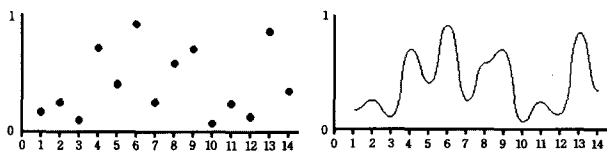


图 1 噪声的离散点

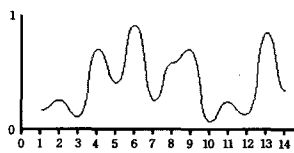


图 2 连续平滑图

3.2.2 构建 Perlin 噪声函数

假设有以下光滑连续的函数,它们各自的频率和振幅不同,把所有这些函数叠加起来时,就得到了一个噪声函数,即

perlin 噪声函数,如图 3 所示。

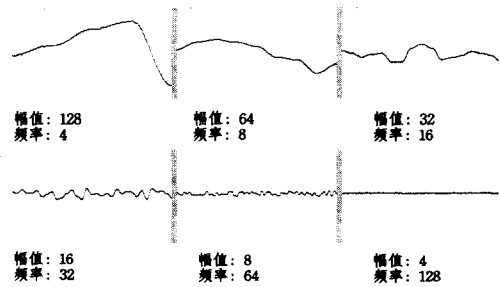


图 3 6 个不同的噪声函数

图 3 展示不同的曲线变化级别。把以上 6 个函数叠加,可得到图 4 效果,甚至可以把这些图形想象成一系列山脉的轮廓。很多绘图应用程序就是运用二维噪声函数来生成山水风景的。以下讨论二维空间的噪声函数,图 5 所示则是在二维空间中创建不同频率的噪声函数。

噪声函数之和=柏林噪声



图 4 6 个噪声函数叠加后的效果图

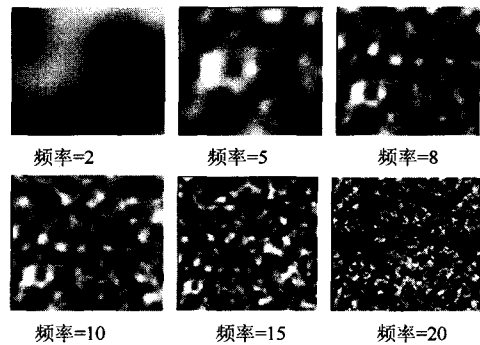


图 5 不同频率的二维柏林噪声

3.2.3 持续性参数(persistence)

噪声函数在叠加的时候,需要确定各个噪声函数的振幅和频率,以实现理想叠加效果。而在一维例子中的一系列被叠加的图案之间,采用了频率加倍、振幅减半的方法。当然,也可以在不同的噪声函数之间构建不同的频率和振幅变化幅度来达到这种效果。举例来说,如果要构建一个光滑起伏的山丘,可把振幅大、频率低的噪声函数和振幅很小、频率很高的噪声函数相叠加。如果想构建相对平坦但是多岩石的山丘轮廓,则可以使用较小的振幅和较低的频率相叠加的方法。

本文用持续性参数值来指示每个频率下的振幅,根据 Mandelbrot 的原理知道,持续性参数是指有很多高频信号的噪声持续性参数,数值较低。而本文对持续性参数的定义如下:

$$\text{频率} = 2^i, \text{幅值} = \text{persistence}^i$$

这里的 i 是叠加中用到的第 i 个噪声函数,图 5 展示了不同的持续性参数对 Perlin 噪声产生的输出影响。每个 Perlin 函数都是由 6 个源噪声函数叠加而来。从图 5 中能够总结出随持续性参数的变化,Perlin 函数产生的变化规律。

从图 6 可以看出,不同的持续性参数对 Perlin 噪声输出产生的影响是不同的,利用不同的持续性参数可以实现不同

的绘制效果。接下来介绍 Perlin 噪声函数的重要概念 Octave。

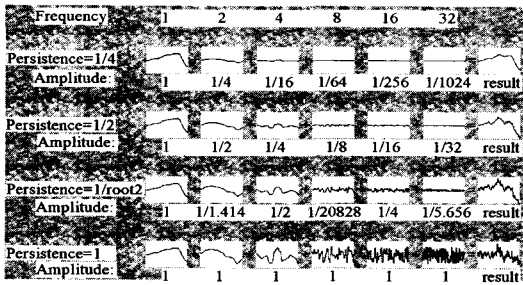


图 6 不同持续性参数对 Perlin 噪声输出的影响

3.2.4 Octave

Octave 指的是一系列被叠加的噪声函数,每个噪声函数都是由前一个噪声函数的频率加倍所得到的,而在音频中,Octaves 也有着相同的特性。用 Perlin 噪声函数在计算机屏幕上显示图像时,像素点是最基本的单位,所以当 Octaves 的频率高于像素点频率时,图像将变得不可见,屏幕像素不足以显示这些高频噪声函数的微小细节。因此,什么振幅能够达到足够小,取决于持续性参数的数位、Perlin 函数的振幅和屏幕的分辨率等因素。接下来将介绍如何从产生一个噪声函数到对这些离散噪声点进行插值和做平滑化处理,再合成一个有效的柏林噪声函数的具体过程。

3.2.5 构造噪声函数

噪声函数从本质上来说就是一个随机数,但是它并不像通常程序中用到的其他随机数,每次调用时都生成一个不同的随机数。这个噪声函数产生的随机数是由一个或多个参数计算生成的。当每次输入相同的参数时,它产生相同的返回值,而当输入的参数不同时,它产生的返回值是不同的随机数。

通过以上分析就清楚知道需要怎样的随机数生成器,接下来就是去寻找产生这种随机数的函数。下面的代码就是一个符合要求的函数,它的返回值是介于-0.1 和 1.0 之间的浮点数。

```
function IntNoise(32-bit integer: x)
    x = (x << 13) ^ x;
    return (1.0 - ((x * (x * x * 15731 + 789221) +
    1376312589) & 7fffffff) / 1073741824.0);
end IntNoise function
```

3.2.6 构建插值函数

在构造噪声函数之后,需要把它的输出变得连续且平滑,需要对离散的噪声函数进行插值处理。而一个标准的插值函数有 3 个参数: x, y, z , 其中 x 和 y 是插值函数的两个端点,且 $z \in [0, 1]$, 插值函数根据 z 的值,返回一个数 n 且 $n \in [x, y]$ 。接下来介绍几个插值方法。

(1) 线性插值法

```
function Linear_Interpolate(x, y, z) //线性插值法
{
    return x * (1 - z) + y * z
} end of function
```

图 7 是线性插值后的效果,看起来不太好,曲线过渡幅度比较大,达不到平滑过渡效果。不过可以用这种方法来处理物体的轮廓图,简单快速。



图 7 线性插值法

(2) Cosine 插值法

```
function Cosine_Interpolate(x, y, z) //Cosine 插值法
{
    ft = z * 3.1415927
    f = (1 - cos(ft)) * 0.5
    return x * (1 - f) + y * f
}end of function
```

图 8 是用 Cosine 插值法得到的曲线,比用线性插值法得到的效果更加明显,曲线较光滑,过渡较自然,但是处理效率低。



图 8 Cosine 插值法

于是本文将结合线性插值法和 Cosine 插值法的优点,构建组合插值法来产生水墨画的艺术效果图。图 9 所示是采用不同频率和振幅的 Perlin 噪声函数及使用不同插值平滑处理后的渲染效果图。

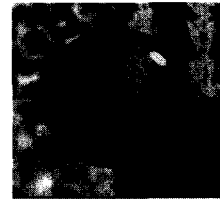


图 9 基于柏林噪声的四种不同二维噪声效果图

以上分析了多种噪声函数和插值方法,将其整合起来,构建了一个新的 Perlin 噪声函数。这个过程事实上是几个插值噪声函数的叠加。而 Perlin 噪声只是一个函数,当输入一个或多个参数时,就产生一个值。以下便是一个简单的 Perlin 函数。

Perlin 函数的主体是一个循环结构,每个循环体是将各个 Octave 叠加,然后频率加倍。各个循环体都调用上次的噪声函数,把第 i 个 Octave 用到的噪声函数记为 $noise_i$ 。实际上不必为每个 Octave 都产生噪声函数,因为所有的噪声函数本质上都是相同的,且用到了 3 个很大的质数。当然可以在 Perlin 函数体内重用相同的一段代码,或者改变一下 $noise_i$ 中的质数,以达到模拟绘画要求。

一维 Perlin 函数代码如下:

```
function Noise1(integer x) //噪声函数
    x = (x << 13) ^ x;
    return (1.0 - ((x * (x * x * 15731 + 789221) +
    1376312589) & 7fffffff) / 1073741824.0);
end function
function SmoothedNoise_1(float x) //噪声平滑函数
```

```

return Noise(x)/2 + Noise(x-1)/4 + Noise(x+1)/4
end function
function InterpolatedNoise_1(float x) //噪声插值
integer_X = int(x)
fractional_X = x-integer_X
v1 = SmoothedNoise1(integer_X)
v2 = SmoothedNoise1(integer_X + 1)
return Interpolate(v1,v2,fractional_X)
end function
function PerlinNoise_1D(float x)//柏林噪声函数定义
total = 0
p = persistence
n = Number_Of_Octaves-1
loop i from 0 to n
frequency = 2i
amplitude = pi
total = total + InterpolatedNoise1(x * frequency) * amplitude
end of i loop
return total
end function

```

3.3 墨竹仿真算法

竹子是我国绘画史上的传统题材,当纯以墨笔画竹,称之为墨竹,墨竹画在我国绘画史上已有上千年历史,在中国水墨画题材中占有非常重要的地位。将文章《基于分形算法的仿真“竹”的实现》中所得的三维竹模型信息、纹理信息和上述的柏林噪声原理结合起来处理,可得到最终三维墨竹画的效果,若通过 GPU 实现,则该算法可达到实时运算的速度。

在下面的算法中,先将柏林噪声函数计算和存储为柏林噪声函数值分布表。这是为了加快处理速度,不需每次对柏林噪声进行计算,只需一次性计算出所需已知值范围内的分布。以下是水墨模拟的 Cgfx(Nvidia 公司提供的一种在图形处理器上编译运行的开发语言,主要为用户进行实时图形图像开发)程序代码。

```

vertexOutput std_VS(appdata IN){ //完成模型在坐标空间的变换
vertexOutput OUT = (vertexOutput)0;
OUT. WorldNormal=mul(WorldITXf,IN. Normal). xyz;
OUT. WorldTangent=mul(WorldITXf,IN. Tangent). xyz;
OUT. WorldBinormal=mul(WorldITXf,IN. Binormal). xyz;
float4 Po = float4(IN. Position. xyz,1);
float3 Pw = mul(WorldXf,Po). xyz;
OUT. LightVec = (Lamp0Pos-Pw);
#ifdef FLIP_TEXTURE_Y
OUT. UV = float2(IN. UV. x,(1.0-IN. UV. y));
#else /* ! FLIP_TEXTURE_Y */
OUT. UV = IN. UV. xy;
#endif /* ! FLIP_TEXTURE_Y */
#ifdef USE_SHARED_SHADOW
float4 Pl = mul(ShadowViewProjXf,Pw);// "P"
in light coords
float4x4 BiasXf = make_bias_mat(ShadBias);
OUT. LProj = mul(BiasXf,Pl);//bias to make texcoord
#endif /* USE_SHARED_SHADOW */
OUT. WorldView=normalize(float3(ViewIXf[0]. w,ViewIXf[1]. w,ViewIXf[2]. w)-Pw);

```

```

OUT. HPosition = mul(WvpXf,Po);
return OUT;
}
PIXEL SHADING
float4 std_PS(vertexOutput IN):COLOR //计算每一个像素的效果
float3 Ln = normalize(IN. LightVec);//对灯光向量进行归一化
float3 Nn = normalize(IN. WorldNormal);//对当前像素的法向量进行归一化
float ldn = dot(Ln,Nn); //计算内积
ldn = max(ldn,0.0); //取最大值
float2 duv=IN. UV. xy; //得到纹理坐标信息
duv. y* =repeatV; //重复计算纹理
float3 diffuseColor =tex2D(ColorSampler,duv). rgb;
//得到表面纹理颜色
float3 result = diffuseColor * (ldn * Lamp0Color + AmbiColor); //计算表面颜色
float u=(result. x+result. y+result. z) * 0.33; //取出水平纹理坐标信息
float2 uv=float2(u,0.5);
float4 color=tex2D(MapSampler,uv);//根据水平纹理坐标查询预处理函数纹理
// return as float4
return color;//float4(result,1); //返回最终颜色
}

```

在上述水墨画模拟的程序后加入柏林噪声函数,从而影响水墨画的浓淡、渗透、飞白等效果的线性分布。图 10 为不同等级参数得到的函数预处理存储图。

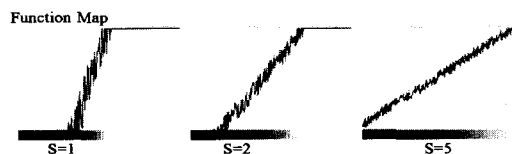


图 10 线性插值生成函数

3.4 仿真实验结果及分析

在仿真实验中通过构建多个柏林噪声函数来对墨竹画进行艺术风格的非真实感绘制,验证了本文算法的可行性和有效性。而各种水墨效果的运用则是一个随时变化的动态叠加和融合过程。图 11 是本文模拟出的墨竹画动态艺术特征图,它体现出了墨色的变化以及墨法的艺术效果。该图像在 GeForce 7900,显卡为 128M,软件平台为 MAYA 2009 的实验环境下能获得每秒 20 帧的实时渲染速度,表明了所提算法的实时水墨动态特性,体现出了墨竹画的浓淡、渗透和飞白等艺术特征,能很好地表现出水墨画水晕墨染的韵味。



图 11 运用本算法于三维竹模型的实时水墨效果图

结束语 本文首先对中国水墨画的基本艺术特征进行了概括和总结;然后分析了当前研究成果的优缺点,针对当前水墨画仿真模型的问题,提出了一个基于柏林噪声的水墨画仿真算法。最后通过构建柏林随机噪声函数来模拟水墨画形态

(下转第 296 页)

exchange based on feature semantics and geometry representation[C]//Proceedings of the CSCWD, Nanjing, 2006; 1-5

- [8] Guarino N, Welty C A. A formal ontology of properties[C]//Proceedings of the 12th European Workshop on Knowledge Acquisition, Modeling and Management. 2000; 97-112
- [9] Libkin L, Fagin D R, Kolaitis P, et al. Data exchange; semantics

and query answering [J]. Theoretical Computer Science, 2005, 336(1): 89-124

- [10] Libkin L. Data exchange and incomplete information [C]//Proceedings of the Twenty-fifth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems. Chicago, IL, USA, 2006; 60-69

(上接第 255 页)

- [4] Compton K, Hauck S. Reconfigurable computing; A survey systems and software[J]. ACM Computing Surveys, 2002, 34(2): 71-210
- [5] 汪斌强. 可重构路由器构件组研制(国家高技术研究发展计划项目课题申请书)[A]. 2007
- [6] Montz A B, Mosberger D, O'Malley S W, et al. Scout: A communications-oriented operating system [C]//Proc. of the 5th Workshop on Hot Topics in Operating Systems (HotOS-V). 1995; 58-61. http://lahtermaher.org/pub/plan/xkernel/Papers/scout_hotos.ps
- [7] Kohler E. The Click modular router [D]. Department of Electrical Engineering and Computer Science, MIT University, 2001
- [8] Decasper D, Dittia Z, Parulkar G, et al. Router plugins: A software architecture for next generation routers[J]. IEEE/ACM Trans. on Networking, 2000, 8(1): 2-15
- [9] Lee D C, Midkiff S F. Reconfigurable Routers; a New Paradigm for Switching Device Architecture[OL]. <http://www.ccm.ece.vt.edu/papers/>, 1998

- [10] Lee D C, Harper S J, Athanas P M, et al. A stream-based reconfigurable router prototype [A]//Proceedings of the IEEE International Conference on Communications [C]. Vancouver, B. C., Jun 1999
- [11] Albrecht C, Foag J, Koch R, et al. DynaCORE-a dynamically reconfigurable coprocessor architecture for network processors [A]//Proc. of the 14th Euromicro Conference on Parallel, Distributed and Network-based Processing (PDP 2006) [C]. IEEE Computer Society, Montbeliard-Sochoux, France, 2006; 101-108
- [12] Houidi I, Louati W, Zeghlache D. An extensible software router data-path for dynamic low-level service deployment[C]//IEEE Workshop on High Performance Switching and Routing 2006. Poland, June 2006; 161-166
- [13] Reussner R H, Becker S, Firus V. Component composition with parametric contracts[C]//Proceedings of the Net. Object Days, 2004; 155-169
- [14] Bracciali A, Brogi A, Canal C. A formal approach to component adaptation[J]. Journal of Systems and Software, 2005, 74(1): 45-54

(上接第 281 页)

的自然分布,以实现中国水墨画的典型艺术效果。结果说明,本文算法比较真实地模拟了中国水墨画的部分典型特征,使得墨竹模拟效果比较逼真,能够较好地实现模拟要求。同时该算法还存在不完善的地方,如尚未充分分析柏林噪声函数的倍频和振幅的变化对水墨扩散效果产生的影响,这有待进一步研究。

参 考 文 献

- [1] Lansdown J, Schofield S. Expressive rendering; A review of non-photorealistic techniques [A]//IEEE Computer Graphics and Application[C]. 1995; 29-37
- [2] 俞剑华. 中国画论类编(上卷)[M]. 北京:人民美术出版社,1986
- [3] 齐亚峰,孙济洲,商毅. 中国水墨画的基本艺术特征及其计算机仿真实现[J]. 中国图像图形学报, 2003, 5(8): 562-5
- [4] Strassmann S. Hairy brushes [A]//Computer Graphics Proceedings, Annual Conference Series[C]. ACM SIGGRAPH, D-

allas, 1986; 225-232

- [5] Lee J. Difusion rendering of black ink paintings using new paper and ink models[J]. Computers and Graphics, 2001, 25(2): 295-308
- [6] Way D L, Shih Z C. The synthesis of rock textures in Chinese landscape painting [A]//Proceedings of Eurographics' 2001 [C]. Manchester, 2001; 123-131
- [7] 石永鑫,孙济洲,张海江,等. 基于粒子系统的中国水墨画仿真算法[J]. 计算机辅助设计与图形学报, 2003, 6(15): 667-672
- [8] 焦景山,孙济洲. 中国水墨画效果的图形仿真[A]//见辜居一,主编. 数字化艺术论坛[M]. 杭州:浙江人民美术出版社, 2002; 190-204
- [9] 谭彬. 组物外观模拟的研究与设计[D]. 杭州:浙江大学, 2004
- [10] Perlin K. An Image Syntlesizer [C]//SIGGRAPH' 85. 1985; 287-296
- [11] 杨刚. 全景图拼接算法的设计与实现[J]. 重庆工学院学报:自然科学版, 2007, 21(19): 107-110

(上接第 284 页)

- [3] Ridd M K, Liu J. A comparison of four algorithms for change detection in an urban environment [J]. Remote Sens. Environ., 1998, 63(2): 95-100
- [4] Byrne G F, Crapper P F, Mayo K K. Monitoring land cover change by principle component analysis of multitemporal Landsat data [J]. Remote Sens. Environ., 1980(10): 175-184
- [5] 黎夏,叶嘉安. 利用主成分分析改善土地利用变化的遥感监测精度[J]. 遥感学报, 1997, 1(4): 282-289
- [6] 张路. 基于多元统计分析的遥感影像变化检测方法研究[D]. 武汉:武汉大学, 2004
- [7] 范海生,马蔼乃,李京. 采用图像差值法提取土地利用变化信息

方法——以攀枝花仁和区为例[J]. 遥感学报, 2001, 5(1): 75-80

- [8] Wang G. An Expert System for Forest Resource Inventory and Monitoring in the Frame of Multi-Source Data [M]. Helsinki, Finland: Helsinki Univ. Press, 1996
- [9] Lillesand T M, Kiefer R W. Remote Sensing and Image Interpretation [M]. New York: Wiley, 2000
- [10] Eklundh L, Singh A. A comparative analysis of standardized and unstandardized principal components analysis in remote sensing [J]. International Journal of Remote Sensing, 1993, 14(7): 1359-1370
- [11] 盛辉,廖明生,张路. 基于典型相关分析的变化检测中变化阈值的确定[J]. 遥感学报, 2004, 8(5): 451-457
- [12] 佘袁勇. 基于遥感影像的变化检测研究 [D]. 武汉:武汉大学, 2005