2013
Electronics and
Telecommunications
Trends

모바일 GPU 동향

Trends of Mobile GPU

임베디드 소프트웨어 &
시스템반도체 기술 특집

한진호 (J.H. Han) 멀티미디어프로세서연구실 선임연구원
변경진 (J.G. Byun) 멀티미디어프로세서연구실 실장
엄낙웅 (N.W. Eum) 시스템반도체연구부 부장

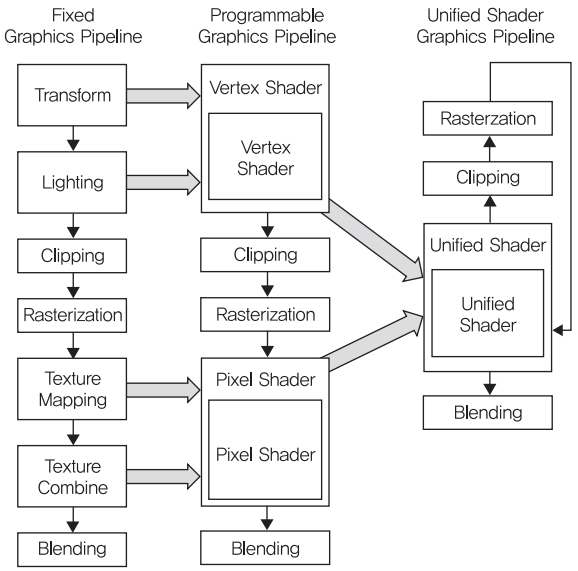
스마트폰 및 태블릿 PC에 들어가는 핵심 부품인 AP(Application Processor)는 모두 GPU(Graphics Processing Unit)를 내장하고 있다. 이는 칩 면적의 제약과 사용 가능한 전력의 한계로 데스크톱의 그래픽 카드에 탑재된 고성능 GPU와는 다른 설계 제약을 받는다. 본고에서는 고성능 GPU와 다른 설계 조건을 갖는 mobile GPU 기술에 대해서 알아보았고 대표적인 commercial mobile GPU인 Imagination, ARM, Qualcomm, NVidia사의 mobile GPU의 특징 및 성능에 대해서 알아보았다.

Ⅰ. Mobile GPU

Mobile GPU(Graphics Processing Unit)라 함은 스마트폰 및 태블릿 PC 등과 같은 핸드헬드(handheld) 제품에서 사용되는 AP(Application Processor) 칩에 내장되는 GPU를 칭하며 GPU만을 칩으로 제작되는 것과 비교하여 전력 소모와 집적 면적의 제약으로 기존의 하이엔드(high end)나 데스크톱 제품과 비교하여 전력 효율성이 대두되는 GPU이다. 이를 위해 기존의 GPU architecture의 차이점 및 이를 위한 mobile GPU 기술 그리고, 현존하는 mobile GPU의 기술 수준을 알아보고 앞으로 Mobile GPU가 어떻게 변화할지를 알아보겠다.

1. GPU Architecture

GPU는 크게 geometry, rasterization, composite로 나뉘어 구성이 된다. Geometry는 애플리케이션에서 graphics API를 call하게 되면 3D 공간 데이터를 2D 공간 데이터로 전환하는 기하학(geometry) 처리를 하게 된다. 그렇게 구해진 2D 공간 데이터로부터 영상을 구성하는 조각(fragment, 결국 화면에서의 pixel)의 색상, 화면 깊이 등의 정보를 계산한다. 그런 다음에 composite에서 각 조각의 표면 무늬(texture)를 입히게 된다.



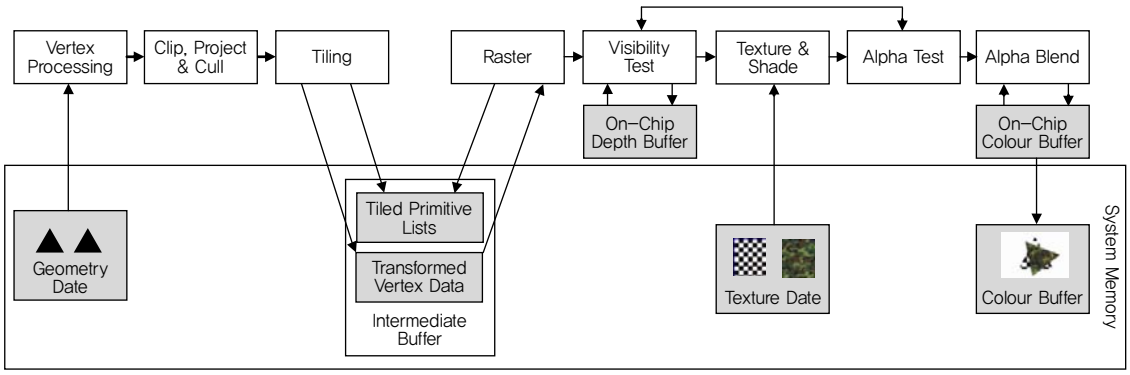
(그림 1) GPU Pipeline 진화[1]

이러한 GPU의 rendering pipeline은 하드웨어 블록을 고려하여 (그림 1)과 같이 변천하였다.

Fixed graphics pipeline 구조에서는 geometry 기능을 transform, lighting, clipping으로 구분하였으며 composite를 texture mapping, texture combine, blending으로 구분하였으며 후에 transform, lighting, texture mapping, texture combine 부분은 고정된 하드웨어보다는 programmable한 vertex shader, pixel shader로 구성한 programmable graphics pipeline을 실현하여 좀 더 다양한 3D 효과를 표현할 수 있도록 하여 현실감 있는 이미지를 표현할 수 있었다. 그 후에 vertex shader와 pixel shader의 기능을 구분할 필요 없이 두 기능을 모두 수행할 수 있는 unified shader를 두어 vertex 연산과 pixel 연산을 같은 programmable logic에서 수행하도록 하였다. 이로 인해 같은 수의 shader를 장착 시 좀 더 많은 shader를 동시에 이용할 수 있어 영상의 품질을 높일 수 있게 되었다.

2. Mobile GPU

Image Technology사의 PowerVR MBX는 mobile GPU의 표준처럼 되어 있다. 초기 PowerVR 제품군은 데스크톱 PC 시장에서 3dfx와 같은 기존 회사의 그래픽 카드보다 더 나은 가격대 성능비로 경쟁하는 제품이었다. PC 그래픽 카드 시장이 OpenGL과 Direct3D 등장 이후로 빠르게 변화하면서, PowerVR과 같은 소규모 업체는 시장에서 퇴출되었다. 이후 PowerVR은 노트북 컴퓨터를 대상으로 한 저전력 그래픽 카드로 방향을 전환하였다. 시간이 지나면서 임베디드 시스템, 핸드헬드 장치, 스마트폰에 사용되는 SoC에 사용될 수 있도록 변화하였다. PowerVR GPU는 PowerVR에서 직접 제조하지 않으며, TI, 인텔, NEC, 삼성, ST마이크로일렉트로닉스, 프리스케일, 애플, NXP 등 반도체 제조 회사에 반



(그림 2) Tile Based Rendering Pipeline[2]

도체 설계 IP로써 집적이 되도록 라이선스가 되었다. 이렇게 시작된 mobile GPU의 구조는 저전력과 제한된 리소스를 사용할 수 있도록 설계가 되었다. 이러한 PowerVR 기반으로 적용한 GPU 기술들을 알아보겠다.

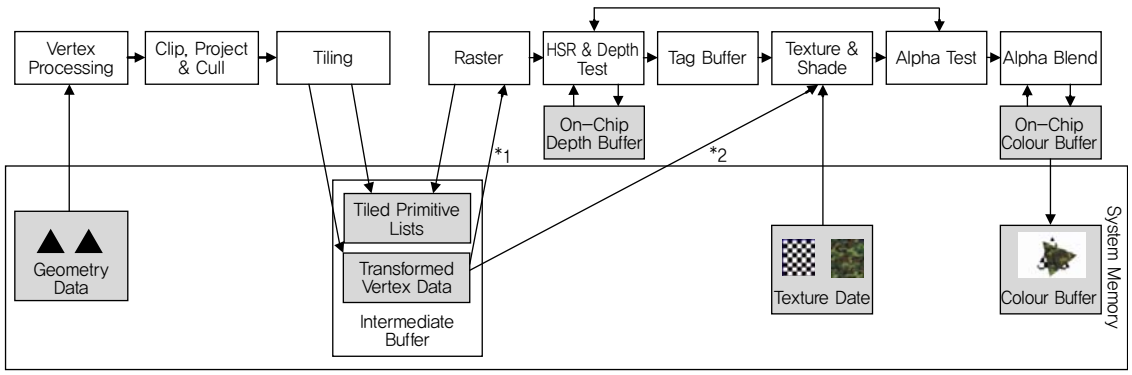
II. Mobile GPU 기술

1. Tiled Based Rendering(TBR)

기존의 GPU의 rendering pipeline인 Immediate Mode Rendering(IMR)과 비교하여 rasterization 전에 tiling 이라는 작업이 들어간다. 이는 프레임을 tile로 분할하여 처리할 vertex 정보를 tile 단위로 분리하게 된다. Tile의 경계에 겹치는 폴리곤(polygon)은 각각의 tile에 포함되는 더 작은 폴리곤으로 나뉘게 된다. 이렇게 tile 단위로 나뉜 정보는 intermediate buffer에 저장이 되고 이후의 rasterization 이하의 작업은 tile 단위로 처리되게 된다. 이렇게 되면 tile 단위로 vertex 정보를 처리하는 작업과 tile 단위 처리 후에 나뉘어진 tile을 모아 프레임을 구성해야 작업이 추가되지만 시스템 메모리에서 읽어 와야 하는 정보는 tile 단위로 줄어들게 되어 모든 정보를 on-chip memory에 둘 수가 있어 시스템 메모리로의 잦은 read-write-modify 액세스가 줄어들게 되는 장점이 있다. 또한 rasterization 이후로는 tile 간의 독립성으로 tile 단위로 병렬처리가 가능하다. 이는 폴리곤 단위 병렬보다 시스템 메모리로 인한 병목 현상이 없기에 병렬처리 효율성을 높일 수 있다((그림 2) 참조).

2. Tile Based Deferred Rendering(TBDR)

폴리곤을 생성하는 응용 프로그램이 드라이버에 삼각형 정보를 전달해 주면, 드라이버에서는 메모리에 연속된 삼각형이나 인덱스 형식으로 저장한다. TBDR은 다른 아키텍처와는 다르게, 현재 비디오 프레임에 사용되는 모든 폴리콘 정보가 도착하기 전까지 렌더링이 실행되지 않는다. 또한, 가능한 경우 각각 픽셀에서 어떤 표면이 보이는지를 결정하기 전까지 텍스처와 셰이딩 작업이 지연된다. 이를 deferred rendering이라 부르는 이유이다. TBR에서 전체 프레임은 작은 사각형 그리드로 나뉘며, 각각의 그리드는 타일로 취급된다. 각각 타일에는 그 타일 내에서 보이는 삼각형 정보가 연결되며, 타일별 렌더링된 이미지가 합쳐져서 전체 이미지가 나타난다. TBR만에서는 각각 픽셀은 카메라에서 가장 가까운 삼각형에 의해서 그려진다. (그림 3)의 TBDR pipeline에서는 개별 타일 행과 연결된 다각형의 깊이 정보를 계산한다. 그리고 안 보이는 폴리곤을 그리지 않아도 된다는 장점이 있다. 또한 Early-Z technique과 다르게 폴리곤 처리 순서와는 관계 없이 반투명 폴리곤(translucent polygon)의 투명도를 올바르게 처리할 수 있다. 한 번에 렌더링이 한 타일만 이루어지기 때문에



(그림 3) Tile Based Deferred Rendering Pipeline[3]

전체 타일을 더 빠른 칩 메모리에 저장해 두고 다음 타일을 처리하기 전에 비디오 메모리에 복사할 수 있다. 일반적인 경우 개별 타일은 프레임당 한 번만 처리된다. 이러한 기술은 고성능 GPU에서도 연구되었다. 이전에 마이크로소프트의 탤리스먼 프로젝트, 기가픽셀사의 타일 기반 3D 그래픽 기술 또한 유사한 기술이었으나 3dfx에 인수되었고 이후 엔비디아에 인수되어서 사용되고 있지 않다. 인텔은 x86 CPU 칩에 사용되는 내장 그래픽에서 비슷한 개념인 존 렌더링을 사용하고 있으나, 이후에 설명할 HSR(Hidden Surface Removal) 및 지연된 텍스처 작업(deferred texturing)을 완벽하게 실행하지 않는다.

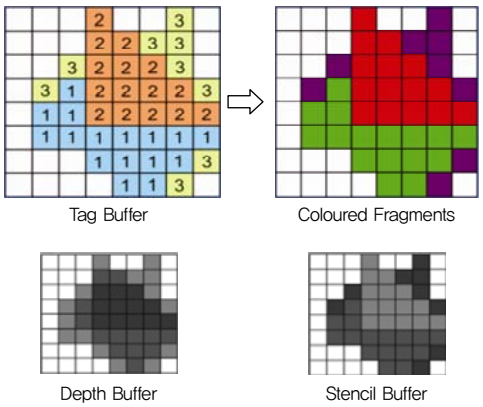
이러한 처리는 불투명 부분에 대한 처리를 하지 않기에 intermediate buffer와 texture buffer에서의 정보를 읽어 오는 것을 감소시킬 수 있다.

또한 해당 정보처리를 하지 않기에 workload 또한 줄일 수 있다. 다만, HSR 처리를 해줘야 하며 HSR 결과물을 저장하기 위해 (그림 3)에서처럼 tag buffer가 필요하게 된다. Tag buffer는 각 픽셀이 어떤 폴리곤에 의해 표현되고 있는지 즉, 각 픽셀 단위마다 눈에 보이는 색상 정보를 얻을 수 있는 폴리곤 Index를 갖고 있게 된다.

TDR과 다르게 (그림 3)의 '*1'번 화살표 끝의 raster는 단지 새로운 폴리곤의 깊이 정보만을 읽어 와서 on-chip depth buffer에 저장된 이미 읽어온 tile 내의 픽셀에 대한 깊이 정보와 비교하여 눈에 보일 수 있다면 새로운 폴리곤 tag를 tag buffer에 갱신한다. 그리고, 폴리곤의 색상에 대한 정보를 읽는 것은 미루어진다. 그리고, texture & shade에서 tag buffer의 정보를 활용하여 (그림 3)의 '*2'와 같이 픽셀 단위로 화면에 보이는 부분의 색상 정보를 갖고 있는 폴리곤을 판단하여 해당 픽셀의 색상 정보만 읽어 오게 된다.

(그림 4)에서 depth buffer의 정보를 이용하여 왼쪽 상단의 tag buffer를 구성하였고 새로운 폴리곤의 색상 정보를 이용하여 색상 정보를 담고 있는 stencil buffer는 갱신이 되어 오른쪽 상단의 tile의 색상을 구성하고 있으며 보이지 않는 픽셀에 대한 처리를 하지 않는다.

HSR은 고성능 GPU에서 채택하고 있는 Early-Z technique과 비교하여 겹치는 object 또한 뒤에 있는 부분은 구분하여 처리를 하지 않는다.



(그림 4) Hidden Space Removal 동작[4]

이는 object 단위가 아닌 픽셀 단위로 미리 깊이를 비교하기 때문이다. 이를 위해 tag buffer를 구비하고 이렇게 함으로써 오직 화면에 보이는 픽셀 정보만을 읽어와서 texture & shade 이후 과정을 하게 된다.

현재 Imagination사의 PowerVR MBX Series부터 이러한 TBDR 구조를 채택하고 있다.

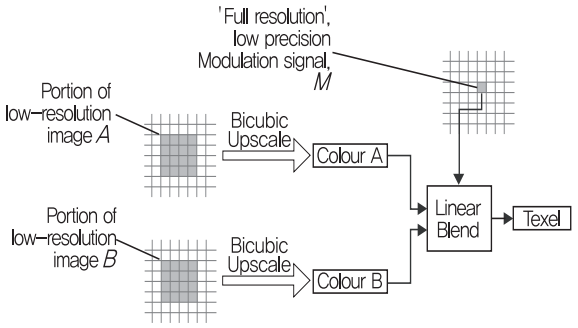
3. Texture Compression Using Low-Frequency Signal Modulation(PVRTC)

Catmull[5]에 의해 소개된 이래로 3D graphics에서는 현실감을 위해 texture mapping을 사용하고 있다. 그러나 이는 3D 장면을 렌더링하는 동안은 계속해서 texture data를 읽어야 하기에 내부 RAM 사용을 증가시키고 memory bus의 사용률을 높인다. 그래서, 이러한 memory bus의 bandwidth 제약으로 texture cache를 사용하고 있다. 하지만 이 두 가지 문제점을 해결하기 위해 mobile GPU에서는 texture compression을 사용하고 있으며 다음의 사항이 고려되어 설계가 되었다.

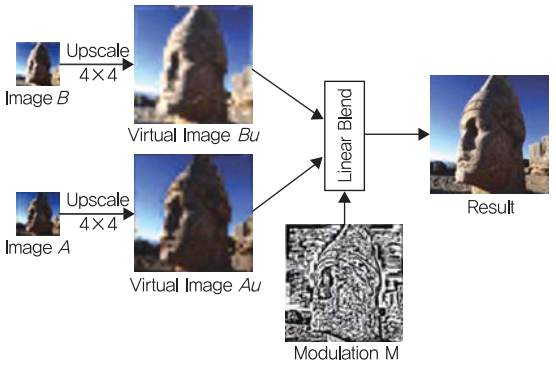
- Decoding speed
- Random access
- Compression rate and visual quality
- Encoding speed

한 예로 JPEG 압축 방식은 random access를 허용하지 않는다.

이러한 네 가지 사항을 고려한 texture compression 알고리즘이다. 그 방식은 다음과 같다. 텍스처는 하나의 object에 대한 표면을 이미지로 갖고 있는 것으로 object의 크기가 다양해 질 수 있기에 다양한 배율의 이미지를 갖고 있어야 한다. 이를 mipmap 방식이라고 하며 다양한 배율 이미지에 대한 압축을 고려해야 한다. 그래서 (그림 5), (그림 6)은 두 개의 낮은 해상도의 이미지 A, B 및 최고 해상도에 대한 낮은 정확도의 모듈레이션 정



(그림 5) PVRTC 2bpp/4bpp복원 과정[6]



(그림 6) PVRTC 기본 방식[6]

보를 이용하여 texel을 decompression하는 방식이다.

즉, 다른 크기의 이미지 A, B를 최고 해상도로 upscale하고 이를 다시 modulation 정보 M을 이용하여 linear blend를 하여 원하는 해상도의 texel을 복원한다.

PVRTC4bpp와 PVRTC2bpp의 두 가지 방식이 있으며 이들 각각은 픽셀당 4bit, 2bit의 텍스처 정보로 압축할 수 있는 기술이다. 이는 픽셀 텍스처당 32bit 정보로 이루어진 텍스처와 비교해 8배, 16배의 압축률을 보이는 것으로 텍스처의 품질을 감소시키지 않고 시스템 메모리의 액세스를 상당히 줄일 수 있다. 또한 opaque (불투명 색 포맷으로 또는 RGB)와 translucent(반투명 색 포맷, RGBA) 텍스처를 모두 지원한다. S3TC와 같은 텍스처 압축 기술은 translucent를 지원하기 위해서는 특별한 포맷을 지원해야 하는 문제가 있다.

4. Super Sampling Full-scene Anti-aliasing (FSAA)

좋은 anti-aliasing 결과를 얻을 수 있는 하나의 방법은 super-sampling 방법을 사용한다. 이는 전체 프레임의 anti-aliasing을 할 때 본래의 해상도보다 수평 2배, 수직 2배, 또는 4배의 영상 크기에서 anti-aliasing을 수행한 후 프레임 크기로 down-sampling하는 방법이다. 이는 TBR에서는 추가적인 메모리의 증가 없이 수행이 가능하고 다면 처리할 tile 수를 줄이면 된다. 이런 super-sampled size 만큼의 프레임 메모리 및 다른 on-chip 메모리를 요구하는 IMR 방식과 비교해 많은 장점이 있다.

5. 32bpp, Zero Cost Color Operation

Color operation 시 color depth를 이용한 연산 시 IMR에서는 보통에 프레임 버퍼의 제약으로 16bit color depth를 이용한다. 이러한 경우 최종 색의 부정확성으로 banding effect가 발생하기에 dithering 작업을 해주지만 'grainy'한 결과를 야기한다. Deferred rendering에서는 오직 한번 color operation을 수행하고 그것도 32-bit color depth를 사용할 수 있을 정도의 on-chip memory 확보가 가능하다. 이렇기에 banding effect가 없이 color operation 결과물을 얻을 수 있다.

Ⅲ. 상용 Mobile GPU

1. Multi-core, Multi-execution Unit

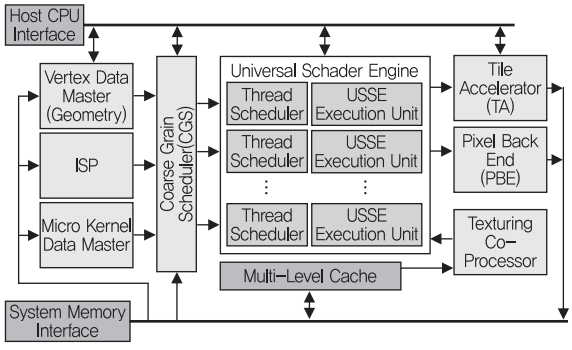
PowrVR Series 5 SGX는 USSE(Universal Scalable Shader Engine)를 장착하고 있다. 이는 GPU 구조에서 언급한 unified shader의 일종으로 TBDR에 특화되었다. 이러한 engine 내부의 thread scheduler + USSE인 execution unit을 증가시켜서 여럿 USSE를 운용할 수 있다.

또한 SGX-MP는 GPU 코어를 증가시킬 수 있으며 tile 기반의 독립적인 구조에 의해 단일 코어와 비교하여 95%의 연산속도를 높일 수 있으며 이에 따른 시스템 메모리 대역폭의 증가는 1% 이하가 된다. 또한 단일 코어에서 수행되는 응용 소프트웨어를 변경 없이 다중 GPU 코어를 구비한 환경에서 사용이 가능하다[4],[7].

2. PowerVR, Mali, ULPGeforce, Adreno 성능 비교

Mobile GPU 시장은 IP만을 공급하는 회사는 Imagination, Vivante, ARM, DMP 등이 있으며 이 가운데 Imagination의 시장 점유율은 46.6%에 달한다. 그 외에 자체 SoC에 탑재하여 판매하는 업체는 Qualcomm과 NVidia가 있다. 2012년 상반기 Mobile GPU 시장은 Qualcomm의 Adreno가 두 번째로 많이 팔아 26%의 점유율을 갖는다. Qualcomm은 스냅드래곤 및 모뎀과 통합 SoC를 공급하기에 많은 점유율을 기록하고 있다. Imagination의 PowerVR 5 Series SGX는 Apple의 모든 iPhone, iPad에 탑재되는 A5, A6에 공급되고 있다. ARM의 Mali 시장 점유율은 약 13.1%로 Samsung의 스마트폰과 태블릿 PC에 탑재되고 있다((그림 7) 참조)[5].

이들의 제품 중에 Qualcomm S2에 들어가는 Adreno 220, NVidia의 Tegra에 들어가는 ULPGeforce, Imagination의 SGX543MP, ARM의 Mali-400MP의 성능을



(그림 7) PowerVR SGX-MP 구조[4]

〈표 1〉 Mobile GPU 제품별 성능 비교[7]–[10]

제품	Fill Rate (MTri.ps)	Fill Rate (Gpixelps)	Core #	동작주파수	SoC	GPU
Adreno 220	88	2.4	Single	400MHz	Qualcomm/S2	Multi-Unit GPU
ULPGeforce	85	1.2	Single	333MHz	nVidia/Tegra	Multi-Unit GPU
SGX543MP4	35	1	Quad	200MHz	Apple/A5	Quad-Core GPU
Mali-400MP	44	1.6	Quad	400MHz	Samsung/Exynos	Quad-Core GPU

비교한 표는 〈표 1〉과 같다.

Adreno, ULPGeforce는 내부 execution unit을 확장하여 성능 증대를 하고 있으며 SGX543MP, Mali-400MP는 GPU core 수를 늘려 성능을 증대를 하고 있다. 그래서 SGX543MP, Mali-400MP는 단일 core의 성능을 제시하고 있다. Fill rate는 35~88MTri.ps, 1~2.4Gpixelps을 나타내며 주파수는 200~400MHz 수준이며 SGX543MP4가 가장 낮은 주파수를 나타내고 있다.

Ⅳ. 결론

지금까지 mobile GPU의 동향을 이해하기 위해 GPU의 rendering pipeline의 변화를 알아보았고 이러한 pipeline 구조가 mobile GPU에서는 어떻게 변화되었는지를 알기 위해 mobile GPU에 적용된 기술 등을 알아보았다. 그리고, 상용화된 mobile GPU의 구조 및 성능을 비교해 보았다.

용어해설

Texture 3D 그래픽을 폴리곤 단위로 재현한 후에 물체의 재질 및 색상을 나타내기 위한 폴리곤의 깊이와 색상 정보 외에 추가 정보

Anti-Aliasing 원근감에 따라 물체 단위로 재현되는 3D 그래픽 영상은 그 경계가 매끄럽지 않은 Aliasing 현상이 생기게 되고 이를 없애주는 작업을 Anti-Aliasing이라고 함.

RGBA 3D 그래픽 영상에서는 색상을 표현할 때는 Red, Green, Blue 외에 Alpha가 있다. 이 Alpha는 반투명 정도를 표현하는 것으로 다른 물체와 겹치게 표현될 경우 그 투과율을 의미함.

약어 정리

AP	Application Processor
FSAA	Full-scene Anti-aliasing
Gpixelps	Giga pixels per seconds
GPU	Graphics Processing Unit
HSR	Hidden Surface Removal
IMR	Immediate Mode Rendering
ISP	Image Synthesis Processor
MTri.ps	Mega Triangles per seconds
PVRTC	PowerVR Texture Compression
Raster	Rasterization
TA	Tile Accelerator
TBDR	Tile Based Deferred Rendering
TBR	Tile Based Rendering
USSE	Universal Scalable Shader Engine

참고문헌

[1] J.H. Woo, “Design Paradigm of Mobile APs,” Texas Instrument, Dallas, U.S. 2012.

[2] Imagination Technologies Ltd. PowerVR, “POWERVR MBX Technology Overview,” 2009.

[3] F. Policarpo and F. Fonseca, “Deferred Shading Tutorial,” *SBGAME*, 2006.

[4] Imagination Technologies Ltd. PowerVR, “POWERVR Series5 Graphics SGX Architecture Guide for Developers,” 2011.

[5] 존페디리서치(JPR), “ GPU 시장 현황,” 2012.

[6] S. Fenney, “Texture Compression Using Low-Frequency Signal Modulation,” *Graphics Hardware*, 2003.

[7] QdevNet Adreno User Community, “Adreno Graphics Processing Units,”.

[8] AnandTech, “Adreno 320 Performance Preview,”.
[9] Vivante Corporation Signs 15th GPU. www. vivantecorp.com
[10] T. Olson, “Mali-400 MP: A Scalable GPU for Mobile Devices,” HPG, 2010.