

State of the Art and Development of Wearable Computer Graphics Processing Unit

Jiye Jiao¹, Tao Li², Huimin Du², Jungang Han¹

¹School of Computer Science, Xi'an University of Posts and Telecommunications, Xi'an Shaanxi

²School of Electronic Engineering, Xi'an University of Posts and Telecommunications, Xi'an Shaanxi

Email: jiaojie@126.com

Received: Aug. 19th, 2015; accepted: Sep. 3rd, 2015; published: Sep. 14th, 2015

Copyright © 2015 by authors and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

Emerging wearable computer has tremendous potential applications, and the use of GPU rendering rich, intuitive and highly interactive human-machine interface is the core technology for many wearable applications. This survey of wearable graphics processor describes the current state of research and development in the field, analyzes the trend of GPU for wearable device, and gives key technologies of GPU implementations for wearable computer.

Keywords

GPU, Wearable Computer, Low Area, Low Power, Data Secure

适合可穿戴设备的图形处理器发展现状和发展趋势

焦继业¹, 李 涛², 杜慧敏², 韩俊刚¹

¹西安邮电大学计算机学院, 陕西 西安

²西安邮电大学电子信息工程学院, 陕西 西安

Email: jiaojie@126.com

收稿日期: 2015年8月19日; 录用日期: 2015年9月3日; 发布日期: 2015年9月14日

摘要

新兴的可穿戴设备蕴藏着巨大的潜在应用价值，采用GPU渲染的交互性强的人机界面是众多可穿戴应用的核心支撑技术。本文主要介绍了适合可穿戴设备的GPU的发展现状，分析目前适合可穿戴设备的GPU发展中存在的问题和发展趋势，并给出适合可穿戴设备的GPU实现关键技术。

关键词

图形处理器，可穿戴设备，小面积，低功耗，数据安全

1. 引言

过去两年，可穿戴设备成为新的研究领域的主角，从智能手表、手环、眼镜等可穿戴设备延伸到智能家居、智能汽车、医疗健康等更广泛的领域，而拥有用户界面良好的人机接口则成为可穿戴设备的基本要求。可穿戴设备人机界面的首要目标就是要使屏幕显示内容展现的直观、自然，并且宜于使用，因为这是用户和设备之间第一个接口。虽然可穿戴设备显示屏幕较小，所有交互的信息(如文本、图像、图标、视频等)都需要简洁方式显示在屏幕上，但较小屏幕的并不意味着设备处理能力简单，相反的，作为一个单独应用 SoC 处理器，其不仅需要实时完成数据处理、数据通信、传感器信息分析处理等，还需要完成用户界面显示，因此可穿戴设备必须采用图形处理器(graphic process unit, GPU)支持，才能完成所需的用户界面信息显示。

相比移动 GPU，可穿戴设备的 GPU 应用有自己特点。首先由于可穿戴设备是电池供电，相比手机应用，可穿戴设备需要更长的工作时间，但图形显示占用设备大量功耗，所以应用于可穿戴设备的 GPU 必须采用多种方法进行超低功耗处理，保证设备可以长时间工作。其次可穿戴设备本身需要小型化，整机尺寸要小、物料(bill of material, BOM)要少，因此可穿戴设备的 GPU IP 面积必须要小，同时也需要减少 GPU 渲染所需的外部 DDR 存储。而采用目前常用的移动 GPU 实现可穿戴设备的人机界面渲染，存在诸多使用限制，如移动 GPU 芯片面积过大，支持的图形功能太复杂，所需 DDR 存储较大，工作时功耗较大导致发热严重，图形驱动程序复杂、占用系统开销较大、着色器预编译时间较长等。所以需要针对可穿戴设备的 GPU 特点，进行小面积、超低功耗的 GPU 体系结构的研究，而这种针对可穿戴设备的小面积、超低功耗的 GPU 我们称之为可穿戴 GPU。

2. 可穿戴设备的 GPU 发展现状

2014 年 3 月，Google 发布了可穿戴设备平台 Android Wear [1] [2]操作系统，该操作系统图形显示部分必须需要硬件的 GPU 支持，才能实现对可穿戴设备图形显示的多种优化，所以，从 Google 的 Android 的生态环境角度看待这个问题，运行 Android Wear 操作系统的显示界面精美的可穿戴设备必然迎来爆发式增长。因此，采用 GPU 渲染的丰富、直观、交互性强的人机界面将是众多可穿戴设备的核心支撑技术。目前支持 Android Wear 操作系统的智能手表产品有的 MOTO 360、LG G Watch 、HUAWEI Watch 等，这些智能手表界面精美，但一般仅能工作一天，就需要再次充电，因此可穿戴设备的低功耗设计是一个首要解决的问题。

2.1. GPU 国内研究现状

根据文献，拥有 OpenGL ES 3D 图形 IP 核的图芯科技公司(Vivante Corporation)，直到 2009 年才在中

国有了第一家客户。而中国关于 GPU 的研发项目是在 2009 年才有所展开, 到目前为止也只有长沙景嘉微电子一家有成型的芯片, 其采用 $0.13\ \mu\text{m}$ 的工艺实现, 3D 图形处理性能指标仅达到早期桌面级图形硬件加速的水平。西安邮电大学从 2009 年开始自主研究 GPU, 提出了文献[3] [4]所示多态并行图形架构, 该结构的处理单元可以在 SIMD 和 MIMD 两种模式下运行, 兼有异步执行机制, 还可以实现分布式指令并行处理。配套软件包含 OpenGL1.3 和 2.0、扩展的 OpenCL [5]语言和编译、并行编程开发环境 IDE 和仿真器。同步完成的还有 OpenVX 机器视觉系统[6]。同时自主研发的另外一款嵌入式 GPU [7] [8]芯片, 已经进入测试阶段。此外国内研究图形加速器的单位还有北大众志微处理器研究中心、中国科技大学、西安电子科技大学[9]等单位。总体来说, 国内有关 GPU 的研究都在起步阶段, 特别是对于可穿戴 GPU 缺乏系统研究。

2.2. 可穿戴 GPU 国外研究现状

国外移动 GPU 研究已经较成熟, 但是可穿戴 GPU 应用才出现不久, 所以针对可穿戴 GPU 的研究都处于起步阶段, 但国外研究机构已经展开对可穿戴 GPU 这一新兴领域的大量研究。

2.2.1. Think Silicon 公司的 NEMA-t100 可穿戴 GPU

2015 年 6 月 4 日, 希腊 Think Silicon 公司发布了工业界面积最小的超低功耗的 3D GPU NEMA-t100 [10], 主要面向可穿戴或者非可穿戴的物联网市场。NEMA-t100 的低功耗特性可以将可穿戴设备的电池从一天延长到五天时间。采用 28 nm 工艺, NEMA-t100 的面积仅有 $0.1\ \text{mm}^2$, 可以运行在 400 MHz。NEMA-t100 采用内存数据压缩技术和 3D 图形算法加速技术, 可运行 OpenGL ES API, NEMA-t100 和 DDR 存储的总功耗仅有 0.07 mW, 而处理器本身功耗仅有 0.03 mW。图 1 为 NEMA-t100 的架构图, 该架构特点是显示缓冲区采用压缩算法实现, 减少缓存面积; Z Buffer 和纹理缓冲区采用对称的压缩和解压缩算法, 同样减少缓存面积, 降低与外部 DDR 存储的吞吐量, 降低系统功耗; 图形光栅化采用硬件实现。NEMA-t100 仅有一个内核, 采用 VLIW 指令集, 支持多线程超标量架构。

2.2.2. DMP 公司的 ANT 可穿戴 GPU

日本 DMP 公司于 2014 年 6 月 31 日发布了针对可穿戴和物联网设备的 GPU IP 核 ANT [11]。ANT 采用 TSMC 28 nm HPC 工艺实现, 面积仅为 $0.25\ \text{mm}^2$, 主要面向可穿戴设备小尺寸显示设备中, 或者其他物联网小尺寸显示设备中。该处理器实现了在小尺寸屏幕上的低功耗和高性能的 2D/3D UI 显示完美结合。

2.2.3. Imagination 公司的 PowerVR GX5300 可穿戴 GPU

英国 Imagination 公司于 2014 年 7 月发布了一款与 Android 兼容的针对可穿戴和物联网设备的 GPU IP 核 GX5300 [12]。GX5300 主要针对可穿戴设备的超低功耗需求设计, 采用 TSMC 28 nm 工艺, 面积仅为 $0.5\ \text{mm}^2$, 符合新的可穿戴设备的高清晰度 GPU 图形性能需求, 支持 480 p 到 720 p 分辨率, 以及最小的 OpenGL ES 2.0 图形集。GX5300 架构如图 2 所示, 采用 PowerVR 系列的区块延迟渲染(tile based deferred rendering, TBDR)的统一着色器架构专利, 该架构可明显减少 GPU 与外部存储器之间的数据吞吐量, 减少系统功耗; 针对 OpenGL ES 2.0 标准中不需要高精度运算, 采用低功耗 16 位浮点数和高精度 32 位浮点数相结合的混合图形运算单元, 减少运算量, 降低功耗; 采用 PVRTC 压缩纹理格式, 减少内存带宽和功耗, 进一步减少系统整机内存需求。

2.2.4. Vivante 公司的 GCNano 可穿戴 GPU

美国 Vivante 公司于 2014 年 6 月 2 日发布针对可穿戴和物联网产品应用的 GPU 产品线, 包括 GCNano Lite、GCNano 和 GCNano Ultra [13]。该产品线针对 MCU/MPU 产品进行了各种优化, 采用 28 nm LP 工

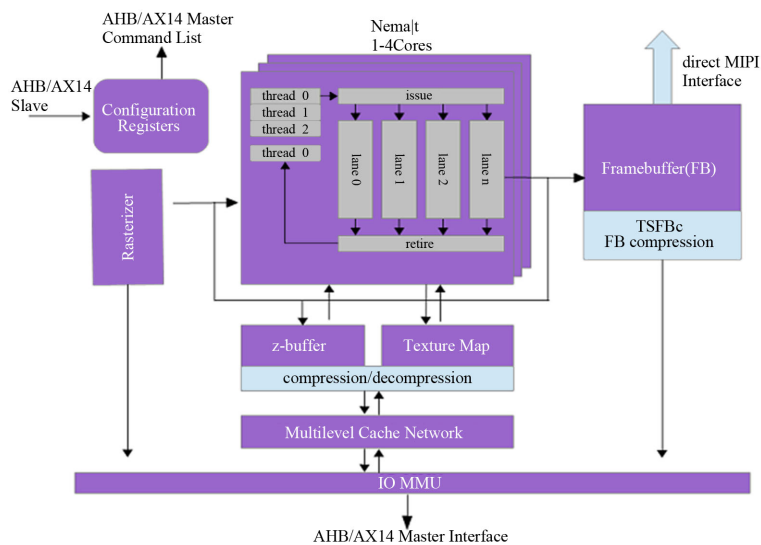


Figure 1. The internal architecture of Think Silicon NEMA|t100 GPU
 图 1. Think Silicon 公司的 NEMA|t100 可穿戴 GPU 架构

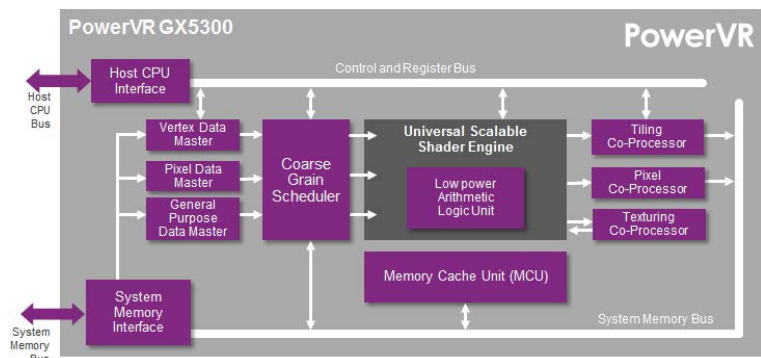


Figure 2. The internal architecture of Imagination PowerVR GX5300 GPU
 图 2. Imagination 公司的 PowerVR GX5300 可穿戴 GPU 架构

艺实现，其中 GCN Nano Lite 面积仅为 0.3 mm²，漏电功耗仅为 0.3 mW。该处理器主要针对硅面积、低功耗、存储容量、驱动尺寸、带宽减少以及显示控制配合上，进行了大量优化。

2.2.5. 小结

考虑到可穿戴 GPU 应用低功耗、小面积、高性价比等特点，可穿戴 GPU 芯片设计工艺目前一般采用 28 nm 成熟工艺节点实现，如表 1 所示。由表 1 可知，可穿戴 GPU 需要支持 OpenGL ES 2.0 图形标准，为了适合可穿戴设备应用，可穿戴 GPU 芯片面积要小，功耗要低，GPU 运行频率较低，同时需要使用各种图形数据压缩技术，可减少 GPU 内部存储面积和总线带宽，降低整体系统功耗。

3. 可穿戴设备的 GPU 发展趋势和面临的问题

与传统的桌面 GPU 和手机 GPU 相比，可穿戴 GPU 有着不同的设计目标。前者以渲染高质量、绚丽的图形和游戏为目的，而后者则以可穿戴设备的人机界面为中心，需要一种小面积 GPU 体系结构的支持，这是目前可穿戴 GPU 研究的一个主要方向。因为可穿戴设备一般采用电池供电，其应用环境要求一次充电后，要长时间工作，而图形显示的能源消耗占设备功耗的主要部分，设计有效的策略延长设备工作时间成为了可穿戴 GPU 的核心问题，这其中包括了 GPU 数据流加速算法、减少存储和存储访问的数据压

Table 1. Wearable GPU performance analysis

表 1. 可穿戴 GPU 性能比较

GPU 型号	发布时间	工艺	面积(mm ²)	漏电功耗(mW)	运行频率(MHz)
NEMA t100	2015.6.4	28 nm	0.1	0.07	400
NEMA t200	2015.6.4	28 nm	0.15		400
NEMA t400	2015.6.4	28 nm	0.25		400
DMP ANT	2014.6.31	28 nm	0.25		
GX5300	2014.7	28 nm	0.5		250
GCNano Lite	2014.6.2	28 nm	0.3	0.3	100~200
GCNano	2014.6.2	28 nm	0.6	0.5	200
GCNano Ultra	2014.6.2	28 nm	1	0.9	200~400

FLOPS	像素填充率(MPixel/s)	支持的图形 API	数据压缩
3.2 G	400	OpenGL ES 2.0	显示缓冲器、纹理、Zbuffer 压缩
6.4 G	800	OpenGL ES 2.0	显示缓冲器、纹理、Zbuffer 压缩
12.8 G	1600	OpenGL ES 2.0	显示缓冲器、纹理、Zbuffer 压缩
1 G	250	OpenGL ES 2.0	PVRTC 纹理压缩
	VGA/WVGA, >60Hz	Vector Graphic, Nano UI	
	VGA/WVGA, >60Hz	OpenGL ES 2.0, Nano GL	
	>720p, >60Hz	OpenGL ES 2.0, Nano GL	

缩算法, 以及芯片的集成电路低功耗设计方法和热量管理机制和策略, 它们构成了可穿戴 GPU 的第二个方面的主要研究内容。此外, 一些与应用密切相关的共性技术也是可穿戴 GPU 基础研究的重要组成部分, 其中包括实时响应的图形渲染场景切换算法, 可穿戴 GPU 渲染图形数据的安全性的研究等。本项目中, 我们拟对可穿戴 GPU 中的体系结构、低功耗算法和预估策略、图形渲染实时响应以及可穿戴 GPU 的渲染流水线数据安全展开研究, 因为这些都是可穿戴 GPU 的核心支撑技术。

3.1. 可穿戴 GPU 渲染架构

目前两种主流的 GPU 渲染架构是区块渲染(tile based rendering, TBR) [14]模式和立即渲染(immediately mode rendering, IMR)模式。其中 TBR 将屏幕划分为多个方块, 如 32×32 像素, 然后分别渲染每个像素块, 该方式可以减少与存储器的数据访问, 但是需要提前建立区块表和相关数据库信息; 而传统的 IMR 模式中, GPU 接受到命令就立即执行, 同时 IMR 也可以采用多种图形数据流加速方法, 如隐藏面消除、Early-Z 等技术, 加速流水线渲染速度, 使芯片功耗和带宽降低。TBR 与传统 IMR 有很多区别的, 但是 IMR 也同样能高效完成图形渲染, 所以在实际的可穿戴 GPU 图形架构设计中, 该如何选择哪种模式进行图形渲染, 如何优化处理器体系结构, 使 GPU 面积最小, 使之符合可穿戴设备应用, 这些都是可穿戴 GPU 设计首要考虑的问题。

3.2. 可穿戴 GPU 提高图形渲染性能和降低功耗的算法

采用电池供电可穿戴设备, 提高图形渲染性能和降低功耗的算法与策略是首要研究问题。可穿戴 GPU 渲染过程中减少不必要的图形渲染, 可以降低总体计算量, 进一步减少功耗, 这些算法包括深度剔除[15]、

包围盒剔除、三角形延迟渲染[16]、可编程的剔除运算单元[17]等算法。可穿戴 GPU 提高图形性能和降低功耗的另一个关键是减少内部运算部件和内存之间的存取操作，因为内存带宽功耗比计算功耗增加更迅速[18]。与传统的桌面 GPU 和手机使用的移动 GPU 相比，前者由于渲染目标不同，更注重纹理压缩[19]技术的应用以减少与内存之间的存取操作，而后者除了纹理压缩算法，还需要研究设计显示缓冲区、深度缓冲区[20]和模版缓冲区的对称压缩算法。纹理压缩算法因为应用关系，可以尝试复杂的芯片片外压缩算法，然后在片内实现快速解压，所以称纹理压缩算法为非对称的压缩算法；而缓冲区压缩算法需要在芯片内部完成快速压缩、快速还原图形的算法，所以缓冲区压缩算法称为对称压缩算法。这些压缩算法不仅可以减少运算部件和内存之间的存取数据总量，同时加快图形渲染速度，还可以减少存储访问引起的功耗，并显著的减少外部 DDR 存储容量，有助于降低系统物料(bill of material, BOM)成本，这对可穿戴设备至关重要，因为更小的设备尺寸意味着更低的成本和更低的功耗。上述可穿戴 GPU 的算法研究，不仅可以减少 GPU 渲染数据流总量、降低与外部存储的数据交互量，还可以减少 GPU 功耗，并减少 BOM 成本，因此这些算法对可穿戴 GPU 的超低功耗设计、实现有重要的意义。

3.3. 可穿戴 GPU 能量和热量管理策略

虽然可以采用多种集成电路低功耗设计方法，降低系统功耗，减少发热，但是可穿戴 GPU 与桌面 GPU 应用环境有显著的不同，可穿戴 GPU 通常是在一个密封的、没有专用散热器件的环境下使用，所以可穿戴 GPU 需要解决的另外一个主要问题就是芯片发热控制[21] [22]。芯片过热会造成两个非常不利的影响。首先，芯片的漏电功耗(Leakage Power)会随温度升高快速增长，造成芯片功耗过高。其次，芯片温度超过设计阈值会导致芯片出错，进而导致可穿戴设备关机或死机。因而可穿戴 GPU 需要进行发热检测或发热预估，当发现过热现象已经发生或者将要发生时，进行功耗干预措施。所以发热监控要与动态电压和频率调节(dynamic voltage and frequency scaling, DVFS) [23]机制类似，需要增加芯片的温度传感器，根据温度和图形流水线负载，预估各个模块发热贡献，综合采用 DVFS 和电源门控功耗(Power Gating) [24]策略，以保证芯片在可接受的温度范围内正常工作。因此，GPU 的能量和发热管理策略的研究对可穿戴 GPU 应用至关重要。

3.4. 可穿戴 GPU 渲染场景快速切换

图形处理管线包含了诸多硬件加速器和可编程染色器，图形渲染硬件流水线往往有几百级长度，每一级都含有许多需要保存的状态信息。上下文转换对于流水线而言是非常复杂和费时的。对于可穿戴 GPU 来说，上下文切换缓慢将是一个噩梦，因为用户需要实时的信息反馈到屏幕上，而由于内部切换过慢，导致信息反馈不及时，对某些应用场合将是致命的。因此如何高效的进行图形上下文场景快速切换，也将是可穿戴图形应用领域不可缺少的关键机制。

3.5. 可穿戴 GPU 渲染数据安全性

随着可穿戴的普及，所有的智能设备通过网络连接在一起。个人隐私数据如何在网络下受到保护，则成为了可穿戴设备发展需要着重考虑的问题。在一个可穿戴设备应用的 SoC 中，GPU 和操作系统如何保护图形流水线数据的可靠性，则成为一个崭新的研究方向。为了使数据更加安全，可以采用至少两级的认证、加密、网络安全认证以及可信区域授权等，这些技术目前已经非常成熟，但可穿戴 GPU 数据安全不是要取代任何现有的协议，而是要增强整个图形渲染过程的数据安全性，给用户更加安全的使用环境。与 GPU 相关的隐私数据包括：任何可视的在 GPU 缓冲区渲染的安全信息，如数字、密码、受保护的视屏、用户界面中的安全区域，以及在系统存储器、高速缓冲器的 GPU 数据和 GPU 的虚拟化数据。这些数据如何在操作系统中划分安全和非安全区域，如何在 GPU 内部进行保护，这些都是需要研究探索

的问题。针对这些问题, 促使图形开放标准组织 Khronos Group 在嵌入式系统图形库(Embedded systems graphics library, EGL)标准接口中提出了一个新的 EXT_protected_surface [25]扩展功能, GPU 硬件设计厂商通过该接口, 可以自行设计硬件配合该扩展功能使用, 使操作系统、驱动层和 GPU 硬件建立可信数据通道, 保护图形数据安全性。纵观数据信息安全领域, GPU 的数据安全并没有被关注过, GPU 只是被认为渲染逼真的 3D 图形或执行计算任务的图形/图像处理器。但展望未来, 随着可穿戴设备大量普及, 可穿戴 GPU 保护隐私安全的图形数据机制, 将成为可穿戴图形安全应用的基本保障。

综上所述, 可穿戴 GPU 中图形处理器架构、图形数据流加速和压缩、低功耗和发热策略, 以及图形实时响应和数据安全都具有现实的理论意义和应用价值, 但因可穿戴 GPU 独特的技术要求和制约因素, 使得研究和设计一款适合可穿戴应用的、小面积的、超低功耗的、BOM 成本低的、高可靠性的、高实时响应的 GPU 面临不小的挑战。

4. 总结

总之, 尽管在近一年时间内, 对这一崭新课题的研究已经有了部分成果, Think Silicon、Imagination、Vivante 和 DMP 公司推出了针对可穿戴应用的 GPU, 但或多或少的存在一些缺陷与不足, 主要包括图形运算开销大、能耗高、可穿戴图形应用特征不明显等, 因此目前关于这一课题的研究基本上还处于初步的探索阶段, 要最终形成成熟的应用技术, 还可能需要更具创新性的研究, 或者大量细致的完善工作。在国内外可穿戴图形应用研究日渐重视的背景下, 适时启动可穿戴 GPU 的体系结构研究、图形数据流加速、低功耗和发热预估策略, 以及图形快速响应和数据安全等关键支撑技术的研究课题是必要而且迫切的。

基金项目

国家自然科学基金重点项目(No. 61136002)。

参考文献 (References)

- [1] Nardone, M. and Silva, V. (2015) Discovering android wear. Pro android games. Apress, New York, 339-368. http://dx.doi.org/10.1007/978-1-4842-0587-7_11
- [2] Rawassizadeh, R., Price, B.A. and Petre, M. (2014) Wearables: Has the age of smartwatches finally arrived. *Communications of the ACM*, **58**, 45-47. <http://dx.doi.org/10.1145/2629633>
- [3] 李涛, 杨婷, 易学渊, 等 (2014) 萤火虫 2: 一种多态并行机的硬件体系结构. *计算机工程与科学*, **2**, 191-200.
- [4] 杜慧敏, 王涌钢, 李涛 (2015) 集成萤火虫 2 号 GPU 的片上系统设计和实现. *西安邮电大学学报*, **3**, 39-43.
- [5] Khronos Group (2015) OpenCL 1.X [OL]. <http://www.khronos.org/opencv/>
- [6] Khronos Group (2015) OpenVX 1.0 specification [OL]. <https://www.khronos.org/openvx/>
- [7] 韩俊刚, 蒋林, 杜慧敏, 等 (2010) 一种图形加速器和着色器的体系结构. *计算机辅助设计与图形学学报*, **3**, 363-372.
- [8] 邓军勇, 李涛, 蒋林, 等 (2014) MIGPU-9 多核交互式图形处理器的设计. *计算机辅助设计与图形学学报*, **9**, 1468-1478.
- [9] 焦继业 (2013) 低功耗高性能移动图形顶点处理器设计关键技术研究. 西安电子科技大学, 西安.
- [10] Think Silicon (2015) NEMAt100 [OL]. http://www.think-silicon.com/docs/PR_TSI_NEMA_t_June_4_2015.pdf
- [11] DMP. GPU IP Core “ant”. <http://www.design-reuse.com/news/35168/dmp-gpu-ip-core-wearables-iot.html>
- [12] Imagination. Power VR GX5300. <http://blog.imgtec.com/powervr/powervr-gx5300-the-worlds-smallest-gpu-for-next-generation-wearables-and-iot>
- [13] Vivanti. G Cnano. <http://www.vivantecorp.com/index.php/cn/media-article/news/281-20140602-vivante-intros-worlds-first-complete-line-of-gpus-for-wearables-and-iot.html>

-
- [14] Wong, T.H. (2010) A comparison of graphics performance of tile based and traditional rendering architectures. Royal Melbourne Institute of Technology, Melbourne.
- [15] Akenine-Moller, T. and Strom, J. (2003) Graphics for the Masses: A hardware rasterization architecture for mobile phones. *ACM Transactions on Graphics*, **22**, 801-808. <http://dx.doi.org/10.1145/882262.882348>
- [16] Aila, T., Miettinen, V. and Nordlund, P. (2003) Delay streams for graphics hardware. *ACM Transactions on Graphics*, **22**, 792-800. <http://dx.doi.org/10.1145/882262.882347>
- [17] Hasselgren, J. and Akenine-Moler, T. (2007) PCU: The programmable culling unit. *ACM Transactions on Graphics*, **26**, Article No. 92. <http://dx.doi.org/10.1145/1276377.1276492>
- [18] Owens, J. (2005) Streaming architectures and technology trends. In: *Proceedings of ACM SIGGRAPH 2005 Courses*, ACM Press, New York, 457-470.
- [19] Strom, J. and Akenine-Moller, T. (2005) iPACKMAN: High-quality, low-complexity texture compression for mobile phones. In: *Proceedings of ACM SIGGRAPH/EUROGRAPHICS Conference on Graphics Hardware*, ACM Press, New York, 63-70.
- [20] Hasselgren, J. and Akenine-Moler, T. (2006) Efficient depth buffer compression. In: *Proceedings of the 21st ACM SIGGRAPH/Eurographics Symposium on Graphics Hardware*, ACM Press, New York, 103-110.
- [21] Al Maashri, A., Sun, G., Dong, X., et al. (2009) 3D GPU architecture using cache stacking: Performance, cost, power and thermal analysis. In: *Proceedings of IEEE International Conference on Computer Design*, IEEE Computer Society Press, Los Alamitos, 254-259.
- [22] Sheaffer, J.W., Skadron, K. and Luebke, D.P. (2005) Studying thermal management for graphics-processor architectures. In: *Proceedings of IEEE International Symposium on Performance Analysis of Systems and Software*, IEEE Computer Society Press, Los Alamitos, 54-65.
- [23] Mochocki, B.C., Lahiri, K., Cadambi, S., et al. (2006) Signature-based workload estimation for mobile 3D graphics. In: *Proceedings of the 43rd annual Design Automation Conference*, ACM Press, New York, 592-597.
- [24] Wang, P.H., Yang, C.L., Chen, Y.M., et al. (2011) Power gating strategies on GPUs. *ACM Transactions on Architecture and Code Optimization (TACO)*, **8**, Article No. 13.
- [25] Khronos Group (2014) EGL_EXT_protected_surface. https://www.khronos.org/registry/egl/extensions/EXT/EGL_EXT_protected_surface.txt