

硕 士 研 究 生 读 书 报 告



题目 使用OpenGL ES在移动设备上呈现X3D内容

作者姓名 李林

作者学号 21651048

指导教师 李启雷

学科专业 移动互联网与游戏开发技术

所在学院 软件学院

提交日期 二○ 一七 年 四 月

# Rendering of X3D Content on Mobile Devices with OpenGL ES

Major Subject: Software Engineering

Advisor: liqilei

By

Lilin

Zhejiang University, P。R。 China

2017

摘要

移动设备在移动3D图形方面，有时配备图形加速器，将具有更加强大的可用性。在本文中，我们利用移动设备（OpenGL ES）上3D渲染的主要新兴标准，为X3D和H-Anim内容构建移动播放器（或称为MobiX3D）。 MobiX3D播放器的渲染引擎支持经典照明和着色算法。 我们讨论播放器的性能，并将其应用于手语可视化。

**关键词**：3D渲染、移动设备、OpenGL ES、X3D、手语

Abstract

The availability of more powerful mobile devices, sometimes equipped with graphics accelerators, is making it easier to experiment with mobile 3D graphics. In this paper, we exploit the main emerging standard in 3D rendering on mobile devices (OpenGL ES) to build a mobile player (called MobiX3D) for X3D and H-Anim content. The rendering engine of the MobiX3D player supports classic lighting and shading algorithms. We discuss the performance of the player and we apply it to sign language visualization.

**Keywords:** 3D rendering、mobiledevices、OpenGLES、X3D、sign language

区块链技术发展现状与展望

## 1、介绍

近年来，诸如个人数字助理（PDA）或高端移动电话之类的移动计算设备的日益增长，其性能已经允许这些设备支持越来越复杂的应用。然而，在移动设备上渲染3D图形仍然是一项艰巨的任务。移动设备的确具有桌面系统方面的一些严重限制：

1. 有限的CPU和内存；
2. 缺少或有限的图形加速器的性能；
3. FPU的缺乏或性能有限；
4. 移动设备硬件和软件设计人员的能源消耗问题；
5. 缺乏强大的开发和调试环境。

在本文中，我们使用OpenGL ES 1.1 API在移动设备上实验了基于多边形的X3D内容渲染，构建了一个实现经典着色，照明和导航算法的渲染引擎。这个渲染引擎被集成到我们的X3D移动播放器中， 称为MobiX3D。 目前，MobiX3D播放器完全支持H-Anim [人型动画工作组2004]标准，并支持X3D Interactive配置文件的一大部分。 此外，MobiX3D播放器已被用于移动设备上的手语可视化。 我们的最终目标是支持X3D Interactive配置文件。

本文组织结构为: 第二节调查相关工作。第3节详细描述了我们的工作，分析架构，MobiX3D播放器最有趣的实现细节，以及它支持的X3D的子集。第4节描述了在手语语言可视化的上下文中使用这个播放器。第5节讨论了我们的播放器的表现。第6节提供结论并概述未来的研究方向。

2，相关工作

2.1，移动设备上的3D渲染

虽然关于3D渲染的文献很广泛，但移动设备上的3D渲染仍然是几乎不被探索的主题。 此外，由于移动设备最近才达到了允许他们管理3D图形的性能，所以关于移动设备上的3D渲染的大多数可用的论文集中在硬件或远程渲染架构上，而不是机载软件解决方案。

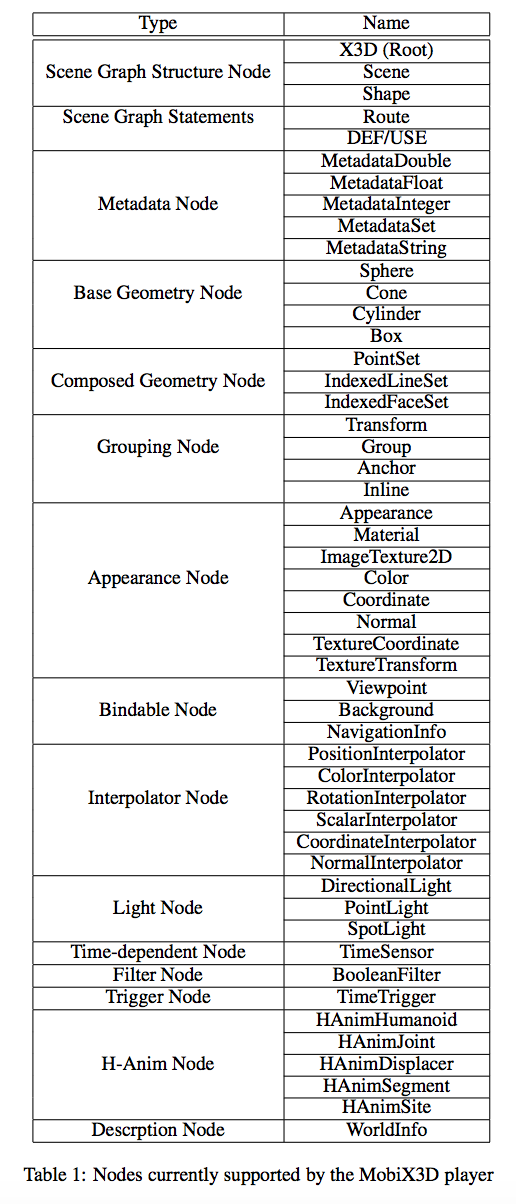
2.2 OpenGL ES

在最近几年中，已经提出了用于移动设备上的3D渲染的各种库。 Palm OS上的miniGL [Digital Sandbox Inc. 2000]和Microsoft PocketPC上的PocketGL [PocketGL 2000]是第一个，它们都是OpenGL API的子集。

3， mobieX3D 播放器

在本节中我们将详细介绍MobiX3D播放器。 为了开发我们的播放器，我们使用了混合Rasteroid库[Hybrid Ltd. 2005]，OpenGL ES 1.1 API的实现，以及用于渲染操作的GlutES [Pouderoux和Marvie 2005]工具包。 GlutES是用于OpenGL的Glut工具包的移动版本，并提供一组用OpenGL ES管理3D图形的高级功能。

MobiX3D播放器目前支持X3D Interactive配置文件和完整的H-Anim标准的子集。 MobiX3D当前支持的X3D节点列在表1中。



3.1，架构

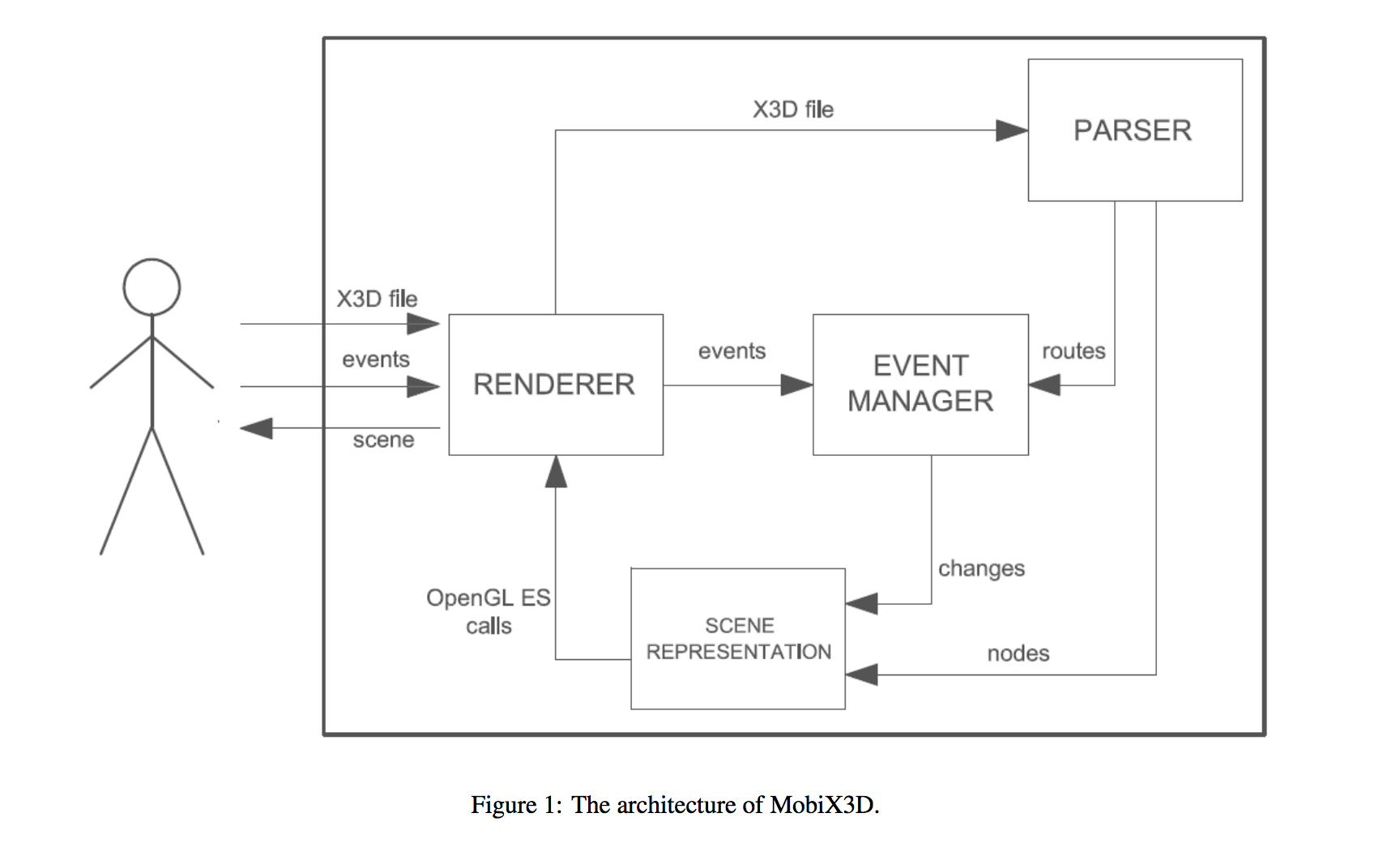
图1显示了MobiX3D播放器的高级体系结构，分为四个模块：

• 渲染器：此模块管理所有渲染过程，必要时调用其他模块。其输入是X3D文件，其输出是3D场景的显示。

• 事件管理器：此模块管理事件。它包含X3D场景的所有路线的集合。它的输入是一组事件（由Renderer提供），其输出是定义场景的X3D节点属性的一组变化。处理两种类型的事件：（i）用户事件：当用户与播放器交互时产生（例如，阴影模型改变）; （ii）动画事件：生成以实现动画（例如定时器更新）。

• 场景表示：该模块包含X3D场景图的内部表示。其输入是场景管理器发送的一组更改，其输出是对Renderer模块的一组调用，用于在显示器上显示这些更改。

• 解析器：该模块解析X3D文件。它的输入是一个X3D文件（从Renderer提供），其输出是定义3D场景的X3D节点集（发送到用于初始化场景图形的场景表示）和定义动画的路由集（发送到用于初始化其路由列表的事件管理器）。



3.2，实现

MobiX3D已经在C ++中开发用于PocketPC 2003平台。 已经实现了两个版本：一个使用eMbedded Visual C ++ 4.0，另一个使用Visual Studio 2005。

3.2.1，X3D场景管理

MobiX3D播放器中最复杂的部分是管理X3D动画和场景图形表示的部分。

在X3D中，动画由一组在节点之间建立事件路径的路由来实现。参与动画的节点通常是TimeSensor和Interpolator节点，用于构建时间依赖的动画，TimeTrigger和BooleanFilter节点，以便在上一部分完成后立即启动一部分动画，以及在动画过程中修改属性的几何节点。

在MobiX3D播放器中，路由由一个定时器调用的功能管理，每个定时器每50毫秒（每秒20次）进行一次。在此功能中，X3D场景中的所有Routes语句都被解决，如果动画处于活动状态，则会更新TimeSensor节点的值。然后，计算相关的Interpolator值，并更新几何节点属性。最后，计算TimeTrigger和BooleanFilter节点的值以启动动画的下一部分。

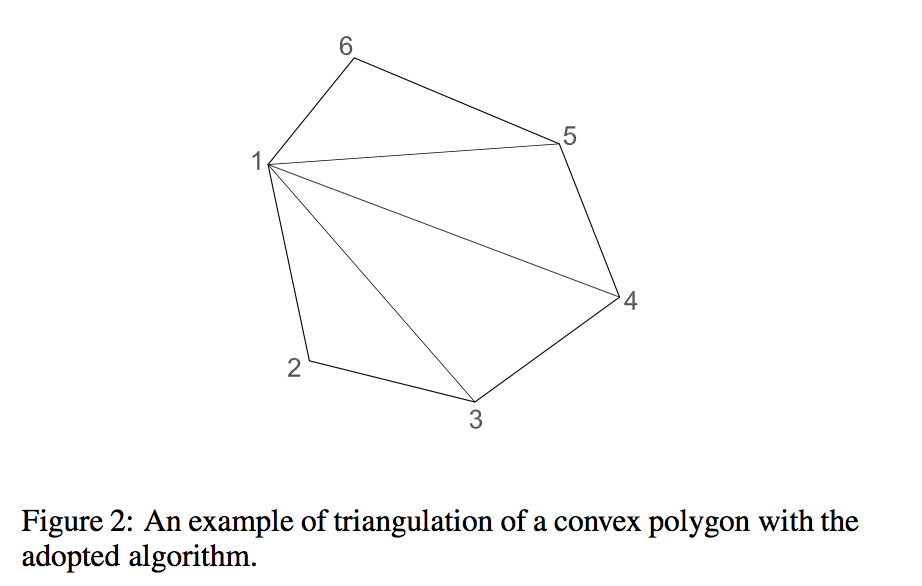
场景图由场景表示模块中的专用类表示。我们的渲染引擎支持的每个X3D节点都由相关的类来表示。与X3D节点相关的每个类都包含两种方法：绘制和解析。绘制方法包含渲染它们所属的X3D节点所需的OpenGL ES调用。解析方法包含它们属于的X3D节点的解析指令。这些方法在渲染引擎中具有根本的作用：通过顺序调用节点的解析方法，解析所有节点的属性的初始值，并将其解析存储在内存中。 X3D场景绘图通过依次调用场景中X3D节点的绘制方法进行。

3.2.2，一些实现事项

开发渲染引擎时遇到的问题与仅实现OpenGL功能子集的OpenGL ES的限制有关。

首先，OpenGL ES仅支持渲染三角形，线条和点，而OpenGL支持渲染所有（凸）多边形，线和点。X3D中的IndexedFaceSet节点指定一个由一组结构化的多边形（具有任意数量的边）组成的3D形状。每个多边形由顶点坐标中的索引列表指定。这些索引可以顺时针或逆时针顺序指定。因此，为了支持这个节点，我们必须开发一种将每个凸多边形转换为三角形的算法。

给定具有n个边的多边形P，我们的三角测量算法可以形式化如下：（i）从任意顶点开始，从1到n的数字标记多边形顶点。第一个顶点标有数字1，第二个标题为数字2，最后一个顶点的数字为n; （ii）多边形被划分成以下三角形组：{（1,2,3），…，（1，n-1，n）}。图2示出了由该算法进行三角测量的多边形的示例。



该算法不是最佳的三角形多边形，因为它假定多边形三角形是凸的，并迫使第一个顶点属于所有所得到的三角形。我们选择使用这个算法，因为它的复杂度是线性的，并且不需要额外的数据结构来工作：我们已经将该算法嵌入在Parser中，当Parser识别出一个新的多边形时，该算法使用Parser数据结构自动对其进行三角测量。在文献中有一些很好的算法（例如[Narkhede和Manocha 1995]）也支持非凸多边形，但是它们更复杂（O（n log n）），并且它们不容易集成到解析器中，因为它们需要附加数据结构。

所选择的算法可以使用OpenGL ES指令glDrawArrays（GL TRIANGLE FAN）实现，但是该指令在顺时针和逆时针顶点顺序之间具有不同的行为（逆时针顺序指示正面三角形，顺时针顺序指示背面三角形）。为了避免实体对象的后台淘汰，我们不使用glDrawArrays指令对多边形进行三角测量：我们的算法总是按逆时针顺序排列获得的三角形的顶点。

此外，在OpenGL ES中，只有一种渲染一组图元的方法。例如，为了渲染一组三角形，必须以三个步骤进行：（i）声明一个数组用于所有将被渲染的三角形顶点的坐标; （ii）使用glVertexPointer指令创建指向数组中定义的顶点的指针; （iii）使用glDrawArrays（GL TRIANGLES）指令同时渲染三角形。

在OpenGL中，其他方法可用于渲染一组三角形。最常用的方法是通过以下步骤：（i）执行glBegin（GL TRIANGLES）指令来启动顶点列表; （ii）使用glVertex指令来指定每个顶点的坐标（glVertex调用只能指定一个顶点）; （iii）使用glEnd指令来结束顶点列表。这种方法更灵活，因为可以在绘制原始图元（在glStart和glEnd指令之间）时计算顶点坐标，并且不需要指定将被渲染的图元数量。

图3显示了在执行手语时在我们的测试中使用的H-Anim人形生物。 手语中的短语是由X3D文件实现的，该文件包含H-Anim人形生物和由TimeSensors，TimeTriggers，BooleanFilters，OrientationInterpolators和Routes的结构化系列实现的动画。 在我们的测试中使用的手语语句是用H-Animator系统构建的[Buttussietal. 2006]。 我们用它来建模手语，并将它们连接成手语句子。

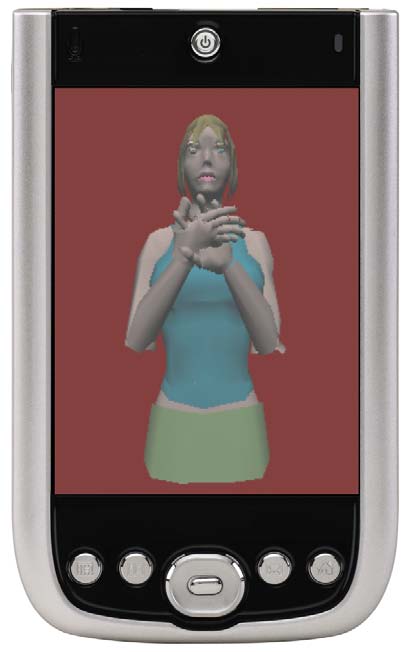


Figure 3: Humanoid performing sign language

4，手势语言应用

MobiX3D播放器已被用于在移动设备上显示手语语句。手语语言可视化对于聋哑儿童的手语教学，定义手语视觉词典或加上语音识别引擎，将自然语言翻译成手语可能是有用的。然而，显示手语是一项艰巨的任务：如果动画不是非常精确，用户可能会误会手势的含义。此外，当没有与单词相关联的手势时，需要手写该单词。手指拼写包括用手指运动拼写一个字母的字母，由于手指的大小和与字母相关联的运动的相似性，它很难显示。

由于这些原因，尽管在文献中已经研究了手语可视化，但由于性能有限，所以没有关于移动设备的提案。

手语语言可视化被用于教聋儿童的手语[Karpouzis et al。 2006; Geitz et al。 1996年佐川和竹内2002]或定义手语视觉词典[Wilcox et al。 1994]。 ViSiCAST项目[Elliott et al。 2000]旨在通过使用监视器来改善公共服务的可及性，这些监视器显示人形生物执行员工所说的手语翻译。显示人形生物比使用文字字幕更好的解决方案有两个原因：（i）大多数聋人认为手语是母语; （ii）人形生物可以吸引用户的关注焦点，并能传达更多的对话和情感线索。

5，总结和未来的工作

据我们所知，MobiX3D是第一款可用于移动设备的公众X3D播放器。我们项目的最终目标是支持完整的X3D Interactive配置文件和H-Anim标准。 MobiX3D可以从http://hcilab.uniud.it/MobiX3D下载。使用MobiX3D播放器，我们目前通过X3D世界获得3.9 fps导航，大约有30000个三角形和5.54 MB纹理。

我们的研究现在正在几个方面进行。首先，我们将使用更有效的算法进行着色和纹理化，如[Ström和Akenine-Möller2005]中所述的算法。

其次，我们将考虑将MobiX3D移植到其他平台（例如Linux，Symbian或Palm）。

第三，我们将改进手语视觉效果器，建立更准确的手势并改善人形生物。然后，我们将在聋哑用户对手语翻译应用程序进行测试。

最后，我们将研究在移动设备上实现无缝形状变形算法（例如[Babski和Thalmann 1999]）的可能性。无缝形状变形算法将人形生物映射为单个网格，并将运动变形成该网格的变形。这些变形是参数化的，并模拟皮肤，肌肉和骨骼的行为。这些算法解决了传统运动中的不连续问题。

6，致谢

Roberto Ranon和Stefano Burigat在描述的作品的开发过程中提供了宝贵的建议。

我们的研究部分得到意大利教育部，大学和研究部（MIUR）在PRIN 2005项目“自适应，上下文感知，移动设备多媒体指南”项目的支持。

参考文献

ANTOCHI, I., JUURLINK, B., AND VASSILIADIS, S. 2002. Selecting the Optimal Tile Size for Low-Power Tile-Based Rendering. In ProRISC 2002: Proceedings of the thirteenth Annual Workshop on Circuits, Systems and Signal Processing, 1–6.

BABSKI, C., AND THALMANN, D. 1999. A Seamless Shape for HANIM Compliant Bodies. In VRML ’99: Proceedings of the fourth symposium on Virtual reality modeling language, ACM Press, New York, NY, USA, 21–28.

BURIGAT, S., AND CHITTARO, L. 2005. Location-aware Visualization of VRML Models in GPS-based Mobile Guides. In Web3D ’05: Proceedings of the tenth international conference on 3D Web technology, ACM Press, New York, NY, USA, 57–64.

BUTTUSSI, F., CHITTARO, L., AND NADALUTTI, D. 2006. H-Animator: A Visual Tool for Modeling, Reuse and Sharing of X3D Humanoid Animations. In Web3D ’06: Proceedings of the eleventh international conference on 3D Web technology, ACM Press, New York, NY, USA.

CHANG, C.-F., AND GER, S.-H. 2002. Enhancing 3D Graphics on Mobile Devices by Image-Based Rendering. In PCM ’02: Proceedings of the Third IEEE Pacific Rim Conference on Multimedia, Springer-Verlag, Berlin, Germany, 1105–1111.

DIGITAL SANDBOX INC. 2000. MiniGL. http://www.dsbox.com/minigl.html.

DUGUET, F., AND DRETTAKIS, G. 2004. Flexible Point-Based Rendering on Mobile Devices. IEEE Computer Graphics and Applications 24, 4, 57–63.

ELLIOTT, R., GLAUERT, J. R. W., KENNAWAY, J. R., AND MARSHALL, I. 2000. The Development of Language Processing Support for the ViSiCAST Project. In Assets ’00: Proceedings of the fourth international ACM conference on Assistive technologies, ACM Press, New York, NY, USA, 101–108.

GEITZ, S., HANSON, T., AND MAHER, S. 1996. Computer Generated 3-Dimensional Models of Manual Alphabet Handshapes for the World Wide Web. In Assets ’ 96: Proceedings of the second annual ACM conference on Assistive technologies, ACM Press, New York, NY, USA, 27–31.

GRIMSTEAD, I. J., AVIS, N. J., AND WALKER, D. W. 2005. Visualization Across the Pond: How a Wireless PDA can Collaborate with Million-Polygon Datasets via 9,000km of Cable. In Web3D ’05: Proceedings of the tenth international conference on 3D Web technology, ACM Press, New York, NY, USA, 47–56.

HUMANOID ANIMATION WORKING GROUP. 2004. H-Anim. http://h-anim.org.

HYBRID LTD. 2005. Rasteroid. http://www.hybrid.fi/main/esframework/index.php.

INTEL. 2004. Intel 2700G Graphics Accelerator. http://www.intel.com/design/pca/prodbref/300571.htm.

KAMEYAMA, M., KATO, Y., FUJIMOTO, H., NEGISHI, H., KODAMA, Y., INOUE, Y., AND KAWAI, H. 2003. 3D Graphics LSI Core for Mobile Phone ”Z3D”. In HWWS ’ 03: Proceedings of the ACM SIGGRAPH/EUROGRAPHICS conference on Graphics hardware, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 60–67.

KARPOUZIS, K., CARIDAKIS, G., FOTINEA, S.-E., AND EFTHIMIOU, E. 2006. Educational Resources and Implementation of a Greek Sign Language Synthesis Architecture. Computers & Education, Special Issue in Web3D Technologies in Learning, Education and Training, Elsevier. (In press).

KESSENICH, J., BALDWIN, D., AND ROST, R. 2004. The OpenGL Shading Language. http://www.opengl.org/documentation/oglsl.html.

KHRONOS GROUP. 2003. OpenGL ES. http://www.khronos.org/opengles/.

LAMBERTI, F., ZUNINO, C., SANNA, A., FIUME, A., AND MANIEZZO, M. 2003. An Accelerated Remote Graphics Architecture for PDAs. In Web3D ’03: Proceeding of the eighth international conference on 3D Web technology, ACM Press, New York, NY, USA, 55–61.

NARKHEDE, A., AND MANOCHA, D. 1995. Fast Polygon Triangulation Based on Seidel’s Algorithm. In Graphics Gems V, A. W. Paeth, Ed. AP Professional, Boston, MA, USA, 394–397.

PARALLELGRAPHICS. 2004. Pocket Cortona. http://www.parallelgraphics.com/products/cortonace/.

PASMAN, W., AND WOODWARD, C. 2003. Implementation of an Augmented Reality System on a PDA. In ISMAR ’03: Proceedings of the second IEEE and ACM International Symposium on Mixed and Augmented Reality, IEEE Computer Society, Washington, DC, USA, 276.

POCKETGL. 2000. http://www.sundialsoft.freeserve.co.uk/pgl.htm.

POUDEROUX, J., AND MARVIE, J. E. 2005. GlutES. http://iparla.labri.fr/softwares/glutes/.

SAGAWA, H., AND TAKEUCHI, M. 2002. A Teaching System of Japanese Sign Language Using Sign Language Recognition and Generation. In MULTIMEDIA ’02: Proceedings of the tenth ACM international conference on Multimedia, ACM Press, New York, NY, USA, 137–145.

SANNA, A., ZUNINO, C., AND LAMBERTI, F. 2004. A Distributed Architecture for Searching, Retrieving and Visualizing Complex 3D Models on Personal Digital Assistants. International Journal of Human Computer Studies 60, 701–716.

SOHN, J.-H., WOO, R., AND YOO, H.-J. 2004. A Programmable Vertex Shader with Fixed-point SIMD Datapath for Low Power Wireless Applications. In HWWS ’04: Proceedings of the ACM SIGGRAPH/EUROGRAPHICS conference on Graphics hardware, ACM Press, New York, NY, USA, 107–114.

STRO ̈ M, J., AND AKENINE-MO ̈ LLER, T. 2005. iPACKMAN: High-quality, Low-complexity Texture Compression for Mobile Phones. In HWWS ’05: Proceedings of the ACM SIGGRAPH/EUROGRAPHICS conference on Graphics hardware, ACM Press, New York, NY, USA, 63–70.

WILCOX, S., SCHEIBMAN, J., WOOD, D., COKELY, D., AND STOKOE, W. C. 1994. Multimedia dictionary of American Sign Language. In Assets ’94: Proceedings of the first annual ACM conference on Assistive technologies, ACM Press, New York, NY, USA, 9–16.

WILL, H. M. 2004. Vincent. http://sourceforge.net/projects/ogl-es.

WOO, R., YOON, C. W., KOOK, J., LEE, S. J., AND YOO, H. J. 2002. A 120mW 3D Rendering Engine with 6Mb Embedded DRAM and 3.2Gbyte/s Runtime Reconfigurable Bus for PDA-Chip. IEEE Journal of Solid-State Circuits 37 (Oct.), 1352–1355.