

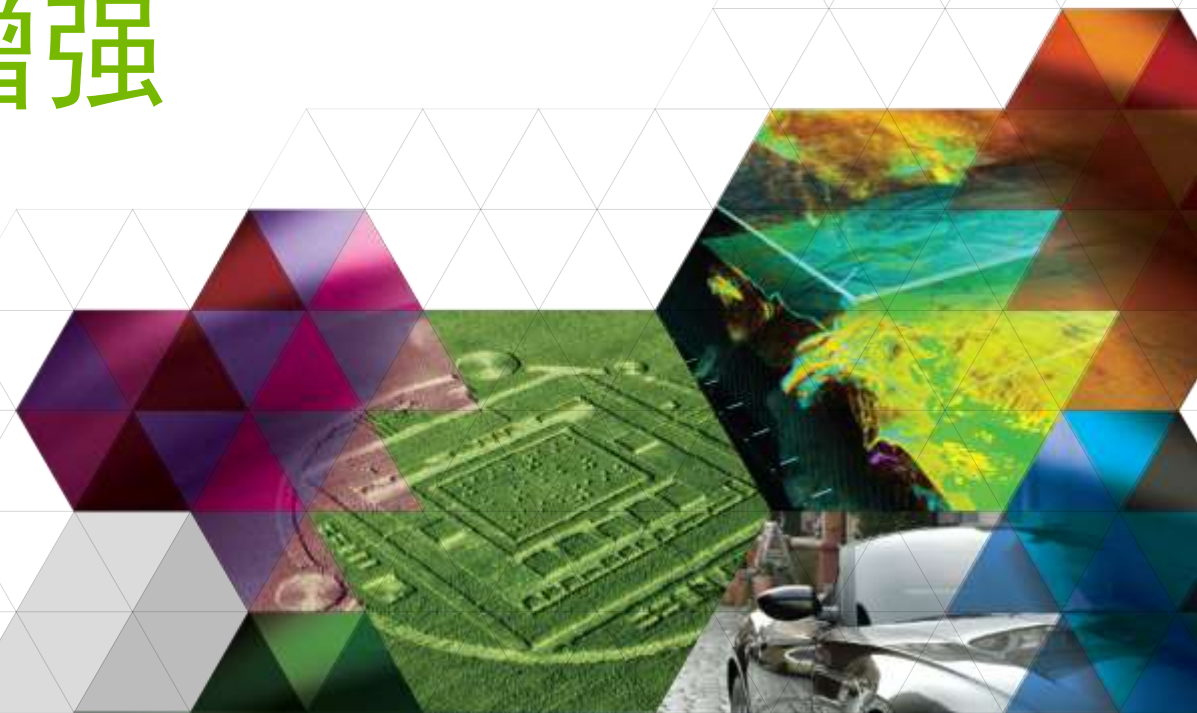
CGDC 2014



基于TEGRA K1的次世代渲染 技术及差异化增强

Jack He

Tegra Devtech



概述

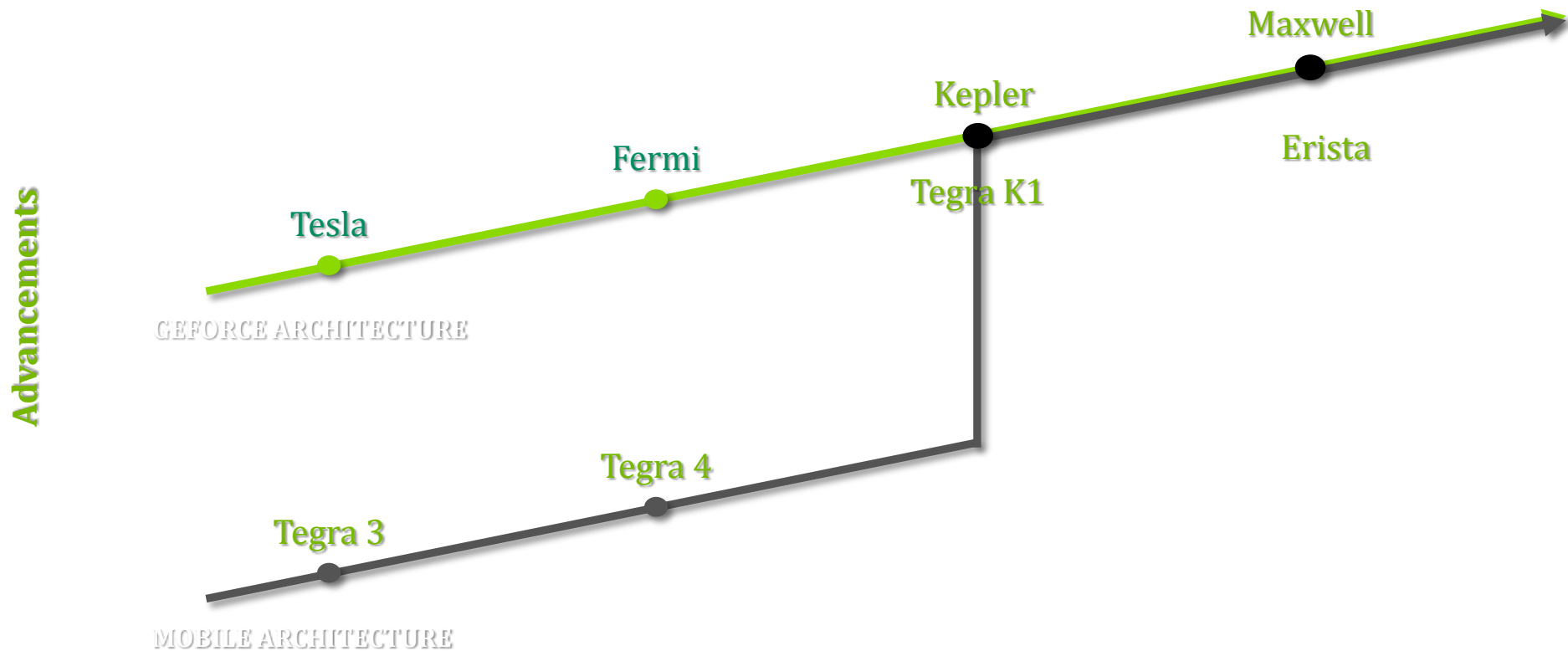
- 安卓市场的现状和 Google 应用市场
- NVIDIA 技术
 - Tegra K1 和 Kepler GPU 解析
- 基于K1的安卓差异化增强及次世代渲染技术
 - HDR多重渲染目标以及延迟渲染
 - 基于物理的渲染
 - 电影化的后处理效果
- NVIDIA GameWorks
 - 效果库，工具 & 示例程

安卓平台和应用市场 2014

- 当今最流行的移动OS
- 超过10亿的装机量 (截止到 2013九月)
- 以每天1百万的速度在增加
- 覆盖了超过190个国家
- Play Store 有超过1百万的应用和游戏



移动和桌面系统的构架的统一



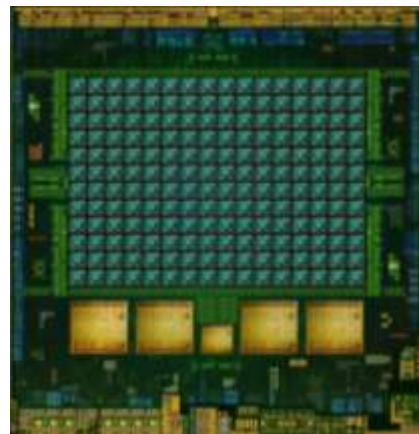
NVIDIA: 驱动移动渲染技术的进步

- 视觉计算的领跑者
- CES 2014 上发布了最新一代的移动处理器
- NVIDIA Tegra K1
 - 四核 A15
 - 192 核 GPU

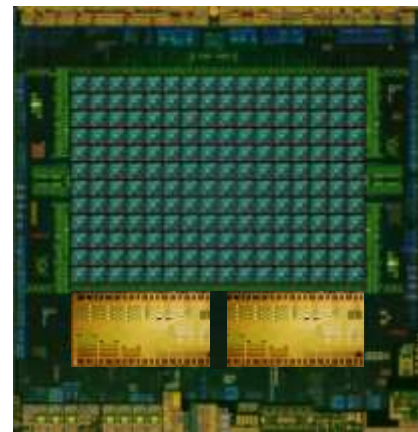


TEGRA K1

一种芯片 — 两个版本



针脚兼容



四核

32-位

3-路 超标量

上至 2.3GHz

32K+32K L1缓存

超级双核

64-位

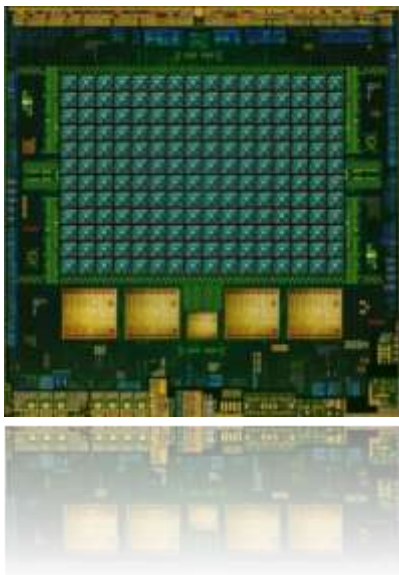
7-路 超标量

上至 2.5GHz

128K+64K L1缓存

TEGRA K1

掌中的家用机



GPU 特性

GPU 计算能力

CPU计算能力

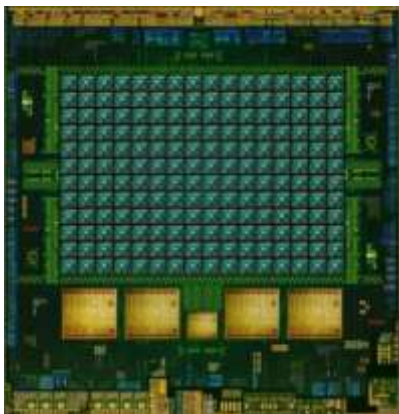
功耗

	Xbox 360	PS3	TEGRA K1
GPU 特性	DX9	DX9	DX11
GPU 计算能力	240	192	365
CPU计算能力	3600	1200 ¹	5600
功耗	100W	100W	5W

GPU Horsepower: Peak fragment shader GFLOPS
 CPU Horsepower: SPECint #s estimated for consoles +/- 20%.
 (1) PS3 GPU: Cell 154 GFLOPS FP32 not included, CPU: SPU's not included.

TEGRA K1

掌中的家用机



GPU 特性

GPU 计算能力

CPU计算能力

功耗

	PS Vita	TEGRA K1	PS4	Erista
GPU 特性	DX9	DX11	DX11	DX12
GPU 计算能力	45	365	1800	512
CPU计算能力	1000	5600	7800	7300
功耗	5W	5W	150W	5W

GPU Horsepower: Peak fragment shader GFLOPS
CPU Horsepower: SPECInt 2k * # cores

1996



2004



100 倍

2014 +



8,000 倍



UNREAL
ENGINE

Next-Gen Gaming is Photo Real

GEOMETRY SHADER

HDR

COMPUTE SHADER

MRT

FUNDAMENTAL GPU TECHNOLOGIES

TEGRA K1：差异化方案

- 动态 & 过程化的内容
- CUDA / OpenGL compute / Android RenderScript
 - 全部由GPU加速
- 图形渲染的增强
 - OpenGL ES 2, 3 & 3.1 和完整的 OpenGL 4.4支持
 - 强大的顶点处理能力允许更多的多边形数量
 - 更高的纹理精度(up to 16k x 16k); ASTC 1 & 2 bit
 - 更多的内存容量以及更高的显示分辨
 - 令人眩目的粒子效果
 - 使用曲面细分增加细节

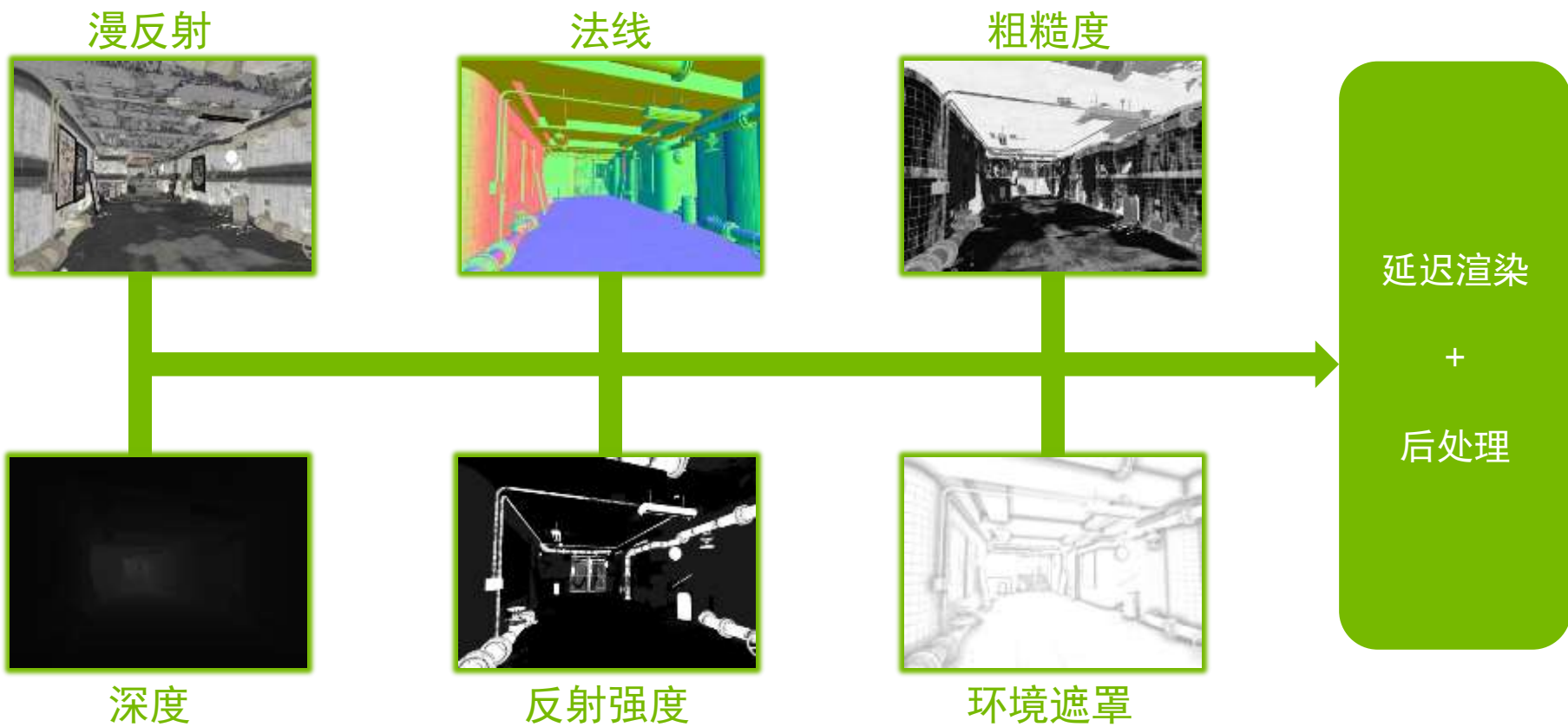
移动平台全新的渲染技术

- 多重渲染目标 (MRT), 高动态范围 (HDR) 及延迟渲染技术
- 基于物理的渲染 (PBR)
- 电影化的后处理特效及景深和散景效果
- Unity 5 在GDC公布的特性
 - PBR shaders
 - 实时全局光照
- Unreal Engine 4在GDC公布的特性
 - 官方对Android 系统的支持
 - 在Tegra K1上, 对 MRT & PBR 渲染管线有着完整的支持

多重渲染目标（MRT）

- 一次性写入多个目标缓冲
- 最初由 OpenGL 2.0 和 Direct3D 9引入
- OpenGL ES 3.0的新特性

延迟渲染技术

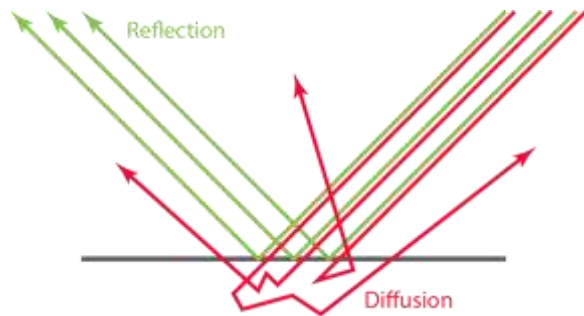


CGDC 2014



基于物理的渲染-1 (PBR)

- 可以被分为 漫射和反射两大类



- 反射

- 镜面反射, 光泽度
- 各向同性, 各向异性



使用偏振滤镜
捕捉的照片

漫射



+

+

镜面反射



||

||

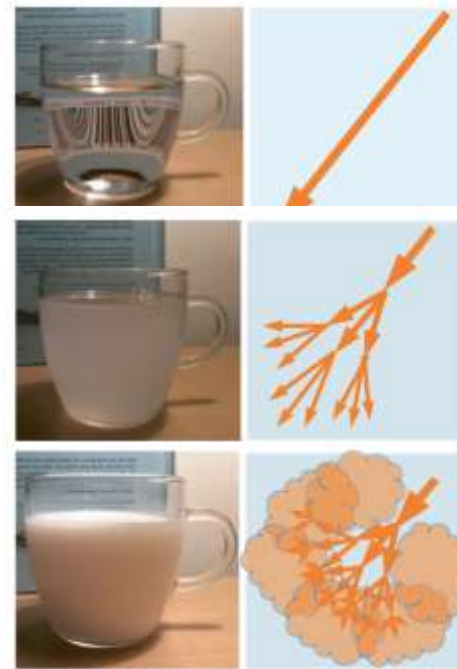
原图



基于物理的渲染-2

■ 半透明物体

- 吸收， 散射
- 折射， 色散



基于物理的渲染-3

■ 能量守恒

— 反射和漫射



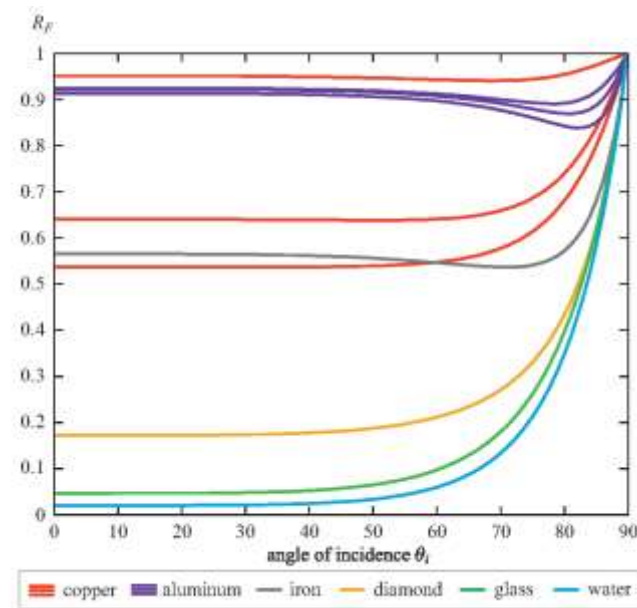
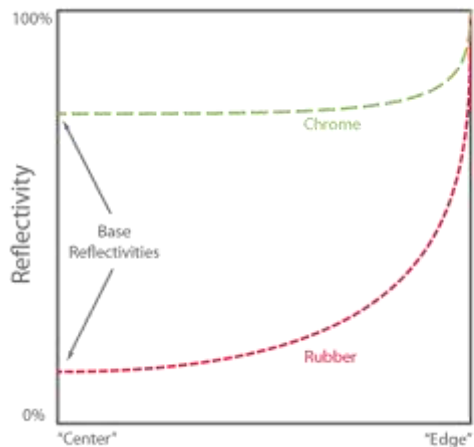
— 光泽度和反射强度



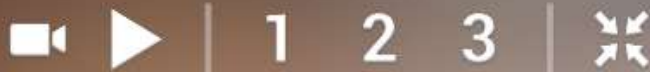
基于物理的渲染-4

■ 菲涅耳反射(Fresnel reflection)

- 不同入射角度的反射率不同
- 平行入射角度达到最大反射率

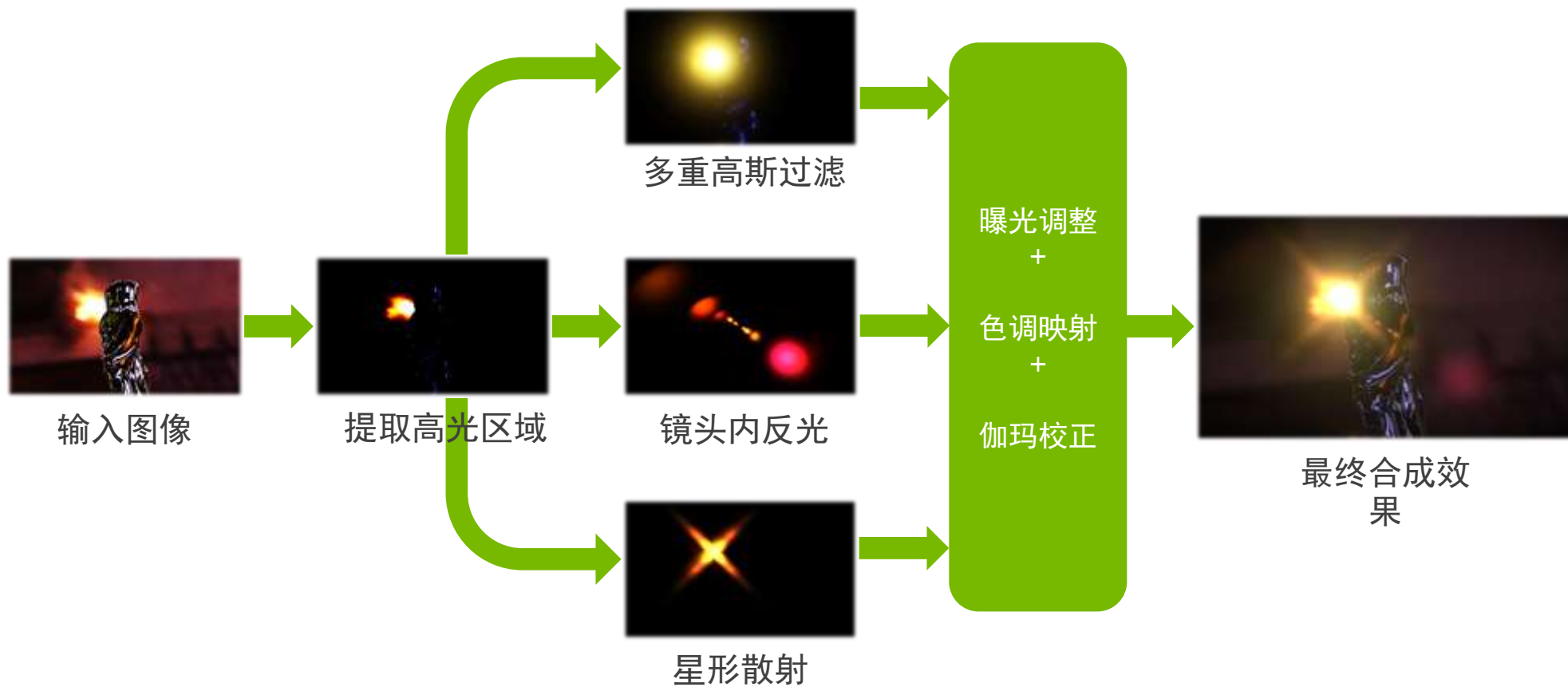


CGDC 2014



电影化的后处理效果

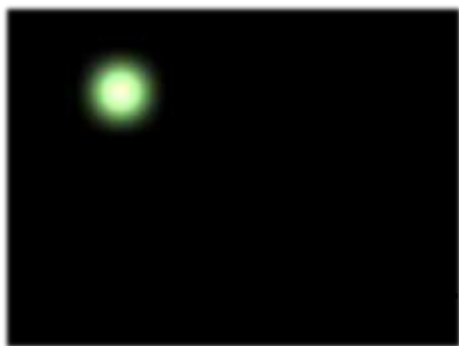
后处理流程图



多重高斯过滤(1)

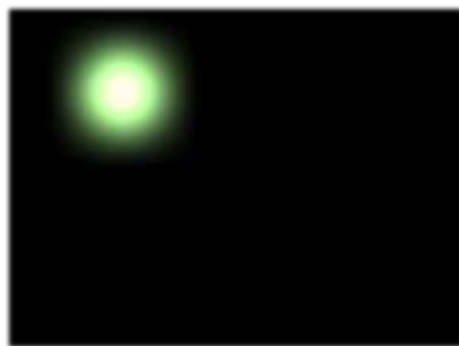
- 大的采样半径非常消耗资源
- 利用缩小后的缓冲区
 - 大的采样半径意味着强力的低通过率
 - 在低分辨率的图像上应用一个模糊滤镜到并使用双线性放大 → 瑕疵几乎不可察觉
 - 通过改变过滤图像的分辨率而不是过滤半径
 - $1/4 \times 1/4$ (1/16 资源消耗)
 - $1/8 \times 1/8$ (1/64资源消耗)
 - $1/16 \times 1/16$ (1/256资源消耗)
 - $1/32 \times 1/32$ (1/1024资源消耗)
 - ...
 - 即使是很大范围的过滤半径在很小的图片尺寸上应用也是非常快速的

多重高斯过滤(2)



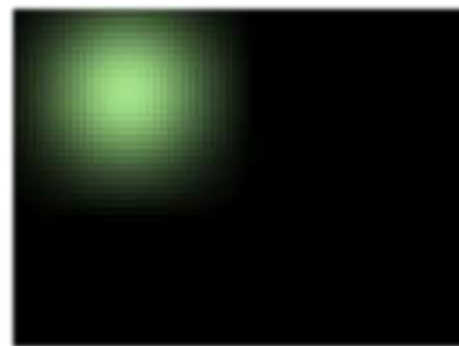
$1/4 \times 1/4$

+

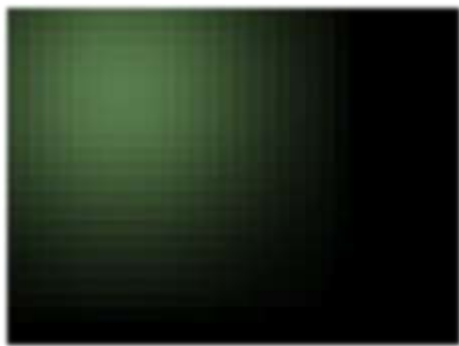


$1/8 \times 1/8$

+



$1/16 \times 1/16$



$1/32 \times 1/32$

+



$1/64 \times 1/64$

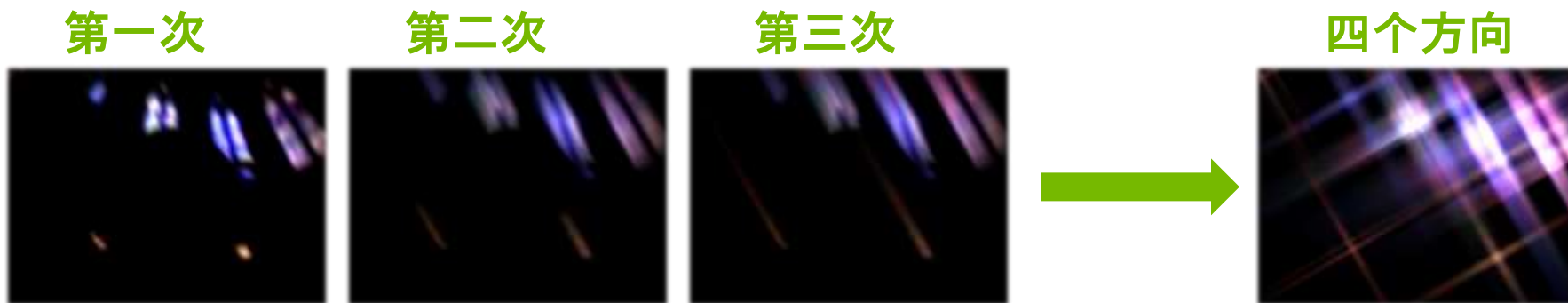
=



最终结果, 非常好!

散射光束模拟

- 可根据需求选择2, 4, 6 或 8 方向上的



- n^{th} pass

- $\text{Texcoord}[s] = \text{rendering point} + b*s$
- $\text{color weight}[s] = a^{(b*s)}$
- $b = 4^{(n-1)}$

- 调制颜色来模拟色散

s : 采样位置
a : 颜色调制

镜头内反光的模拟(1)

- $\text{texcoord} = (\text{original texcoord} - 0.5) * s + 0.5$

s: 一个任意的缩放因子

e.g. 缩放因子 $s = -2.0$

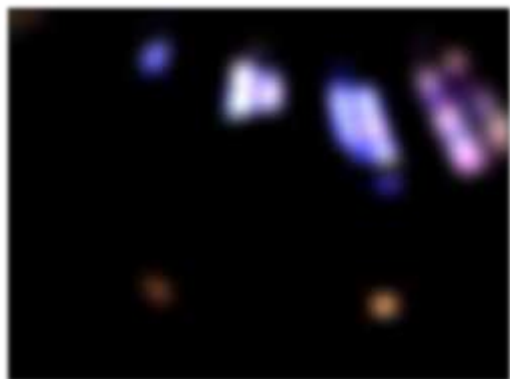
原始纹理坐标

(0,0)

(1,0)

(-0.5,-0.5)

(1.5,-0.5)



缩放因子
 $s = -2.0$



(0,1)

(1,1)

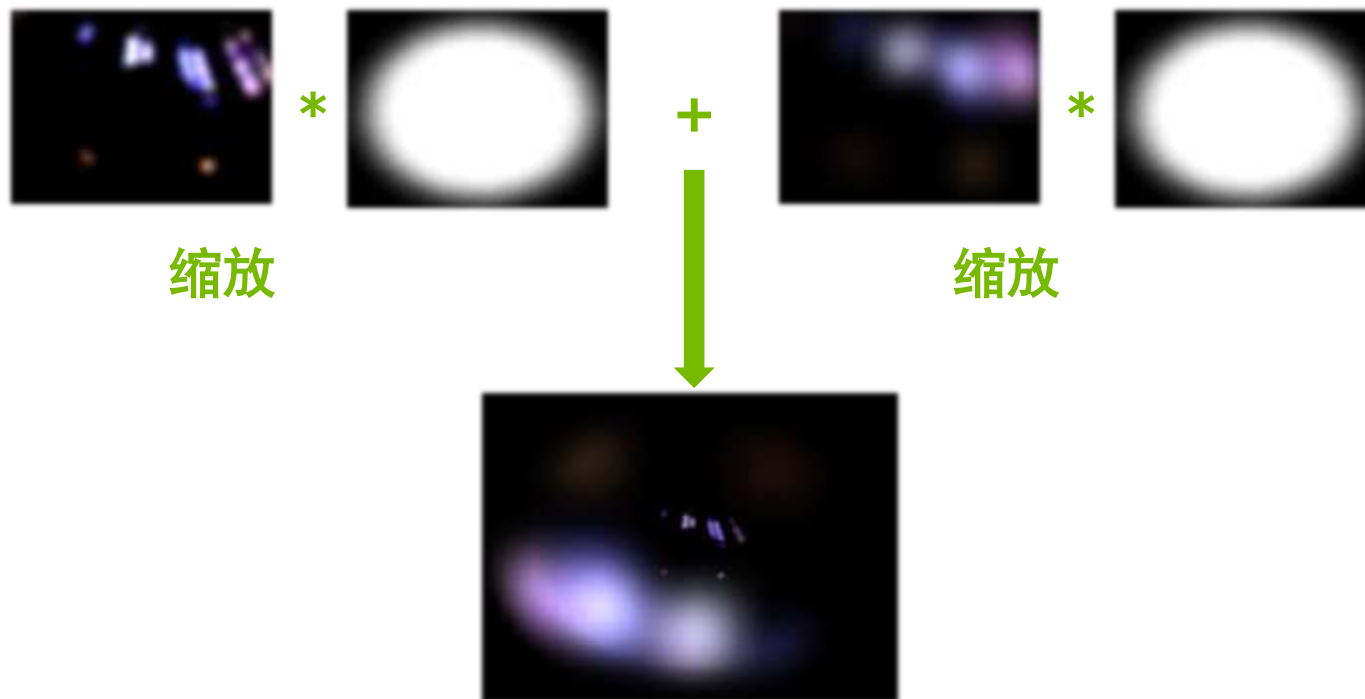
(-0.5,1.5)

(1.5,1.5)



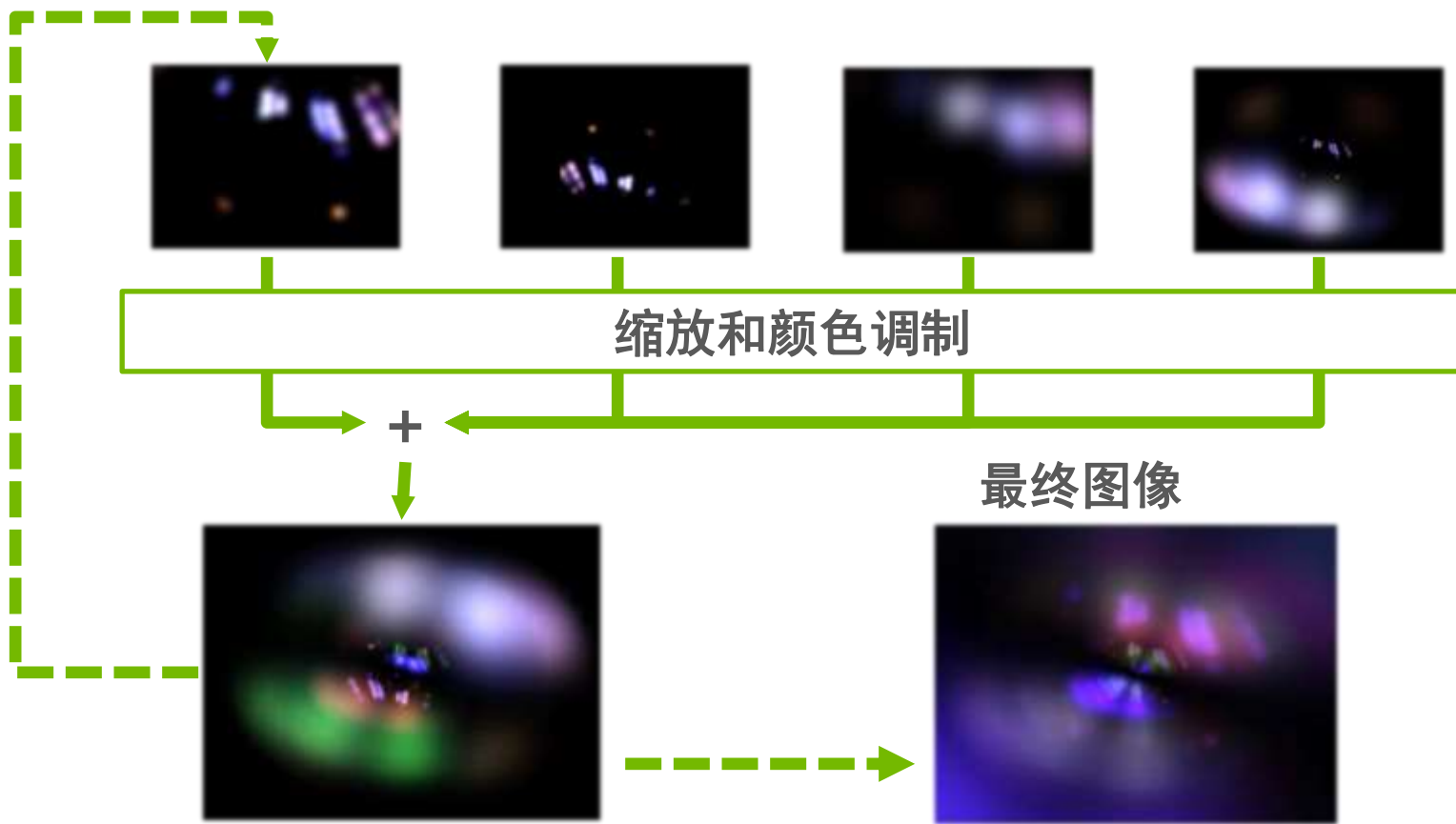
镜头内反光的模拟(2)

- 利用一个光滑的圆作为图像遮罩
 - 防止图像边缘出现棱角



镜头内反光的模拟 (3)

- 应用多次缩放和颜色调制



曝光控制流程

- 获得场景的平均亮度

$$Lum_{avg} = \exp\left(\frac{1}{N} \sum_{x,y} \log(\delta + Lum(x, y))\right)$$

- 计算合适的曝光值

$$Exp = \frac{Scale}{Lum_{avg}}$$

- 模拟自适应渐变曝光过程

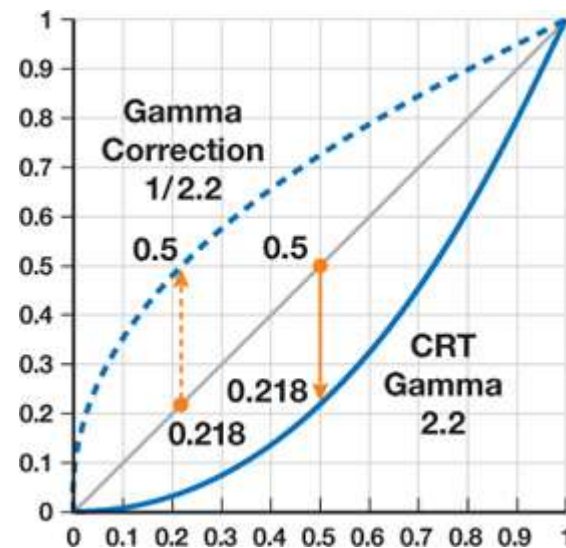
$$Exp_{new} = Exp_{current} + (Exp - Exp_{current}) \times Ratio$$

高动态范围渲染中的伽玛校正问题

- 伽玛校正是高质量的渲染中不可缺少的环节
- 在非线性空间进行运算会在各个环节导致问题：光照计算，Mipmap，纹理过滤...
- sRGB 对应伽玛 2.2
- sRGB 纹理，帧缓冲
- OpenGL 对sRGB的扩展支持

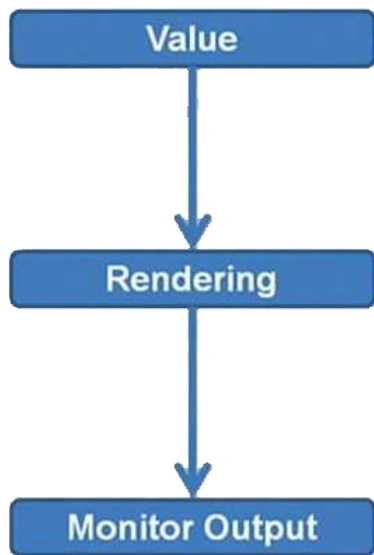
http://www.nvidia.com/dev_content/nvopenglspecs/GL_EXT_texture_sRGB.txt

http://www.opengl.org/registry/specs/EXT/framebuffer_sRGB.txt



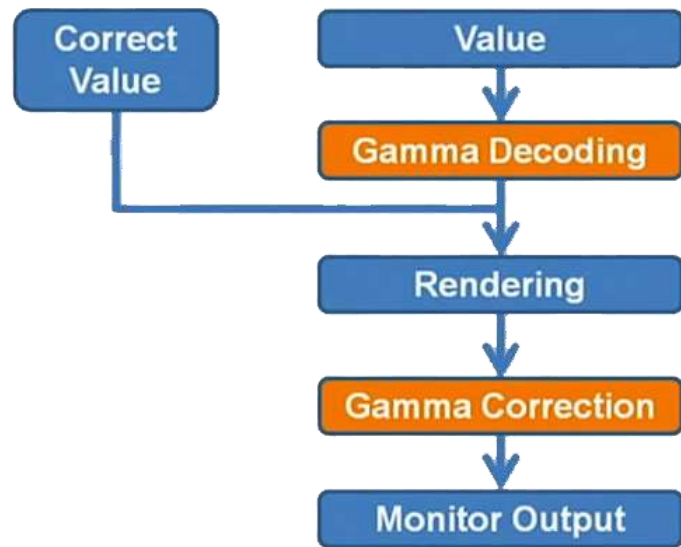
线性处理流程

非线性处理流程



非线性处理结果

线性处理流程



参考图像照片



线性处理结果

例子

- Tegra K1上可达 60FPS, @1080p

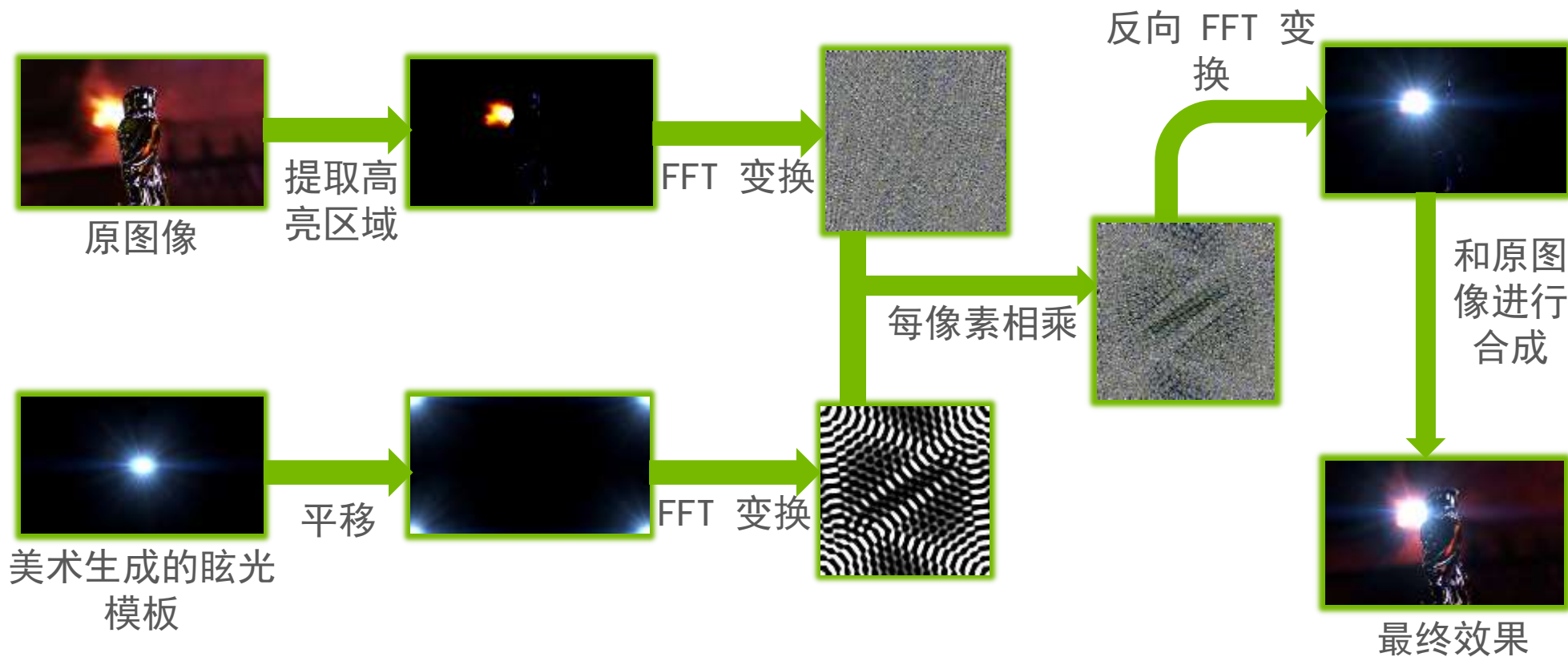


该图在NVIDIA GameWorks 示例程序中截取

另一种全新的后处理方法

- 使用快速傅里叶变换(FFT)
- Compute Shader
- 优势
 - 一次处理可得到的后处理效果
 - 处理时间恒定且和后处理效果复杂度无关
 - 美术可指定处理版蒙版

FFT 后处理流程



CGDC 2014

D3D11 544.21 fps Vsync off (1200x800), R8G8B8A8_UNORM (MS1, Q0)
HARDWARE: NVIDIA GeForce GTX TITAN Z
Time: 1.84 ms

Toggle full screen

Toggle REF (F3)

Change device (F2)

Diffusion + FFT Bokeh

Scene: endless_city

FarBlurSize: 15.00

NearBlurSize: 0.00

FocalRadius: 50.00

PlaneInFocus: 45.0

MinLuminance: 0.33

MaxLuminance: 1.00

Bokeh Intensity: 66.00

Bokeh Sub Fac: 0.00

BokehSize: 32.00

☒ Heptagon Bokeh

Press Esc to exit flight mode



突破Windows和Android
游戏体验极限



GAMEWORKS RESEARCH

300位世界级工程师研发团队，追求艺术与科技的完美结合



GAMEWORKS LIBRARY

视觉效果和物理模拟开发包
包括技术，算法，引擎和运行库



DEVELOPER TOOLS

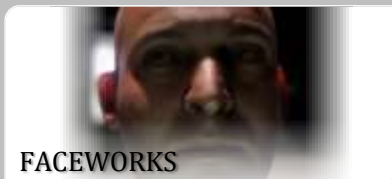
独立的或者集成到IDE的开发工具
包括调试、性能分析等实用工具

NVIDIA GAMEWORKS™

PhysX



VisualFX



OptiX



Samples



<https://developer.nvidia.com/gameworks>

NVIDIA开发者工具 TADP

BUILD. DEBUG. PROFILE.

Microsoft®
DirectX

OpenGL

nVIDIA
CUDA

OpenGL ES



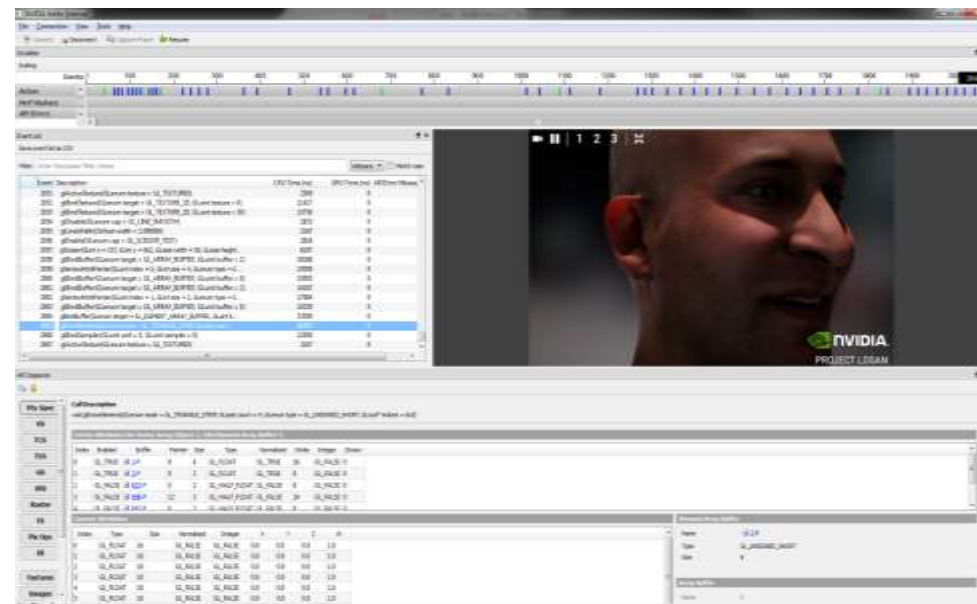
IDE INTEGRATION

Visual Studio

STANDALONE TOOLS



HARDWARE SUPPORT
CPU AND GPU DEBUGGING/PROFILING



<https://developer.nvidia.com/tadp>

总结

- Android 和 Play Store 的现状
 - 众多平台中脱颖而出
- NVIDIA Technology 将会带来
 - 游戏专用的平台 & 极为便捷稳定的开发环境
 - 画质的显著增强: Tegra K1 配合全新的渲染技术使得PC等级的画质在移动平台成为可能
- NVIDIA GameWorks 工具, 示例程序 & 效果库
 - <http://developer.nvidia.com/gameworks>

谢谢！

问题？

jackh@nvidia.com