

一种基于多分辨率和统计的笔刷绘制方法

赵芳芳, 秦茂玲, 任石

(山东师范大学信息科学与工程学院, 济南 250014)

摘要: 采用高斯图像金字塔生成多幅参考图像, 使用不同大小的笔刷进行多层绘制, 模拟画家由粗到细的创作过程, 设计一种新的笔刷绘制方法。在每层绘制时, 使用统计方法确定笔刷的颜色, 运用像素 8-邻域的概念控制笔刷的走势。实验结果表明, 通过多个参数的不同设置可以将一幅照片渲染成具有一定风格的艺术图像。

关键词: 非真实感绘制; 统计; 笔刷模型; 高斯图像金字塔; 8-邻域

Brush Stroke Rendering Method Based on Multi-resolution and Statistics

ZHAO Fang-fang, QIN Mao-ling, REN Shi

(College of Information Science and Engineering, Shandong Normal University, Jinan 250014)

【Abstract】 This paper designs a new brush rendering method, through adopting a Gaussian image pyramid to generate a series of reference images, using different brushes to paint layer by layer in order to simulate the rendering stages from coarser to finer. This method uses statistics method to determine the brush's color and pixels' eight-neighbor concept to control the direction of brush strokes. Experimental result shows that a photo can be transformed to aesthetic images by setting up different parameters.

【Key words】 Non-Photorealistic Rendering(NPR); statistics; brush stroke model; Gaussian image pyramid; eight-neighbor

1 概述

非真实感绘制(Non-photorealistic Rendering, NPR)于 20 世纪 90 年代中期逐渐成为计算机图形学研究的热点, 它是计算机技术和绘画艺术相结合的一个研究领域。由于真实的绘画是由笔刷完成, 因此基于笔刷的绘制(Stroke-based Rendering, SBR)成为绘画模拟的核心。文献[1]提出使用不同大小和不同形状的笔刷对静态图像进行多层绘制的油画模拟算法, 并进一步提出基于能量最小化的方法改进原有绘制算法。文献[2]将笔刷看作一个矩形框, 并用不同的笔刷近似表现在图像中每个目标对象, 通过分层绘制的方法达到 NPR 的绘制效果。文献[3]注重设计不同的笔刷样式, 提出较为完备的基于不同笔刷模型的图像绘制方法。

多分辨率技术绘制的系统和过程符合人类视觉和思维方式。用户根据分辨率的不同设置画笔的大小、形状等, 模拟画家真实的绘画过程。大多数基于笔刷的绘制方法是以某点颜色代替包含该点在内的小区域颜色, 即以一个点代替整体。如此得到的图像颜色过渡明显, 计算量也增大。本文提出的基于多分辨率和统计的笔刷绘制方法得到丰富的笔刷颜色, 获得了较好的模拟效果。

2 基于多分辨率和统计的笔刷绘制方法

2.1 参考图像的产生

多分辨率技术应用于图像分析得到图像金字塔。最常用的金字塔模型是高斯金字塔(GP), 它由图像数据递归分解得到的多分辨率图像表示。

高斯金字塔的产生过程如下^[4]: 假设在初始条件下, 图像由 g_0 表示, 含有 C 行和 R 列像素, 每个像素表示像平面对应点的灰度值, 此图像构成高斯金字塔的第 0 层。在金字

塔的第 1 层中包含图像 g_1 , 它是 g_0 的低通滤波(或压缩)版本。在第 1 层中的每个像素值是第 0 层对应像素和一个 $M \times N$ 矩阵窗进行加权平均得到的值。在第 2 层中包含的图像 g_2 , 它的每个像素值是对第 1 层中的对应像素做相同加权平均所得。

层到层的加权平均过程如下式所示:

$$g_l = \text{REDUCE}(g_{l-1}) \quad (1)$$

式(1)表示对某一层 l , $0 < l < N$, 若它的节点为 (i, j) , 并满足 $0 < i < C_l$, $0 < j < R_l$, 则存在以下表达式:

$$g_l(i, j) = \sum_{m=-2}^2 \sum_{n=-2}^2 w(m, n) g_{l-1}(2i+m, 2j+n) \quad (2)$$

其中, N 为金字塔的总层数; C_l 和 R_l 分别为金字塔第 l 层的行维数和列维数; $w(m, n)$ 为加权模板。

在逐层绘制时, 本方法为各层提供一幅参考图像, 参考图像代表当前大小的笔刷需近似的图像。本文采用高斯图像金字塔算法产生 3 幅包含不同细节信息的参考图像, 如图 1 所示。



(a) 金字塔第 2 层 (b) 金字塔第 1 层 (c) 金字塔第 0 层(原图像)

图 1 采用高斯图像金字塔算法生成的参考图像

作者简介: 赵芳芳(1982-), 女, 硕士研究生, 主研方向: 非真实感绘制, 多分辨率技术; 秦茂玲, 副教授; 任石, 硕士研究生

收稿日期: 2009-01-05 **E-mail:** zfflq@163.com

输入原图像作为高斯金字塔的第 0 层,用 g_0 表示。根据式(2)对 g_0 进行 5×5 低通滤波得到金字塔的第 1 层,用 g_1 表示,即 g_1 中的每个像素值是对 g_0 中对应像素与一个 5×5 的矩形窗进行加权平均后得到的值。对 g_1 进行同样的操作得到金字塔的第 2 层,用 g_2 表示,即 g_2 中的每 1 个像素的值是对 g_1 中对应像素做同样的加权平均得到的。对于彩色图像,要对 RGB 3 个通道分别进行计算。其中,使用的 5×5 标准高斯模板矩阵为

$$W_{3,5} = \begin{bmatrix} 1 & 1 & 2 & 1 & 1 \\ 1 & 2 & 4 & 2 & 1 \\ 2 & 4 & 8 & 4 & 2 \\ 1 & 2 & 4 & 2 & 1 \\ 1 & 1 & 2 & 1 & 1 \end{bmatrix}$$

2.2 笔刷颜色的设置

在大多数基于笔刷的绘制技术中,有一个共同的特点是:以某点的颜色代替包含该点在内的小区域颜色,模拟画家在绘画过程中笔刷的一次落笔。这需要确定小区域中的一点来代替整体。确定这个点后,要求所求区域内的点的颜色与该点的颜色差在某个阈值内。本文使用统计方法确定笔刷颜色,该方法不是使用在区域内某点的颜色代替在整个区域内所有点的颜色,而是统计在该区域内哪种颜色的像素点数最多,以此颜色来代替整个区域内像素的颜色形成笔刷落笔效果。

本方法确定笔刷颜色的具体过程如下:定义 4 个数组 gray_count[256]记录每一级灰度(数组下标)像素的点数;red[256], green[256], blue[256]记录每一级灰度(数组下标)所有像素的红、绿、蓝分量之和。

Step1 逐行扫描原图像,假定当前要求解点 $C(x,y)$ 的颜色值,在原图像中考察以 C 为中心的区域(2.4 节的算法给出具体区域大小)内的每一点,设当前考察 S 点,其红绿蓝分量为 (r,g,b) ,求 S 点的颜色灰度值 $gray=0.299r+0.587g+0.114b$,使 $gray_count[gray]=gray_count[gray]+1$, $red[gray]=red[gray]+r$, $green[gray]=green[gray]+g$, $blue[gray]=blue[gray]+b$ 。这样 gray_count[256]记录该区域内每个灰度级的像素点数。red[256], green[256], blue[256]记录该区域内每一级灰度的所有像素的红绿蓝分量之和。

Step2 在 gray_count[256]中找到最大值 max, $max=gray_count[Nmax]$,即区域中灰度级为 $Nmax$ 的像素点数最多为 max。设 $r_new=red[Nmax]/max$, $g_new=green[Nmax]/max$, $b_new=blue[Nmax]/max$,其中, r_new, g_new, b_new 是所求点的 3 个颜色分量。

2.3 笔刷的产生

笔刷具有一定大小、颜色、长度和形状。本方法采用如下数据结构进行描述:

```
struct brush
{
    RGBstruct    Color;        //笔刷颜色
    integer      R;            //笔刷大小
    integer      Brushlength;  //笔刷长度
    pixels       Startingpoint; //笔刷的起点
    pixels*      Strokelist;   //骨架点列表
}
```

其中,笔刷颜色通过统计对应参考图像中笔刷起点所在区域的颜色值确定;笔刷的骨架点决定笔刷的方向。为了产生笔刷的骨架点,在各层绘制前,将当前参考图像转换成亮

度图像。笔刷形状的生成如图 2 所示。

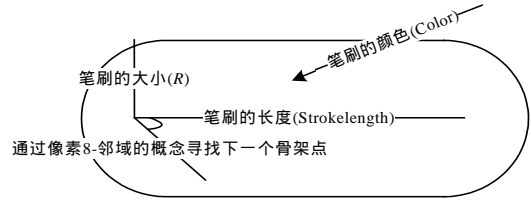


图 2 笔刷形状生成

形成一笔的具体算法如下:

Step1 从笔刷起点 p_0 开始,调用 2.2 节的算法得 p_0 处笔刷的颜色,并将 p_0 点添到 Strokelist。

Step2 运用像素 8-邻域的概念从当前骨架点 p_i 的“米”字形的 8 个方向中选择一个与之亮度差异最小的像素 p_{i+1} 作为下一个骨架点,亮度差异定义为在像素间亮度的欧式距离, $\|L_{p_{i+1}}-L_{p_i}\| = \sqrt{(L_{p_{i+1}}-L_{p_i})^2}$,其中, $L_{p_{i+1}}, L_{p_i}$ 分别表示在点 p_{i+1}, p_i 处的亮度值。设标志位 flag=1 表示 p_i 已被绘制, i 从 1 到 Brushlength 依次循环。

Step3 如果当前参考图像该点的颜色值与在画布中该点的颜色值之差小于参考图像与当前画笔颜色 Color 之差,那么该笔绘制于该点结束。

2.4 基于多分辨率和统计的笔刷绘制

基于图像金字塔绘制的实质是逐层将一系列笔刷结合在图像结构中,自动生成非真实感图像的方法。

2.1 节采用高斯金字塔生成 3 幅参考图像,2.3 节设置笔刷的各属性。本节设置一个阈值参数(T)控制绘制的抽象程度,根据参考图像的不同设置统计时圆半径 R 。笔刷不绘制所有与参考图像不匹配的区域(ErrorArea),绘制与当前参考图像相差在一定阈值(T)内的区域,若阈值越大,则绘制结果越抽象;若笔刷半径越大,则所需的笔刷数量越少,绘制结果也越抽象。

按参考图像分辨率由低到高的顺序进行多层绘制,每层绘制要一个与当前参考图像大小一致的画布,使它与当前参考图像分辨率一致。因此,在绘制的过程中,要转换画布。本方法将在当前画布中的一个颜色值对应下层画布中 4 个像素的颜色值进行画布转换: $C_{k-1}(2i-1,2j-1) = C_{k-1}(2i-1,2j) = C_{k-1}(2i,2j-1) = C_{k-1}(2i,2j) = C_k(i,j)$ 。

每层的绘制过程相似,绘制一层的具体算法如下:

定义 C_k 为当前的画布, I_k 为当前参考图像, R 为当前笔刷大小, T 为阈值。

Step1 逐行扫描画布 C_k ,每隔 3 个像素产生一个像素点 P 。对于每个点 $P(x,y)$,寻找以 $(x-R/2,y-R/2)$ 为起点、 R 为宽和高的区域 A 。

Step2 对于在区域 A 内的每一点 S ,计算它与当前参考图像 I_k 对应点的颜色差值,确定不匹配的区域 ErrorArea。

Step3 如果 $ErrorArea > T$,那么选择不匹配区域 ErrorArea 中颜色差值最大的点 p_i ,作为笔刷起点。

Step4 利用在 2.3 节中设置好的笔刷绘制点 p_i 。依此循环,直至找到该层所有要绘制的点,完成绘制。

3 绘制结果

参考图 1,设置笔刷的半径分别为 $R=8, R=4, R=2$ 模拟画

(下转第 251 页)