

基于样本的纹理合成技术综述

朱文浩 魏宝刚

(浙江大学计算机科学与技术学院, 杭州 310027)

摘要 近年来,纹理合成技术取得了长足的进步,已成为当前计算机图形学、计算机视觉和图像处理领域的研究热点之一。为使人们对该技术有个概略了解,首先针对纹理合成的研究现状,从合成采样、特征分析与匹配、约束合成和多目标合成等4个方面全面论述了纹理合成目前采用的技术方法;然后从合成质量、合成效率等角度对典型的纹理合成算法进行了分析,并指出了它们的长处和存在的不足之处;最后探讨了纹理合成技术今后的发展方向。

关键词 纹理合成 点匹配 块拼接 约束

中图法分类号: TP391.41 **文献标识码**: A **文章编号**: 1006-8961(2008)11-2063-07

The Technology of Sampled-based Texture Synthesis

ZHU Wen-hao, WEI Bao-gang

(College of Computer Science and Technology, Zhejiang University, Hangzhou 310027)

Abstract Texture synthesis has been well studied in recent years. It has become an active research topic of computer graphics, vision, and image processing. Aiming at the current progress of texture synthesis, this work we set forth the algorithms based on synthesis sampling, character matching, constrained synthesis and multi-objects synthesis. Furthermore, we analyze typical algorithms' advantage and the disadvantage on the point of synthesis quality and efficiency. At last, we discuss the orientation of texture synthesis and emphasize that the constrained synthesis techniques would speed up the study development and become the key point of synthesis application.

Keywords texture synthesis, pixel-based, patch-based, constraint

1 引言

现实世界中的实体存在着各种各样的纹理,计算机纹理合成是真实感图像处理的一个重要分支。根据纹理生成方法的不同,纹理合成技术可分为纹理映射、过程纹理合成和基于样图的纹理合成3类。其中,纹理映射是绘制复杂场景真实感图形最为常用的技术,已经成为计算机图形图像硬件的标准技术;过程纹理合成是通过对物体生成过程的仿真来直接在曲面上生成纹理;而基于样图的纹理合成则可以在避免纹理映射和过程纹理合成适用范围有限

和调整参数繁琐的同时,表现几何模型无法体现的细节,这种合成技术近几年发展迅速,已经渐渐成为纹理合成技术发展的主流。

基于样图的纹理合成可分为2维图像纹理合成、曲面纹理合成和视频纹理合成3类。这3类纹理合成方法既有相似之处,又有各自特点,但基本思想是根据给定的小区域纹理样本的表面特征来生成与样本相似的纹理。

除了大规模场景生成、真实感和非真实感绘制、虚拟现实,以及视频动画等领域外,近年来,纹理合成技术还用于无线图像传输、数据压缩过程中丢失数据的恢复,这种修补技术也可用于古文物的虚拟

基金项目: 国家自然科学基金项目(60673088);浙江省自然科学基金项目(M603228)

收稿日期: 2006-05-23; **改回日期**: 2007-06-20

第一作者简介: 朱文浩(1979~),男。现为浙江大学计算机科学与技术学院博士研究生。主要研究方向为机器学习、图像处理。

E-mail: nixon3103@163.com

修复和图像的艺术风格学习。

根据目前国内外的研究状况,本文将基于样图的纹理合成所涉及的主要技术问题归结为合成采样、特征分析与匹配、约束合成及多目标合成等 4 个主要方面。

2 合成采样

合成采样主要包括如何在样图中选择种子点/块、如何提高邻域采点的匹配搜索效率,以及如何保证局部纹理的完整性等问题。在已进行的研究中,基于马可夫随机场模型(MRF)的方法最多,其主要分为“点匹配”和“块匹配”两大类。

2.1 点匹配算法

邻域匹配算法是点匹配算法中最为突出的代表。这种算法最早由 Efros 等人于 1999 年提出^[1],此后的许多算法是在此基础上改进的。Efros 等人提出的非参数化模型的主要思想是对于每一个将要合成的像素,都是在输入图像的“相似邻域”中确定像素值。对于输出图像中待合成的点,则是通过比对该点的邻域和输入图像中所有点的邻域来查找匹配点,并将查找到的与合成点邻域最“匹配”的点作为该点的值。

Efros 等人提出的算法不仅能够很好地保持图像局部信息的完整性,且对于大多数纹理都能取得比较好的合成结果。该算法存在的缺陷包括:(1)像素合成依赖于邻域匹配,当时并未涉及约束合成,而从后续的一些基于此算法的改进算法来看,约束的实施往往是在局部细节部分进行;(2)在产生了几个错误的合成结果后,还有可能因为比对邻域的误差而一直错下去;(3)由于点匹配的穷尽搜索策略,致使很小的纹理计算耗时也是相当大的;(4)对于有较大纹理结构的图像的合成,由于需要使用更大的邻域来保证最后的合成效果,因而使得计算的开销呈几何级增长。即便如此,Efros 提出的算法(以下简称 Efros 算法)可以说是现代纹理合成技术的一个里程碑。

在 Efros 算法基础上,Wei 等人提出了一个改进算法^[2],即采用多分辨率合成的方法来降低邻域的采样范围,同时通过树结构的矢量量化方法进行加速搜索。该改进算法采用了 L 形邻域代替原来的正方形邻域,这样由于在行扫描合成过程中,除了开始的几行,其余像素的合成都是根据已合成的点进

行的,从而进一步保证了合成结果。此外,Wei 等人提出的多分辨率图像金字塔^[8](高斯金字塔 Gaussian pyramid)还解决了大结构纹理的邻域扩大问题,其核心思想是先按照从低到高的顺序分别合成与输入图像对应级数的金字塔图像,然后利用相对少量的低分辨率合成像素来约束高分辨率像素的合成,使得在同样大小的邻域条件下,既能保留原有纹理中的结构,又可以加快处理速度。尽管该算法仍然采用穷尽搜索策略,但由于采用了数据压缩中的 TSVQ 算法,因此将合成速度提升了两个数量级。

Wei 等人提出的算法也存在缺陷,首先 L 形邻域匹配由于并没有考虑到人眼对于物体边界、转角等形状的敏感,因而在处理有明显结构的纹理时,会将边界细节等“忽略”掉,使图像变得模糊,这种情况在引入 TSVQ 加速算法后,更加明显;此外,虽然图像金字塔实现了对合成整体结构的控制,但当低分辨率合成出现偏差时,高分辨率合成可能将偏差“放大”。

2001 年, Ashikhmin 提出的邻域限定合成算法^[3]放弃了穷尽搜索,根据图像的局部相关性,将搜索范围由穷尽搜索减少为只对待合成像素邻域的点所对应相同偏移的像素进行搜索。该算法首先寻找待合成的像素 P 的 L 形邻域中像素 A 在输入图像中的合成点 $\hat{A}$,然后根据点 A 相对于点 P 的位移来得到点 P 的一个候选点 $\tilde{A}$ 。不断重复上述过程就可以得到待合成像素 P 的所有候选点,在进行邻域匹配时仅对这些点进行计算,从而大大减少了运算量。

相对于 Wei 等人提出的算法, Ashikhmin 提出的算法非常直观、简单,运算速度几乎提高了一个数量级。试验表明,它对绝大多数的自然纹理合成都能取得了很好的效果,但是相对于 Wei 等人提出的算法倾向于模糊化目标纹理来说, Ashikhmin 提出的算法正好相反,它是将目标纹理粗糙化,致使该算法通用性不强,对于水波、云彩等“无结构”的纹理,其合成结果往往较差。

虽然上述几个改进算法无论是合成质量还是在合成速度上,相对 Efros 等人提出的算法都有了很大的进步,但是还没有达到真正实时交互的目的。

Zelinka 等人通过将纹理合成分为分析和合成两个步骤,实现了“实时”的纹理合成^[4]。该算法借鉴了 Ashikhmin 提出的算法的思想,其关键改进表现在候选点是根据分析过程中的“跳跃图”得到的。在分析阶段,先利用邻域匹配算法得到输入图像中

每个点的“跳跃图”,即同该点有相似邻域的其他点及其选择概率;然后在合成输出图像像素点时,将“跳跃图”作为主要的依据。

Zelinka 等人提出的这个算法在合成阶段的速度是非常快的,基本达到了实时的要求。但是由于该算法在邻域的匹配方面仍然没有更大的突破,因此从合成效果来看并没有太大的提高。

2.2 块匹配算法

同点匹配算法不同,块匹配算法在进行合成采样时是以纹理块为基本单位。在提高合成速度和保持纹理局部特征方面,块匹配是较理想的选择。

Xu 等人提出的混沌拼接算法^[5]就是一个块匹配算法。该算法可以简单地分为以下两步:(1)先将样本纹理进行反复自映射,并通过混沌变换得到初步合成纹理;(2)然后通过平滑边界来得到最终的合成结果。为了产生一个视觉上随机的合成效果,该算法借鉴了非线性动力学的确定性混沌原理,并利用 Arnold 提出的 cat map,将图像中的随机纹理块通过混沌迭代重复映射到其自身。

该算法的最主要缺陷就是图像拼接过程过于随机,因而对于结构性的纹理,其合成效果往往比较差。

为了解决 Xu 等人提出的算法对结构性纹理合成的缺陷,Efros 等人在 2001 年提出了一个“图像缝合”算法^[6],Liang 等人在同一年也提出了一个非常相似的算法^[7]。同混沌拼接的方法不同,该算法的纹理块拼接过程并不是随机的,其整个合成过程是按照扫描线顺序,将搜索得到的纹理块经过处理后依次拷贝到目标纹理图像中。除了第 1 个纹理块是随机产生的,其他纹理块的产生都是按如下步骤得到的:

(1)指定纹理块的大小以及纹理块之间重合区域的大小;

(2)对于每一个待合成的纹理块,在输入的图像中寻找一组符合该纹理块边界约束的待选纹理块,并随机选择其中的一个;

(3)计算新的纹理块与已经合成区域重合部分的最小误差路径,并将其作为纹理块的新边界,同时将处理过的纹理块合成到目标图像中;

(4)重复步骤(2)、(3),直到完成整个图像的合成。

其中,第 2 步的边界约束使用的是类似文献[2]中的邻域匹配算法;第 3 步中的最小误差路径

用 Dijkstra 的最短路径算法得到,其路径长度是重合区域对应点之间的色彩距离。

Efros 等人提出的图像缝合算法存在的问题是有时纹理出现过度的重复,而且有些边界不匹配。但是这些已经不影响该算法成为纹理合成研究的又一个里程碑。

Cohen 等人于 2003 年提出了一个块合成算法^[8]。该算法引入了计算理论中的“Wang Tile”理论(或者叫做 Wang dominoes),由于其是利用合成的一组 Wang Tile 来进行一系列简单的拼接,从而产生了非周期性纹理。组内的 Tile 数量一般大于 8,而且数量越大,拼接的结果越丰富。

该算法在输入纹理中是先通过随机采样和边界匹配得到 4 个纹理块,其中各相邻边界处的差异要小于给定的阈值;然后每个 Wang Tile 都由这样的 4 个纹理块组合而成,拼接时要通过计算得到一组切割路径来处理纹理块之间的边界,以便对界定纹理区域进行相互邻接。该算法的合成速度和合成效果相对以前的算法有了明显的提高,并且还可以进行适当的扩展,使其能够生成均匀或非均匀的纹理,以及实时的 3 维表面渲染。文献[8]还介绍了用于控制纹理样图布局的泊松分布,并用角落标记来克服跨越两个以上边界的对象可能引起的重复计算。

块拼接的主要问题是块间各边缘的衔接,为了解决这一问题,Charalampidis 在 2003 年提出了改进算法^[9],就是先将图像按大小相等分块,然后在块之间进行聚类生成新的图像块,并按各块的聚类概率标注它们,合成时的标注顺序将作为一个限定条件。

2.3 点块混合算法

从上面的分析可知,基于区块的纹理合成在区块的边界处经常会引入一些不希望看到的分界线;而基于像素的纹理合成则在保持图像纹理一致性的同时,并不一定能够保证纹理的局部特征。

Nealen 等人提出了一个混合纹理合成算法^[10],该算法先进行块纹理合成,然后利用“点匹配”合成将纹理块之间“不匹配”(误差值大于用户的预设)的像素点重新合成。为了在用户定义的重叠容错值下使用尽可能大的区块,以便使整体结构的重建更为完善,该算法采用了递归区块分割方法进行搜索,如果搜索得到的区块同已合成的纹理之间的重合区域的总的不匹配度大于用户设定的阈值,则将该区块四等分,并将其作为新的输入纹理块递归查找,直

到不匹配程度小于阈值。在用“点匹配”对纹理合成进行修复时,还利用基于扩张的修复合成顺序来保证每个像素点的周围都有足够的合适邻域,以保证取得好的合成效果。

该算法对于大多数纹理,特别是有均匀结构的纹理都能得到很好的合成效果,但是由于重叠区域不匹配程度的计算仅仅建立在色彩的不匹配上,而不包括纹理中的结构或是语义信息,因此当遇到一些高频特征(比如细小纹理或物体边界)时,该算法仍没有能够有所突破。

类似的还有 Lee 等人在处理块拼接方法的边界问题时,结合基于像素的方法,在 L 形邻域度量中,提高了边界点的权重。在合成结果取得后,再进行一次重叠区域的处理,并在重叠区域内先找到最小误差路径,然后以此路径为中心划分出 5 pixels 宽的修正带,若修正带内的点误差超过指定的值,则采用由最小误差路径发展而来的区域生长法来修复超标点^[11]。

3 特征分析与匹配

特征分析匹配的思路来源于人对纹理的视觉感知,其基本策略是先抽取样图纹理的表面特征,并定义成特征规则集合,然后再通过重新分布特征生成新的纹理图案。这种方法比马尔可夫随机场方法效率更高。

在早期的一些算法中,Heeger 和 Bergen 提出的统计方法是比较典型的,他们通过匹配图像金字塔的边缘直方图进行纹理建模^[12],即先对金字塔用噪声进行初始化,同时建立样本纹理金字塔;然后对金字塔噪声进行更改,使其与样本纹理金字塔的颜色直方图匹配;最后再倒塌金字塔。如此多次迭代后,即可以合成较强结构性的纹理。

De Bonet 等人发表了进一步的研究成果,即利用多解采样进行纹理分析和合成,以部分解决纹理结构特征的提取^[13]。该算法采用了两个拉普拉斯金字塔及滤波器来处理纹理,并按从顶到底的顺序匹配,同时只考虑已合成的父层点,然后从符合条件的待选点中随机选一个。

类似的还有 Portilla 和 Simoncelli 等人在可控的金字塔模型中,利用统计方法结合小波分析^[14-16]来进行特征分析和匹配。

上述算法对于具有统计特征的纹理,也就是高

随机的纹理上比较有效,而对于具有较强结构性的纹理,其合成的效果仍然不能令人满意。

Dischler 等人在 2002 年提出了一种基于纹理特征(称为纹理粒子)的扩展纹理合成技术^[17]。其主要思想是先提取图像中的纹理粒子并分类;然后统计、分析各种粒子的分布情况;最后进行合成。其核心思想与基于 MRF 的跳跃图算法^[4]非常相似,只是合成的基本单位由单个的像素点变成了像素区域构成的纹理粒子。在对分割出的纹理粒子进行预处理后,再通过分析每一种粒子在输入纹理中的扩散方式(相邻粒子之间的位置关系),就可以得到各种粒子的空间分布。最后的纹理合成就变成了几种纹理粒子在目标图像中的反复扩散。

该算法同样是将纹理合成分解为分析和合成两个步骤,其中分析部分的计算量是非常大的,而合成过程则很快。这对于一个已经分析过的纹理来说,目标纹理的大小对于最后合成时间的影响是非常小的,并且用户还可以通过改变粒子空间分布的参数来部分实现合成结果的约束。该算法非常适用于一些结构性的纹理,其合成速度和纹理合成结果都能令人满意,但是对于形状与位置相关联或是有复杂结构的纹理(如龟裂的土地),该算法还是无能为力。

Wickramanayake 等人提出先利用小波变换对低频图像做 3 级分解,再进行基于块的纹理合成的算法^[18],虽然算法速度较快,但块的大小选择主要依据水平和垂直的重叠误差,并不能真正反映纹理的实际分布。

最近,微软研究院的 Lefebvre 和 Hoppe 在 Siggraph 2005 年会议上发表的并行可控的纹理合成论文^[19]中展示了他们所取得的新成果,该算法是基于邻域匹配的,同时采用了 Gaussian 栈作为新的分析结构,并且利用了结构坐标关联的向上取样、多轮次矫正方法。

最新的进展还有 Nicoll 等人提出的算法^[20],其基本思路是通过低通滤波和傅里叶分析来实现全局规则结构与非规则结构的分离,并按规则结构生成掩码,以控制非规则纹理细节的合成,从而在点匹配和块匹配两种情况下都取得了进展。

4 约束合成

随着研究的深入,特别是应用的需要,人们希望

对纹理合成过程施加更多的约束。事实上正是像图像修复、艺术风格类比等对精确性要求更高的应用使得约束合成被人们所重视。

4.1 基于类比的图像学习

Ashikhmin 等人在其实现的纹理合成算法中^[1],通过引入目标图像作为另一输入图像,实现了对合成结果一定程度上的控制。清华大学的 Bin Wang 等人用区域方向控制的纹理合成方法,将油画样本学习问题简化为纹理合成问题^[21]。

综合了文献[2]和文献[3]的算法后,Hertzmann 等人在 2001 年提出了一个具有多种功能的工作框架,把纹理合成推广到了更一般的情况^[22]。该框架可以处理如类比、纹理合成、纹理传输和编辑等。所谓图像类比就是通过分析和确定两幅内容相同但艺术风格不同的图像之间的“滤波”关系,将其应用在其他目标图像上,使之转换为与源图像有相似艺术风格的新图像。另外,该框架还利用掩码(mask)的方式来约束纹理合成。

在该框架算法中,对每一个待合成的像素点,都是通过比较相似匹配(邻域逼近算法)和一致性匹配(文献[3]算法),以及两者的加权值来得到合成结果。这就保证了在学到艺术风格的同时,还能保持目标图像的整体结构。但是,由于该算法进行了大量的邻域匹配运算,因此其运行速度并不理想。另外,色彩相似度的计算也过于简单。

4.2 主观约束纹理合成

近年来,约束从微观和宏观上都有了进展,并已出现了一些新的约束形式。

2003 年 Zhang 等人尝试利用图元掩码(texton mask)来实现 3 维/2 维表面的约束纹理合成^[23]。所谓图元(texton)同文献[17]中的“粒子”含义类似,指的是某种纹理元素。图元掩码的建立由用户交互地完成。提取了图元掩码后,用户还可以根据需要通过修改掩码来实现对最终合成纹理的约束。这种技术不仅弥补了在 3 维表面合成纹理时常出现纹理断裂问题,而且也可以应用在纹理的渐变/混合合成上。

2005 年以来,约束纹理合成取得了一些新的进展。Lefebvre 等人在进行并行可控的纹理合成时,提出了样本坐标的多解扰动原理^[19],并实现了直观的用户可控制纹理合成,包括多尺度随机、作用于样本和输出上的空间调整、特征拖放(feature drag-and-drop),以及周期约束等几种简单的扰动控制方式。

Vivek 等人在基于样本的纹理合成中通过引入优化算法来将合成问题形式转化为一个能量函数的优化问题^[24],并通过动态流场约束控制纹理的变化来用于视频动画中的水、火、烟等纹理生成。Wei Baogang 等人用分形图案结构来约束纹理合成,其目的是将其用于图案设计,并取得了预期的效果^[25]。

5 多目标合成

早期的纹理合成一般是针对特定的纹理类型,尽管没有统一的纹理定义,但通常是指由随机可重复元素构成的无穷图案。这种可重复性表明,单个纹理元素是有结构的,只是整体排列是随机无结构的。近年来,随着研究的深入,纹理的类型又涉及到了完全无结构的(例如云、烟、雾)、整体有结构的(例如织物等),甚至是同时考虑多种纹理。本文这里所说的多目标合成包括组合纹理合成与多样本纹理合成两种具体形式。尽管目前此方面的研究还不多,而且也不够深入,但从实用角度出发,笔者认为,今后会引起人们的广泛关注。

现实世界中的景物往往是多种纹理的组合,例如一个开阔的牧场并不都是草丛,还会有灌木丛、沙石、河流等,它们交织在一起所展现出的整体风貌才使人们感受到自然景色的美丽。组合纹理合成主要是针对这种多纹理复杂样本进行的。Zalesny 等人采用了层次方法,即先将组合纹理分割成不同的子纹理块,并用标号图(label map)对应于不同的子纹理^[26],标号是一组整数,其目的是为了记录样图中子纹理的整体结构和布局;然后按纹理的复杂性顺序对各子纹理进行排序,合成从最简单子纹理开始。这就简化了各纹理间过渡区域的处理,一个子纹理合成仅考虑与排在它前面的纹理是否可能存在的相交。

不光现实世界中景物纹理多是组合形式,而且人们在主观构造纹理时,往往也想将无直接关系的纹理组合在一起,使新的组合更具美感或艺术感染力。Bar-Joseph 等人曾采用统计学习的方法对两种纹理样本进行混合合成^[27],并取得了很好的效果,但由于该算法较为复杂,耗时长,只适用于一部分纹理,因而有一定的局限性;徐晓刚等人提出了一种基于文献[3]方法的多样本纹理混合合成算法^[28],该算法的主要改进是提高了合成速度。

6 总结与展望

总的来讲,2000 年以前,对纹理合成的研究主要是针对非结构性的随机纹理,而近几年人们则加强了对结构性纹理合成的研究。

合成采样算法研究的发展表明,点匹配方法除了合成速度慢以外,主要缺陷是误差积累问题,特别是对一些非均匀纹理,其合成结果并不理想,但要使合成结果能被适当控制,则很大限度需要依赖点匹配方法;块拼接方法不但提高了合成效率,也有利于保持纹理结构,但难以保证各块间相接部分的连续性。已有研究中,块的大小通常取固定值,或简单地由用户指定。笔者认为,基于块状的可变区域是更理想的选择,应该根据实际纹理的结构特征来确定区域的形状和大小,同时还要考虑对合成目标的要求,样本图像和合成结果图像的对象区域可以有不同的形状和大小。最近研究的“点”、“块”结合方法是要在合成效率和质量上达到一定的均衡,应该引起人们更大的关注。

虽然特征分析与匹配算法比马尔可夫随机场方法的效率更高,但是由于图像纹理的复杂性,对纹理特征的提取也只能是近似的,因此很难找到一个数学模型来准确表达纹理特征。以多分辨率为主的多解图像分析是特征匹配的主要技术,这是纹理合成的一大突破,它具有“邻域扩张”的效应,即能够用较小的邻域获得单分辨率中较大邻域的合成效果,这不仅有利于保持纹理的整体结构,而且可以有效地提高合成速度。

而约束合成和多目标综合合成则都需要更进一步的研究,仍存在较大的发展空间。造成这一现状的原因是先前的研究主要都是针对整幅图像的马尔可夫随机场模型。虽然该模型能够比较好地解决自然均匀的或随机的结构性纹理的合成,但它并不适合于所有类型纹理的合成,对于非均匀或无明显结构的纹理合成以及抽象的人工纹理合成还缺乏有效的技术方法,而在实际应用中,这些纹理又占有很大的比例。另外,通常情况下,人们也希望在保持原纹理特征的基础上,合成结果能够被适当受控,且有一定的变化,近年来约束合成技术的研究也反映出了这种趋势。但笔者认为,这还远远不够,目前所能进行的控制变化大都是局部微观上的,要在更广泛领域应用,约束还必须加强,包括约束的内容、形式和

机制等。

如果说组合纹理合成的难点在于各子纹理对象之间的衔接处理,那么多样本纹理合成更加突出了这一问题。目前多样本的纹理合成要么是随机的组合,要么是简单的纹理叠加,而欠缺的是人对不同纹理组合效果的控制。笔者认为,应该融入不同场合下的人对图案构成和色彩的感知,主要是人对实体对象的色彩、形态、图案构造等的认知规律,同时要研究建立这些与人的抽象思维相关的可计算模型和交互模型,以便实现宏观上的控制合成结果。

以上总结出的 4 个研究问题是相辅相成的,而且有效的约束合成必须建立在好的纹理特征分析的基础上,同时还要有高效率的纹理匹配和采样保证,而多目标综合合成则目前还处于研究的初始阶段。

从目前纹理合成的研究现状分析,下列具体的技术方法应该是今后纹理合成要着重解决的问题:

(1) 基于离散傅里叶变换的多分辨率纹理分析、识别技术和算法;

(2) 对象纹理的结构和特征及其相应的表示方法;

(3) 局部点匹配纹理合成相似性度量中的全局优化技术,组合纹理合成和多样图纹理合成中新的自适应块拼接技术;

(4) 纹理合成过程中的几何约束、统计约束等客观约束,人的物像与美术形象感知的主观约束,以及主客观约束在纹理合成过程中的控制策略和算法;

另外从实际应用的角度考虑,今后的研究重点还会在实时性和 3 维空间上的进一步拓展,以及在大规模场景中的实时渲染或是视频中应用。

参考文献 (References)

- 1 Efros A, Leung T. Texture synthesis by non-parametric sampling [A]. In: Proceedings of IEEE International Conference on Computer Vision [C], Corfu, Greece, 1999:1033 ~ 1038.
- 2 Wei L-Y, Levoy M. Fast texture synthesis using tree-structured vector quantization [A]. In: Proceedings of SIGGRAPH [C], New Orleans, Louisiana, USA, 2000:479 ~ 488.
- 3 Ashikhmin M. Synthesizing natural textures [A]. In: Proceedings of ACM Symposium on Interactive 3D Graphics [C], Chapel Hill, North Carolina, USA, 2001:217 ~ 226.
- 4 Zelinka S, Garland M. Towards real-time texture synthesis with the jump map [A]. In: Proceedings of 13th Eurographics Workshop on Rendering [C], Pisa, Italy, 2002:99 ~ 104.
- 5 Xu Y, Guo B, Shum H-Y. Chaos Mosaic: Fast and Memory Efficient

- Texture Synthesis [R]. Technical Report MSR-TR-2000-32, Microsoft Research, Seattle, WA, USA, 2000.
- 6 Efros A, Freeman W T. Image quilting for texture synthesis and transfer [A]. In: Proceedings of SIGGRAPH [C], Los Angeles, California, USA, 2001:341 ~ 346.
- 7 Liang L, Liu C, Xu Y, *et al.* Real-time Texture Synthesis by Patch-based Sampling [R]. Technical Report MSR-TR-2001-40, Microsoft Research, Seattle, WA USA, 2001.
- 8 Cohen Michael F, Jonathan Shade, Stefan Hiller. Wang tiles for image and texture generation [J]. ACM Transactions on Graphics, 2003, **22**(3):287 ~ 294.
- 9 Charalampidis Dimitrios. Texture synthesis based on cluster transition probabilities [J]. Proceedings of SPIE, 2003, **5108**:327 ~ 338.
- 10 Nealen Andrew, Alexa Marc. Hybrid texture synthesis [A]. In: Proceedings of 14th Eurographics Workshop on Rendering [C], Leuven, Belgium, 2003:97 ~ 105.
- 11 Lee Tong-yee, Yan Chung-ren. Feature-based texture synthesis [A]. In: Proceedings of International Conference on Computational Science and its Applications [C], Singapore, 2005:1043 ~ 1049.
- 12 Heeger D J, Bergen J R. Pyramid-based texture analysis/synthesis [A]. In: Proceedings of SIGGRAPH [C], Los Angeles, CA, USA, 1995:229 ~ 238.
- 13 Jeremy S, De Bonet. Multiresolution sampling procedure for analysis and synthesis of texture images [A]. In: Proceedings of SIGGRAPH [C], Los Angeles, CA, USA, 1997:361 ~ 368.
- 14 Simoncelli E P, Portilla J. Texture characterization via joint statistics of wavelet coefficient magnitudes [A]. In: Proceedings of Fifth International Conference on Image Processing [C], Chicago, Illinois USA, 1998, **1**:62 ~ 66.
- 15 Peachey Darwyn R. Solid texturing of complex surfaces [J]. Computer Graphics, 1985, **19**(3):279 ~ 286.
- 16 Portilla J, Simoncelli E P. A parametric texture model based on joint statistics of complex wavelet coefficients [J]. Journal of Computer Vision, 2000, **40**(1):49 ~ 71.
- 17 Dischler Jean-michel, Maritaud Karl, Lévy Bruno, *et al.* Texture particles [J]. Computer Graphics Forum, 2002, **21**(3):401 ~ 410.
- 18 Wickramanayake D S, Edirisinghe E A, Bez H E. Fast wavelet transform domain texture synthesis [A]. In: Proceedings of SPIE IS&T Electronic Imaging [C], San Jose, CA, USA, 2004, **5308**:979 ~ 987.
- 19 Lefebvre Sylvain, Hoppe Hugues. Parallel controllable texture synthesis [J]. ACM Transactions on Graphics, 2005, **24**(3):777 ~ 786.
- 20 Nicoll A, Meseth J, Muller G, *et al.* Fractional Fourier texture masks: Guiding near-regular texture synthesis [J]. Computer Graphics Forum, 2005, **24**(3):569 ~ 579.
- 21 Wang Bin, Wang Wen-ping, Yang Huai-ping, *et al.* Efficient example-based painting and synthesis of 2D directional texture [J]. IEEE Transactions on Visualization and Computer Graphics, 2004, **10**(3):266 ~ 277.
- 22 Hertzmann A, Jacobs C E, Oliver N, *et al.* Image analogies [A]. In: Proceedings of SIGGRAPH [C], Los Angeles, California, USA, 2001:327 ~ 340.
- 23 Zhang Jing-dan, Zhou Kun, Velho Luiz, *et al.* Synthesis of progressively variant textures on arbitrary surfaces [J]. ACM Transactions on Graphics, 2003, **22**(3):295 ~ 302.
- 24 Vivek Kwatra, Arno Schodl, Irfan Essa. Graphcut Textures: Image and Video Synthesis Using Graph Cuts [J]. ACM Transactions on Graphics, 2003, **22**(3):277 ~ 286.
- 25 Wei Bao-gang, Li Jian-ping, Pang Xiang-bin, *et al.* Using texture synthesis in fractal pattern design [J]. Journal of Zhejiang University, 2006, **7**(3):289 ~ 295.
- 26 Zalesny A, Ferrari V. Composite texture synthesis [J]. International Journal of Computer Vision, 2005, **62**(1/2):161 ~ 176.
- 27 Bar-Joseph Z. Statistical Learning of Multi-dimensional Textures [D]. Master's Thesis, The Hebrew University of Jerusalem, Jerusalem, Israel, June, 1999.
- 28 Xu Xiao-gang, Bao Hu-jun, Ma Li-zhuang. Texture synthesis based on relativity [J]. Progress in Natural Science, 2002, **12**(6):665 ~ 668. [徐晓刚, 鲍虎军, 马利庄. 基于相关性原理的多样图纹理合成方法 [J]. 自然科学进展, 2002, **12**(6):665 ~ 668.]