

# 一种利用 SSE2 多重纹理混合的大范围虚拟地形可视化技术

魏 勇<sup>1,2</sup> 丁雨淋<sup>3,4</sup> 龚桂荣<sup>1</sup> 杜 莹<sup>1</sup> 周 艳<sup>5</sup>

1 信息工程大学地理空间信息学院,河南 郑州,450052

2 四川省应急测绘与防灾减灾工程技术研究中心,四川 成都 6100412

3 西南交通大学地球科学与环境工程学院,四川 成都,611756

4 香港中文大学太空与地球信息科学研究所,香港,999077

5 电子科技大学资源与环境学院,四川 成都,610000

**摘 要:**大规模三维地形的真实感可视化对于表现战场自然环境具有重要意义,OpenGL、D3D 等主流三维图形引擎需要将多层纹理数据发送到 GPU 端进行多重纹理融合操作。对于瘦客户机,在显卡位宽和显存有限的资源下,大量的多重纹理融合操作常会导致 GPU 和带宽负载过大,出现绘制画面不流畅、渲染效率低等问题。为此,提出了一种基于 SSE2 的多重纹理混合技术,通过在 CPU 端进行混合纹理混合操作,降低 GPU 负载,减少数据传送时间。实验证明,该方法在保证 CPU 计算效率的前提下,有效地降低了大范围虚拟战场环境可视化过程中 GPU 和带宽负载,显著提升了瘦客户机中大范围虚拟战场数据的渲染效率。

**关键词:**多重纹理混合;SSE2;三维地形可视化;大范围虚拟地形环境

中图法分类号:P208

文献标志码:A

虚拟战场地形作为虚拟战场实现的基础,其构建和动态显示效果直接影响战场仿真过程和仿真结果的逼真度和可信度。如何构建一个高实时性、高逼真度的三维地形环境,成为目前战场环境仿真研究的重点<sup>[1,2]</sup>。在虚拟战场地形环境中,纹理发挥着重要的作用。一方面,地形纹理能够丰富形象地表达地貌特征,从可视化的角度弥补了 DEM 数据的精度问题,使得较低精度的 DEM 数据能表达丰富的地形起伏;另一方面,战场态势图等作为特殊的地形纹理,能够丰富虚拟战场内容。虚拟战场环境中纹理数据的来源广泛,包括各种遥感卫星的影像数据,传感器获取的态势数据,栅格化的矢量数据,扫描地图和战场态势图等。如何高效集成并展示这些多源纹理数据,有效表达战场环境和态势信息,已经成为虚拟战场环境信息系统成功与否的关键<sup>[3]</sup>。

多源纹理数据集成可视化的前提是对多源纹理数据进行混合<sup>[4]</sup>。目前多层纹理混合的实现方

法主要有两种:使用 OpenGL 或 D3D 等图形引擎提供的多重纹理映射机制<sup>[5]</sup>和使用着色语言方法<sup>[6-7]</sup>。使用多重纹理映射的方法易于实现,只需调用图形引擎提供的多重纹理混合 API 即可,其不足之处在于无法即时调整多层图像的透明度,且 OpenGL 的纹理组合器函数只能从三种来源接受颜色和 alpha 数据<sup>[5,8]</sup>,当纹理层数大于三层时,纹理组合器的使用极其不便。使用着色语言方法进行混合能够实现良好的纹理混合效果,但该方法对显卡要求较高,程序的移植性较差,不利于在瘦客户端实现。这两种方法的纹理混合操作均在 GPU 端完成,其中多层纹理数据从内存发送至显存时会占用较大显存及带宽<sup>[9-10]</sup>。在显卡位宽和显存资源有限的情况下,大量的多重纹理融合操作往往会导致 GPU 和带宽负载过大,从而出现绘制画面不流畅、渲染效率低等问题<sup>[11]</sup>。

针对上述问题,本文提出了一种基于 SSE2 多重纹理混合的大范围虚拟战场地形可视化方

收稿日期:2013-07-26

项目来源:国家自然科学基金资助项目(41101354,41471332,41201440);四川省应急测绘与防灾减灾工程技术研究中心开放基金资助项目(K2015B014)。

第一作者:魏勇,博士生,研究方向为信息抽取与可视化。E-mail:whuwuy@163.com

通讯作者:丁雨淋,博士。E-mail:rainforests@126.com

法,将 GPU 端进行的纹理混合工作转移到 CPU 端,使用 CPU 的并行处理方法在内存中先对纹理数据进行混合,从而减少对显存的占用,降低对显卡硬件的要求。

## 1 虚拟战场地形环境可视化系统中的分层纹理设计

虚拟战场涵盖了多种来源、多种格式的数据,这些数据包括两种类型:一种是目标数据,例如敌我双方重要目标的位置信息,这类数据可以直接在虚拟战场地形环境中以点状或三维实体表达;另一种是环境数据,例如进攻防御图、战场态势信息等,这些数据通常与地理环境相关,数据量较大,表达范围较广,不利于在虚拟环境中直接绘制,常用的策略是将其转换为图像并作为地形纹理来叠加显示。多层纹理数据的混合操作需解决两个问题,首先是图像纹理本身的透明区域和图层透明度的关系,其次是纹理混合的先后顺序。接下来分别阐述解决这两个问题的思路。

为了表达纹理的透明区域,本文将不同格式的图像数据采用 32 位 RGBA 色彩模式进行统一描述,RGBA 各波段采样深度为 8 bit,值域为 0~255。在进行不同图像的统一描述操作时,对不含透明通道的源图像的处理方式是:默认增加一个 alpha 通道,且缺省值均设为 255,以此表示该图像没有透明区域。此外,纹理颜色混合的常见处理方式是采用加法混合,为了避免加法混合操作中产生的色彩亮度增加问题,本文添加了图层透明度参数  $a$ ,以此控制当前层纹理数据在混合过程中所占的比例。在进行两层纹理的混合操作时,上层纹理为混合前景色,下层纹理为混合背景色,混合公式如下:

$$(RGBA) = (RGBA)_d \cdot A_d \cdot (1 - a) + (RGBA)_u \cdot A_u \cdot a \quad (1)$$

式中,  $(RGBA)_u$ 、 $A_u$  为上层纹理的像素值和 alpha 值;  $(RGBA)_d$ 、 $A_d$  分别为下层纹理的像素值和 alpha 值;  $a$  为上层纹理的透明度参数。

多层纹理混合的结果不仅受限于图层的透明度,还与图层的层次混合顺序有关。对于虚拟战场环境系统,本文提出纹理分层设计原则如下:① 基础纹理作为底层数据放在第一层;② 分辨率低的纹理数据放在下层,分辨率高的纹理数据放在上层;③ 地理范围小的纹理数据放在下层,地理范围大的纹理数据放在上层;④ 带透明通道的纹理数据放在无透明通道数据的上层;⑤ 战场

态势图等态势信息数据放在地形数据的上层。

## 2 基于 SSE2 的多重纹理混合算法

### 2.1 SSE2

SSE2(Streaming SIMD Extensions2)指令集被 Intel 官方称为 SIMD 流技术扩展 2 或数据流单指令多数据扩展指令集 2,是 Intel 公司在 SSE 指令集的基础上发展起来的一种 IA-32 架构的 SIMD(单一指令多重数据)指令集<sup>[12]</sup>。SSE2 中包括 `_mm64`、`_mm64i`、`_mm128`、`_mm128i` 等多种打包数据类型,每种打包数据类型均对应一个数学指令,可使用单一指令并行地处理打包数据<sup>[12]</sup>。相比于数据流单指令多数据扩展指令集 SSE,SSE2 新增了 144 指令和 5 种新的数据类型,支持 128 位整数和双精度浮点数运算,其对更高精度浮点数的处理能力使其成为加速多媒体程序的基础配置,从而大幅度提高了对影像、语音、图像和图片处理等软件的优化支持<sup>[13]</sup>。

### 2.2 SSE2 多重纹理混合算法

对于多层级纹理数据,基于 SSE2 的多重纹理混合算法步骤如图 1 所示,每次混合操作的核心步骤为:首先,将各层纹理数据的每 4 个像素数据打包成两个 16 位 `_mm128i` 类型的数据;然后,将前景色和背景色上下两层纹理数据对应的 `_mm128i` 类型数据进行混合;最后,将 `_mm128i` 类型的混合结果数据转换成对应的像素数据。理论上讲,一个 `_mm128i` 类型数值刚好能够表达 4 个像素,但这种打包结果在进行乘法运算时会产生数值溢出问题,因此本文采用每两个像素打包成一个 `_mm128i` 类型数据的方式,其中每个像素的 RGB 值分别采用两个字节来表示。为了进一步优化 SSE2 指令运算,提高效率,本文设计了两个优化策略。

**策略一** 将浮点数运算转换为整数运算。整数运算代替浮点数运算,既可节省打包空间,又能够提高运算效率。因此,在本研究中,浮点数类型的透明度参数  $a$  即通过乘以 255 转换为 `_mm128i` 整型类型,以此提高运算效率。

**策略二** 将除法运算转换为位移运算。在 SSE2 指令集中,除法运算占用的 CPU 运算周期较长,因此本文采用位移运算来近似模拟除法。除数为 255 的近似位移公式有两种:

$$x/255 \sim x/256 = x \gg 8 \quad (2)$$

$$x/255 \sim (x \cdot 32\,897 \gg 23) \sim (x + (x \gg 8) + 1) \gg 8 \quad (3)$$

计算可知,式(2)的误差为 3.91%,式(3)的误差为 0.389%。在实际应用中,式(2)和式(3)都能很好地混合无透明通道的纹理数据,但对于包含透明区域的纹理图像,例如矢量数据切片,式(2)的计算效果较差,多层混合后误差累积,造成透明度降低,纹理较模糊;因此,本文选用式(3)作为除法的近似运算。

基于上述两个策略,优化后的纹理融合公式如式(4)所示,式中  $T_u$  和  $T_d$  为 byte 类型数据,值域范围为 0~255; $a$  为浮点类型,值域范围为 0~1:

$$\begin{cases} T_u = a \cdot 255 \\ T_d = (1 - a) \cdot 255 \\ (RGBA) = [(RGBA)_u \cdot A_u \cdot T_u + (RGBA)_d \cdot A_d \cdot T_d + (RGBA)_u \cdot A_u \cdot T_u \gg 8 + (RGBA)_d \cdot A_d \cdot T_d \gg 8 + 1] \gg 8 \end{cases} \quad (4)$$

### 3 实验与分析

本文设计了两个实验验证该算法的有效性。实验一对比分析本文方法与基于 GPU 的纹理混合方法在进行大范围地形浏览操作时计算机的 CPU 使用率、内存使用量、GPU 负荷和 GPU 使用量;实验二对比分析基于 SSE2 的纹理混合方法与普通混合方法的算法效率。

#### 3.1 实验一

##### 3.1.1 实验数据及环境

地形实验数据共包括三层地形纹理,其中一层为基础纹理数据,另外两层为不同区域的细节纹理数据。实验环境为 Intel(R) 酷睿双核 T6600、2 GB DDRIII 内存、NVIDIA GeForce GT 240 M 显卡、512 MB 显存、Windows XP 操作系统。实验中,CPU 负荷和内存使用量使用 Windows 系统的任务管理器来记录,GPU 负荷和显存使用量通过 GPU-Z 记录。GPU-Z 是硬件网站 TechPowerUp 推出的一款轻量级显卡测试软件,能够显示 GPU 核心、运行频率、带宽以及 GPU 负荷、显存占用量等信息。

为了保证实验结果的正确性,实验一首先制定了一段地形浏览路径,且路径中包含视点拉近、推远、旋转、俯仰等常用地形操作;然后分别使用基于 SSE2 的纹理混合模式和基于 GPU 的纹理混合模式加载同一地形浏览路径,记录地形浏览过程中相同时刻的各项实验数据。战场地形可视化系统使用运行状态与空闲状态下系统内存和显存使用量差值来表示内存与显存的使用量,系统

空闲状态下 CPU 使用率为 1%,GPU 负荷为 0,因此可以近似将可视化运行时的 CPU 和 GPU 负荷作为系统耗用的负荷。

##### 3.1.2 实验结果和分析

实验记录了一段约 30 s 的地形浏览路径,然后采用 GPU 纹理混合模式与 SSE2 纹理混合模式分别启动三维地形浏览系统,记录 1、5、10、15、20、25、30 s 时刻的各项数值,具体结果如表 1 所示。由表 1 可知,基于 SSE2 的纹理混合方法能够使 GPU 使用率降低约 17% 左右,减少显存使用量约 240 MB;同时 CPU 使用率从 15.7% 增加到 26%,而内存使用量的增加量仅为 10.7 MB。由于虚拟战场环境数据在渲染地形纹理之外还需要渲染大量实体目标数据,占用较大显存量,因此,对于瘦客户机用户,减少显存占用量能够有效降低 GPU 负荷。基于 SSE2 的多重纹理混合能够有效减少 GPU 负荷和显存使用量,通过增加 CPU 计算量来平衡 CPU 和 GPU 的负载。

表 1 不同纹理混合模式下的 CPU 和 GPU 参数

Tab. 1 Information of CPU and GPU Under Different Texture Blending Method

		1	5	10	15	20	25	30
GPU 纹理混合模式	CPU 使用率/%	17	13	15	18	14	17	16
	内存使用量/MB	198	202	211	198	205	212	210
	GPU 使用率/%	35	24	36	38	31	36	35
	显存使用量/MB	465	465	467	487	511	512	471
SSE2 纹理混合模式	CPU 使用率/%	26	24	27	31	24	26	24
	内存使用量/MB	214	216	228	214	221	217	221
	GPU 使用率/%	13	8	14	8	9	7	8
	显存使用量/MB	223	225	232	223	223	225	223

#### 3.2 实验二

实验二对比使用 SSE2 的纹理混合方法与普通基于 CPU 的纹理混合方法的效率,通过一组不同纹理块的样本数据进行混合,记录两种方法的混合时间。

##### 3.2.1 实验数据及环境

实验二选用了某地区处理后的地形块纹理数据来进行纹理混合,每块地形纹理的分辨率为 256 像素×256 像素,数据格式为 32 位 BMP 图像;共进行 10 次实验,每次混合纹理块的数量分别为 1、4、9、16、25、36、49、64、81、100,详细实验结果如表 2 所示。

##### 3.2.2 实验结果和分析

实验结果表明,对于不同分辨率的混合纹理数据,混合的时间与图像像素大小之间近似于指数增长关系;使用基于 SSE2 的混合算法与普通混合算法相比,能够节省约 2/5 的混合时间。基

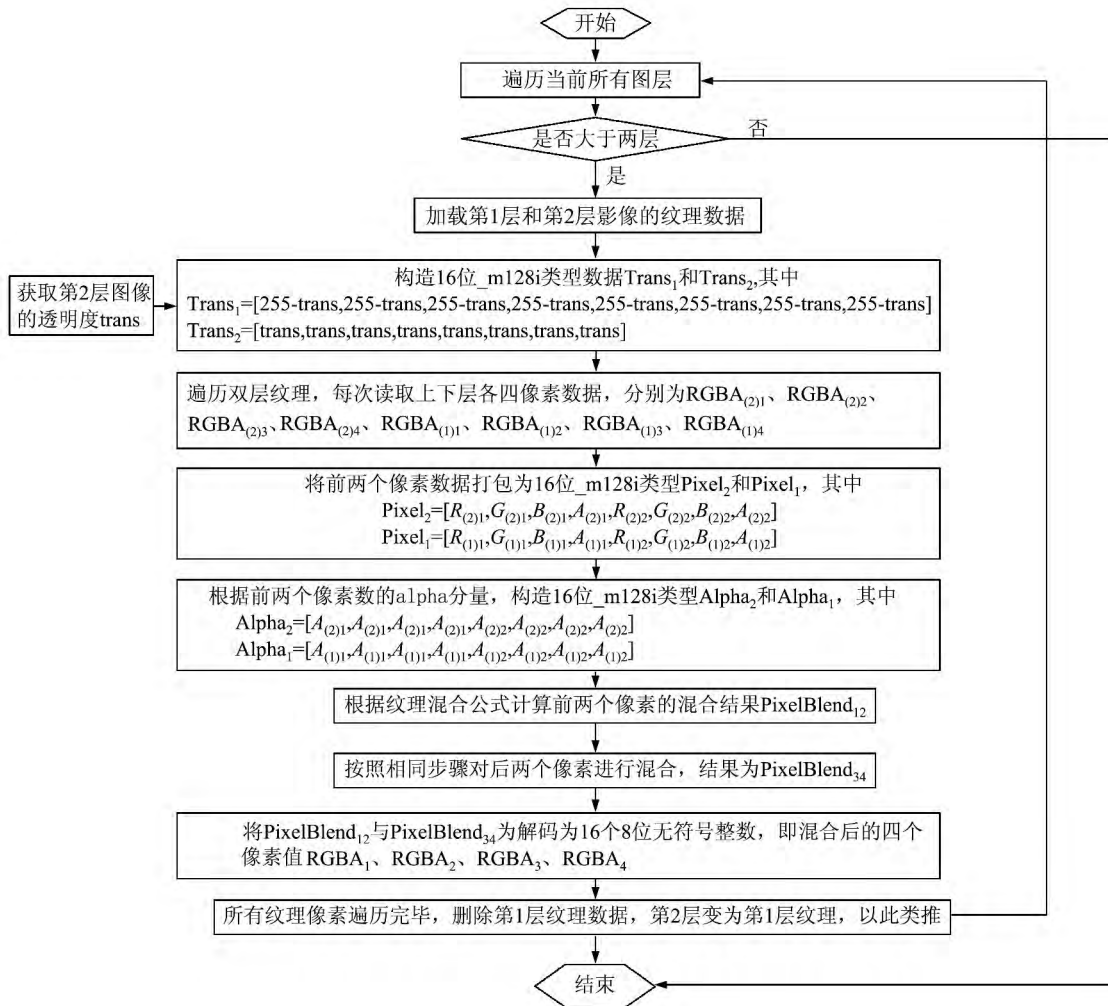


图 1 基于 SSE2 的纹理混合流程

Fig. 1 Flowchart of the SSE2 Based Texture Blending Algorithm

表 2 不同分辨率的纹理数据混合耗时

Tab. 2 Time-consuming Log of Samples with Different Resolution

数据名称	纹理块数	数据大小/字节	普通方法混合时间/ms	SSE2 方法混合时间/ms
样本 1	1	262 200	1.42	0.77
样本 2	4	1 048 800	4.68	2.83
样本 3	9	2 359 800	10.56	6.54
样本 4	16	4 195 200	28.92	11.52
样本 5	25	6 555 000	29.49	17.25
样本 6	36	9 439 200	46.97	28.36
样本 7	49	12 847 800	57.82	34.91
样本 8	64	16 780 800	75.53	45.99
样本 9	81	21 238 200	94.38	58.10
样本 10	100	26 220 000	118.24	72.83

于 SSE2 的纹理混合算法的实质是将两个像素进行打包,然后对打包后的数据进行混合,因此,从理论上来看,基于 SSE2 的纹理混合算法所用时间是普通混合算法的一半,实际测量的结果是普通混合算法耗时的 61%,产生这种情况的原因是纹理数据打包和解包也会占用一定的时间,因此

该算法的实际测量效率比理论效率略低。

### 3.3 具体应用

基于 SSE2 的多重纹理混合技术在“XX 综合应用系统”中得到了具体应用。为了灵活地对多源数据进行配置,系统提供了三维数据集设置工具,该工具可以读取本地、数据库以及网络上的不同格式数据,并实时调整影像图层号、设置图层对应的 LOD 级别。图 2 展示了全球低精度纹理数据、25 万仿真纹理数据、25 万影像数据以及 5 万影像数据 4 层纹理数据的三维地形仿真环境在不同比例尺下的纹理数据混合效果。

由图 2 可以看出,使用基于 SSE2 的优化算法进行纹理混合具有良好的纹理混合效果,能够满足虚拟战场环境的视觉要求。在该系统的纯地形环境中,使用 SSE2 进行纹理混合优化后,系统帧数由优化前的 43 帧提高到 62 帧,满足了虚拟战场地形可视化系统的流畅性需求,添加三维模型数据和战场态势数据之后,系统帧数也能保持在 35 帧以上。

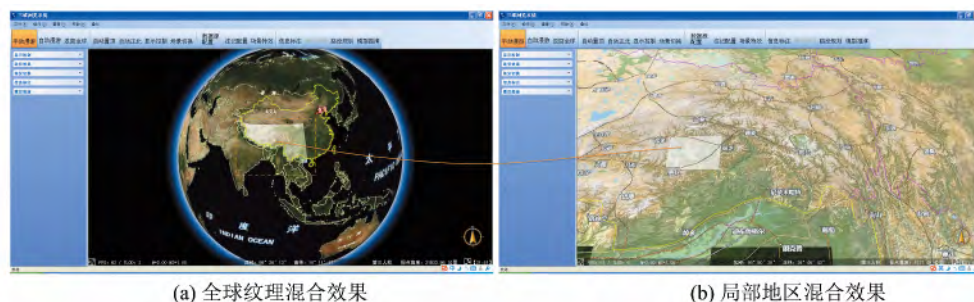


图 2 三维战场仿真系统

Fig. 2 Virtual 3D-battlefield System

## 4 结 语

本文设计并实现了一种基于 SSE2 的多重纹理混合算法。实验结果表明,该方法能够有效读取多种来源不同格式的纹理数据,并能实时地解码和混合。基于 SSE2 的多重纹理混合算法在对预处理过的地形块纹理混合时有着良好的表现,然而在虚拟战场环境中,还有很多纹理是实时动态生成的,这种纹理的内容和范围都是动态变化的,适合使用 GPU 来处理。因此,在大范围虚拟战场环境中,将基于 CPU 与基于 GPU 的纹理数据处理方法合理有效地结合起来,使得系统既能降低 GPU 负荷,提高 CPU 的利用率,实现 CPU 和 GPU 的负载均衡,又能高效处理多种类型纹理,是下一步的研究方向。

## 参 考 文 献

- [1] Sun Kai, Xu Xiaogang. Study of Satellite Cloud Image Simulation Technology Based on WorldWind[J]. *Science Technology and Engineering*, 2011, 11(22): 5 328-5 330, 5 339(孙恺,徐晓刚. 基于 WorldWind 平台的卫星云图模拟技术研究[J]. *科学技术与工程*, 2011, 22: 5 328-5 330, 5 339)
- [2] Jiang Jie. Research on Key Technologies in Global Virtual Geographic Environment Construction[D]. Changsha: National University of Defense Technology, 2010(蒋杰. 全球大规模虚拟地理环境构建关键技术研究[D]. 长沙:国防科学技术大学, 2010)
- [3] Sun Siliang, Chen Ren, Sun Li, et al. Modelling and Simulation of Virtual Earth Scene[J]. *Journal of System Simulation*, 2010, 11(12): 2 877-2 881(孙嗣良,陈韧,孙力,等. 虚拟地球场景的可视化建模与仿真[J]. *系统仿真学报*, 2010, 12: 2 877-2 881)
- [4] 刘星明,刘晓利,殷永凯,等. 真实感三维模型的纹理融合[J]. *计算机辅助设计与图形学学报*, 2012,

24(11): 1 440-1 446

- [5] OpenGL Architecture Review Board. OpenGL Programming Guide[M]. Beijing: Mechanical Industry Press, 2010(OpenGL 体系结构审核委员会. OpenGL 编程指南(第 7 版)[M]. 北京:机械工业出版社, 2010)
- [6] Bushnell J, Mitchell J. Advanced Multitexture Effects with Direct3D and OpenGL[J]. *Pyramid Peak Design & ATI Research*, 1999
- [7] Tomas M E H. Real Time Rendering[M]. USA: A. K. Peters, Ltd, 2000
- [8] Zhu Guozhong, Zhang Fan, Bu Hui. Real-Time Rendering for the Phong-Specular Surface by Multitexture[J]. *Computer Engineering and Science*, 2011, 7(33): 85-88(朱国仲,张帆,布辉. 使用多重纹理实时绘制 Phong 高光表面[J]. *计算机工程与科学*, 2011, 7(33): 85-88)
- [9] H R M, Karuppiah, S M S. Comparison of CPU and GPU Implementation of Computing Absolute Difference[C]. IEEE International Conference, Penang, 2011
- [10] R K J M. Comparison of FPGA and GPU Implementations of Real-time Stereo Vision[C]. IEEE Computer Society Conference, Los Alamitos, 2010
- [11] Xu Xiaogang, Ma Lizhuang. Texture Mixing and Texture Transfer[J]. *Journal of Computer-aided Design and Computer Graphics*, 2003, 15(1): 59-64(徐晓刚,马利庄. 纹理混合与纹理传输[J]. *计算机辅助设计与图形学学报*, 2003, 15(1): 59-64)
- [12] Skoglund J, Felsberg M. Fast Image Processing Using SSE2[OL]. [http://www2.cvl.liu.se/scout/Publications/Papers/ssba05\\_sf.pdf](http://www2.cvl.liu.se/scout/Publications/Papers/ssba05_sf.pdf), 2005
- [13] Intel. Intel 64 and IA-32 Architectures Optimization Reference Manual[OL]. <http://www.intel.com/content/dam/www/public/us/en/documents/manuals/64-ia-32-architectures-optimization-manual.pdf>, 2012

## A SSE2 Based Multi-source Texture Blending Method for Large-scale Virtual Terrain Simulation

WEI Yong<sup>1,2</sup> DING Yulin<sup>3,4</sup> GONG Guirong<sup>1</sup> DU YING<sup>1</sup> ZHOU Yan<sup>5</sup>

<sup>1</sup> Institute of Geospatial Information, Information and Engineering University, Zhengzhou 450053, China

<sup>2</sup> Sichuan Engineering Research Center for Emergency Mapping & Disaster Reduction, Chengdu 610041, China

<sup>3</sup> Faculty of Geosciences and Environmental Engineering, Southwest Jiaotong University, Chengdu 611756, China

<sup>4</sup> Institute of Space and Earth Information Science, The Chinese University of Hong Kong, Hong Kong 999077, China

<sup>5</sup> School of Resources and Environment, University of Electronic Science and Technology of China, Chengdu 610000, China

**Abstract:** The texture data plays an important role in three-dimensional topography simulation for virtual battlefield environment. 3D graphics engine such as OpenGL or D3D provides multi-texture mapping mechanism or shader, which is able to achieve the integration of multiple textures. But the achievement of multiple textures is GPU-based, which will take a large occupancy of GPU memory and bandwidth. In thin clients, such method will cause a lower render delay and lower efficiency. Therefore, a novel SSE2 based multi-texture blending method is presented in this paper, which implements the multiple textures blending process on CPU. The experiment proved that this method can effectively reduce the GPU load, achieve the integration of real-time multi-layered texture, and improve the rendering efficiency of three-dimensional topography simulation system.

**Key words:** multi-source texture blending; SSE2; three-dimension terrain visualization; large-scale virtual terrain environment

**First author:** WEI Yong, PhD candidate, specializes in information extraction and visualization. E-mail: whuw@163.com

**Corresponding author:** DING Yulin, PhD. E-mail: rainforests@126.com

**Foundation support:** The National Natural Science Foundation of China, Nos. 41101354, 41471332, 41201440, Open Research Fund by Sichuan Engineering Research Center for Emergency Mapping & Disaster Reduction, No. K2015B014.

(上接第 502 页)

## A New Method to Extract Terrain Feature Lines

XIONG Hanjiang<sup>1</sup> LI Xiujuan<sup>1</sup>

<sup>1</sup> State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan 430079, China

**Abstract:** In this paper, a new method for the extraction of ridge and valley lines is presented. It takes advantages of mean-shift and multifactorial Falling Water Model. The method consists of three principal steps: 1) By building the topology relationship between contours, the start points for tracing can be abstracted from them. 2) Based on the relationship between contour bendings and terrain feature lines, terrain points are divided into convexes and concaves. At the same time, virtual fan-shaped areas are built. 3) According to Falling Water Model and Maximum Probability Flows, ridge lines and valley lines are extracted. The experimental results presented demonstrate that the feature points can be well extracted and the noises and matching errors can be reduced.

**Key words:** Mean Shift; ridge lines; valley lines; terrain feature lines; terrain analysis; falling water simulation

**First author:** XIONG Hanjiang, PhD, professor, specializes in 3D GIS, Web GIS, and cartographic generalization. E-mail: xionghanjiang@163.com

**Corresponding author:** LI Xiujuan, postgraduate. E-mail: dykllxj@126.com

**Foundation support:** The National High Tehnology Research and Development Program of China(863 Program) Nos. 2013AA12A203, 2012AA12A405; Shenzhen Special Development of Strategic Emerging Industries, No. ZDSY20120618163343353.