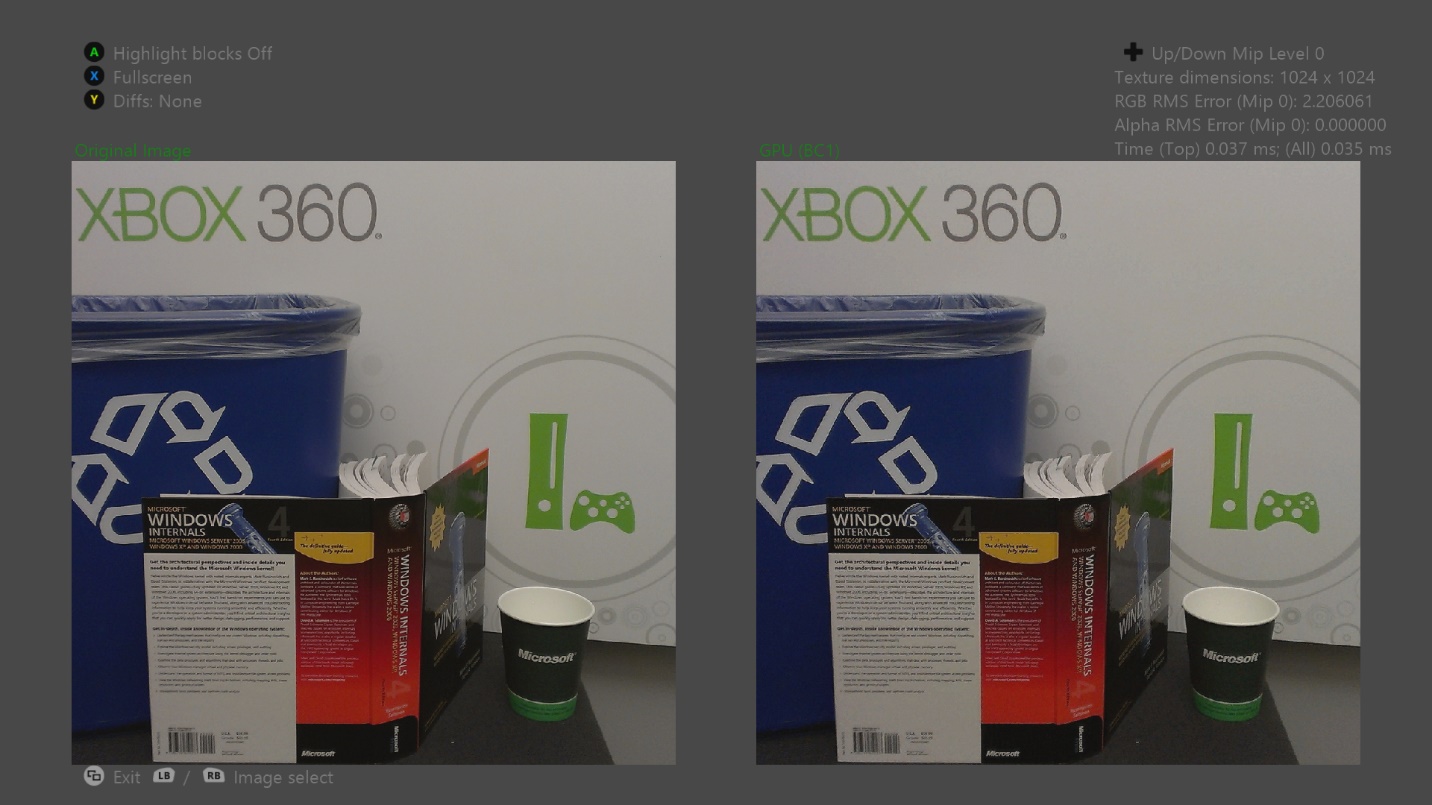


FastBlockCompress 샘플

*이 샘플은 Microsoft 게임 개발 키트 미리 보기(2019년 11월)와 호환됩니다.*

# 설명

# 이 샘플에서는 DirectCompute를 사용하여 런타임에 클래식 *빠른 블록 압축* 알고리즘을 토대로 BC1, BC3 및 BC5 형식에 대해 빠른 질감 압축을 수행하는 방법을 보여 줍니다. 또한 이 샘플에서는 런타임 및 오프라인 압축 모드 간을 토글하면서 시각적 품질을 비교할 수 있습니다.



# 샘플 빌드하기

Xbox One 개발 키트를 사용하는 경우 활성 솔루션 플랫폼을 Gaming.Xbox.XboxOne.x64로 설정하세요.

Project Scarlett을 사용하는 경우 활성 솔루션 플랫폼을 Gaming.Xbox.Scarlett.x64로 설정하세요.

*자세한 내용은 GDK 문서에서* 샘플 실행하기*를 참조하세요.*

# 샘플 사용

|  |  |
| --- | --- |
| 작업 | 게임패드 |
| 이전 또는 다음 이미지 | 왼쪽 또는 오른쪽 범퍼 |
| 이전 또는 다음 압축 방법 | D-패드 왼쪽 또는 오른쪽 |
| 이전 또는 다음 밉 수준 | D-패드 아래쪽 또는 위쪽 |
| 카메라 이동 | 오른쪽 스틱 |
| 확대/축소 | 왼쪽 또는 오른쪽 트리거 |
| 전체 화면 및 나란히 보기 | X |
| 블록 강조 표시 | A |
| 순환 차이 모드 | Y |
| 종료 | 보기 단추 |

# 배경

Xbox One는 Xbox 360에서 제공하는 512MB의 10배에 달하는 5GB의 통합 메모리를 전용 앱에 제공합니다. 안타깝게도 IO 대역폭과 저장소 미디어 용량은 여기에 미치지 못하고 있습니다. 블루레이 미디어는 49GB를 포함하며, Xbox 360 게임 디스크 버전 3에서 제공하는 7.8GB에서 6.3배 정도만 증가한 크기입니다.

스트리밍 설치가 도입된 상황과 이러한 사실을 고려한다면, 로드 시간을 최소화하고 게임 자산을 사용 가능한 저장 공간에 압축하기 위한 효과적인 압축 방법이 여전히 중요하다고 할 수 있습니다.

종종 타이틀은 오프라인 이미지 압축 형식을 사용하여 게임 질감을 인코딩함으로써 많은 양의 저장 공간을 절약할 수 있습니다. Xbox One에는 하드웨어 JPEG 디코더가 기본적으로 제공되어 있으므로 JPEG를 선택하면 유용할 수 있습니다. 그러나 JPEG 하드웨어는 질감을 메모리에 압축되지 않은 YUV 형식으로 디코딩하며, 이 형식은 렌더링에는 적합하지 않습니다. 이 방법을 사용하는 경우 타이틀은 런타임에 GPU가 지원하는 블록 압축 형식 중 하나로 텍스처를 다시 압축해야 합니다.

이 샘플은 GPU를 사용하여 질감을 BC1, BC3 및 BC5 형식으로 효과적으로 압축합니다. 오프라인 블록 압축에 사용되는 표준 알고리즘은 지금까지 너무 느려서 실시간으로 실행할 수 없었으며, 이 샘플에서 사용하는 알고리즘은 속도와 관련해서 심각한 품질 저하를 가져옵니다.

메모리 대역폭 때문에 현재 알고리즘에서 병목 상태가 발생하므로, 다른 방법을 사용하여 성능을 약간만 저하시키면서 현저한 품질 개선을 달성할 수 있을 것입니다.

# 구현 참고 사항

# 샘플의 각 DirectCompute 압축 셰이더는 세 가지 변형 형태로 제공됩니다. 즉, 단일 밉 버전, 두 밉 버전, 테일 밉 버전이 있습니다.

# 단일 밉 셰이더는 원본 질감의 단일 밉을 압축합니다.

# 두 밉 셰이더는 원본 질감의 단일 밉을 읽고, 내려가면서 LDS(로컬 데이터 저장소) 메모리까지 밉을 샘플링합니다. 그런 다음, 이 셰이더는 소스 및 아래쪽 샘플링 버전을 모두 압축하고, 해당 밉 수준을 출력 질감에 씁니다.

# 이 프로세스는 두 번째 밉 수준에 대한 소스 질감 읽기를 방지하여 메모리 대역폭을 절약합니다. 하지만 실제로는 추가된 셰이더 복잡성과 더 높아진 GPR과 LDS 사용량으로 인한 선점 효과 감소로 인해 성능 게인이 대체적으로 상쇄됩니다.

# 테일 밉 셰이더는 다양한 밉 수준에서 작동하도록 다른 스레드를 선택하여 단일 디스패치 호출에 16×16 ~ 1×1의 밉 소스 질감 수준을 압축합니다.

# 최소 wavefront 크기는 64개 스레드이기 때문에 별도의 디스패치 호출로 각 테일 밉을 압축한 기법은 사용 가능한 대부분의 스레드를 낭비할 수 있습니다. 테일 밉 셰이더는 하나의 wavefront 및 디스패치 호출만 사용하여 낭비되는 대량 작업을 방지합니다.

# Direct3D에서는 BC 형식 질감을 UAV로 바인딩하는 것을 허용하지 않으므로 계산 셰이더에서 블록 압축 질감에 직접 쓸 수 없습니다. 샘플은 쓰기 가능한 형식의 중간 질감을 블록 압축 텍스처와 동일한 메모리 위치로 앨리어싱함으로써 이 제한을 해결합니다. 중간 질감은 해당 크기의 1/4이고 각 텍셀은 압축된 질감의 블록에 해당합니다.

# 이 방식으로 텍스처 메모리를 앨리어싱하려면 두 질감의 바둑판식 모드와 메모리 레이아웃을 정확하게 일치시켜야 합니다. 또한 Direct3D는 메모리 앨리어싱을 감지하지 못하므로 GPU는 동일한 메모리 위치로 앨리어싱된 다른 리소스에서 작동하는 여러 그리기 또는 디스패치 호출을 동시에 예약할 수 있습니다.

# 즉, 중간 질감에 쓰는 셰이더는 앨리어싱된 블록 압축 질감에서 읽는 그리기 호출과 동시에 예약될 수 있습니다. 이러한 위험을 방지하려면 적절한 펜스 안에 수동으로 삽입해야 합니다.

오프라인 회선 압축 알고리즘은 [DirectXTex](https://github.com/Microsoft/DirectXTex/)에서 구현됩니다.

# 대안

이 샘플의 주요 목적은 클래식 "JPG/FBC" 솔루션을 디스크 내 질감 저장 및 런타임의 메모리 내 소비량을 최소화하는 다른 대안과 비교할 수 있는 테스트 사례를 제공하는 것입니다.

* 기반 유니버설([GitHub](https://github.com/BinomialLLC/basis_universal/)) - 이 솔루션은 런타임에 디스크의 질감을 BC7(모드 6)을 비롯한 다양한 형식으로 트랜스코딩할 수 있는 E[ETC1](https://github.com/Ericsson/ETCPACK) 변형으로 압축합니다. 이 경우 클래식 JPG/FBC 파이프라인과 비교할 때 비슷하거나 더 나은 이미지 품질을 구현하면서 보다 넓은 대상 GPU 배열이 지원될 뿐만 아니라 디스크 사용 공간이 더 줄어듭니다. .basis 의 다중 GPU 트랜스코딩 측면은 콘솔 타이틀보다 모바일에서 훨씬 더 유용하지만, 디스크 절감 측면에서도 평가할 가치가 있습니다.

# 참고 자료

Microsoft Advanced Technology Group. Fast Block Compress sample. Xbox 360 SDK February 2010.

Tranchida, Jason. [Texture Compression in Real-Time Using the GPU](http://www.gdcvault.com/play/1012554/Texture-compression-in-real-time). GDC 2010. March 2010.

van Waveren, J.M.P[. Real-Time DXT Compression](https://software.intel.com/sites/default/files/23/1d/324337_324337.pdf). Intel Software Network. May 2006.

van Waveren, J.M.P., and Castaño, Ignacio. [Real-Time YCoCg-DXT Compression](https://www.nvidia.com/object/real-time-ycocg-dxt-compression.html). NVIDIA developer site. September 2007.

van Waveren, J.M.P., and Castaño, Ignacio. [Real-Time Normal Map DXT Compression](http://developer.download.nvidia.com/whitepapers/2008/real-time-normal-map-dxt-compression.pdf). NVIDIA developer site. February 2008.

# 업데이트 기록

2019년 9월에 릴리스됨

# 개인정보처리방침

샘플을 컴파일하고 실행할 때 샘플의 사용을 추적하는 데 도움이 되도록 샘플 실행 파일의 파일 이름이 Microsoft에 전송됩니다. 이 데이터 수집을 옵트아웃하려면 Main.cpp에서 "샘플 사용 원격 분석"이라고 레이블이 지정된 코드 블록을 제거할 수 있습니다.

Microsoft의 일반 개인정보취급방침에 대한 자세한 내용은 [Microsoft 개인정보처리방침](https://privacy.microsoft.com/en-us/privacystatement/)을 참조하세요.