

텍스처 압축 관련 RnD 보고

게임제작 1 팀

박정민 전임

비고

1. 텍스처 압축 테스트 기준 내용
2. RGBA 32
3. AOS - ETC2
4. AOS – ETC2 : Down sizing
5. IOS – PVRCT : Alpha 픽셀 추가로 POT 이미지 사용
6. IOS – PVRCT : Alpha 픽셀 추가로 POT 이미지 사용, Down sizing
7. ASTC 4x4
8. ASTC 6x6
9. ASTC 8x8
10. 요약 및 의견

1. 텍스처 압축 테스트 기준 내용

- 기본 텍스처 크기 : 1080x1500
- 오픈 스펙 카드 수 : 108 장
- 업데이트 예상 스펙 카드 수 : 200 장
- RGBA 32 : 압축 없는 원본 이미지 포맷
- ETC2 : AOS 기본 압축 포맷.
 - 텍스처의 가로, 세로 픽셀 수가 4 의 배수여야 함
 - OpenGL 3.0 ES 부터 지원.
- PVRTC : IOS 기본 압축 포맷
 - 텍스처가 정사각형이어야 하며, 픽셀 수가 2 의 제곱승이어야 함
- ASTC : 가변 블록 크기로 압축.
 - 4x4 블록부터 12x12 까지 사용. 블록이 클수록 압축률이 올라감.
 - OpenGL 3.0 ES 부터 지원, IOS A8 Process 부터 지원

2. RGBA 32

- 현재 Large카드 원본 텍스처 크기 : 1080x1500
- 용량 : 6.2MB (Format : RGBA 32bit)
- 오픈 스펙 상 카드 수 : 108장
- 업데이트 예상 스펙 : 200여장
- 오픈 스펙 용량 : 670MB
- 업데이트 스펙 용량 : 1.2GB



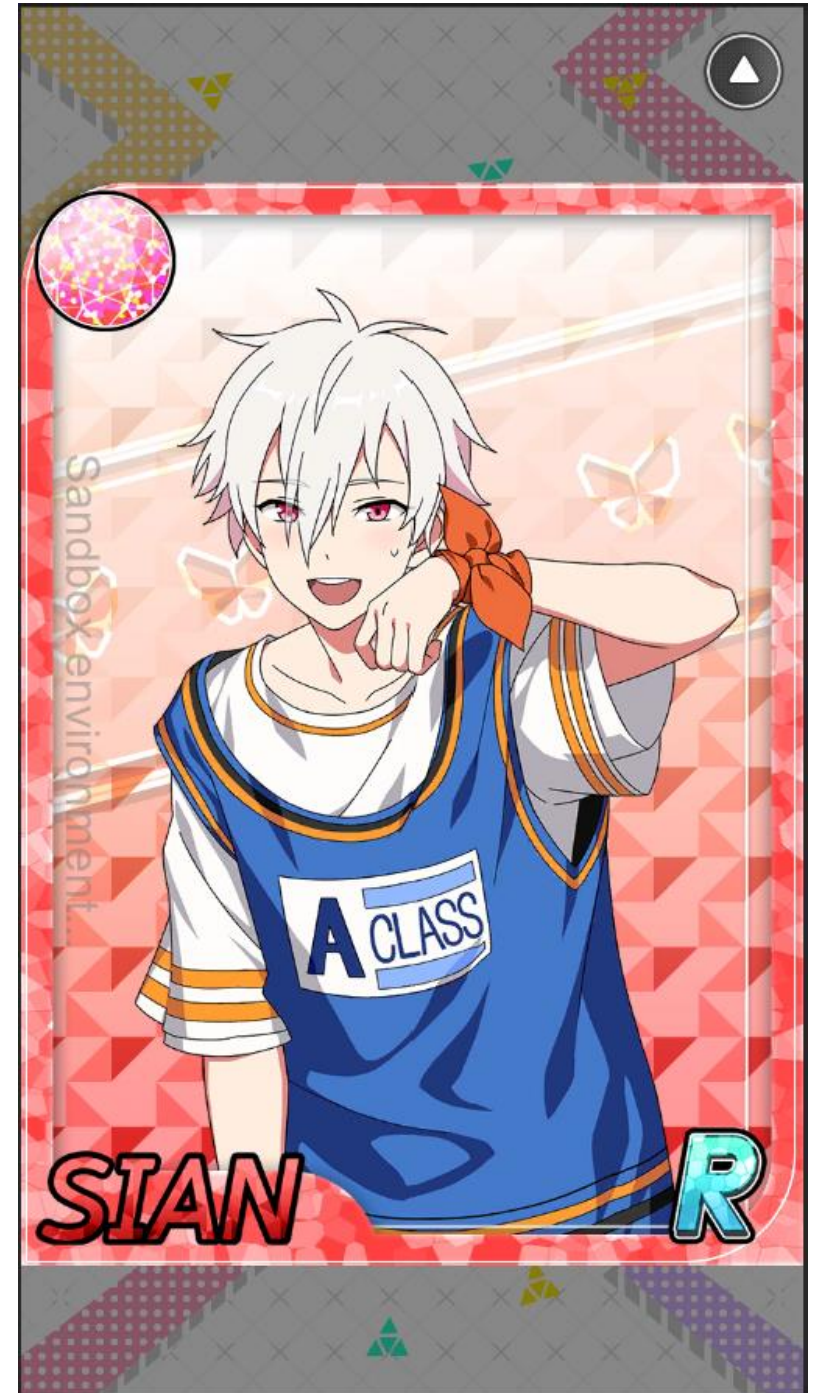
3. AOS – 크기 변경 없음, 기본 압축 적용

- 용량 : 1.5MB (Format : ETC2)
- 오픈 스펙 용량 : 162MB
- 업데이트 스펙 용량 : 300MB



4. AOS – Down sizing, 기본 압축 적용

- Down sizing : 736x1024
- 용량 : 0.7MB (Format : ETC2)
- 오픈 스펙 용량 : 75.6MB
- 업데이트 스펙 용량 : 140MB



5. IOS – 기본 압축 적용

- 2048x2048 이미지에 원본을 넣고 남은 영역을 Alpha처리
- 용량 : 2MB (Format : PVRTC)
- 오픈 스펙 용량 : 216MB
- 업데이트 스펙 용량 : 400MB



6. IOS – Down sizing, 기본 압축 적용

- 1024x1024 이미지에 736x1024를 넣고 남은 영역을 Alpha처리
- 용량 : 0.5MB (Format : PVRTC)
- 오픈 스펙 용량 : 54MB
- 업데이트 스펙 용량 : 100MB



7. ASTC 압축 4x4

- 용량 : 1.5MB
- 오픈 스펙 용량 : 162MB
- 업데이트 스펙 용량 : 300MB



8. ASTC 압축 6x6

- 용량 : 0.7MB
- 오픈 스펙 용량 : 75.6MB
- 업데이트 스펙 용량 : 140MB



9. ASTC 압축 8x8

- 용량 : 0.4MB
- 오픈 스펙 용량 : 42.8MB
- 업데이트 스펙 용량 : 79.3MB



10. 요약 및 작성자 의견

● 요약

- 카드 종류에 따라 사용되는 Large 카드에서, 다운로드 용량 및 대기시간, 메모리 점유율 등을 고려했을 때 압축 이미지 사용이 필요함
- AOS-ETC2 압축의 경우 가장 최고의 퍼포먼스라는 평이지만, ETC2를 지원하지 않는 기기 처리를 해야함.
- IOS-PVRTC 압축의 경우 색감 표현이 다른 압축에 비해 좋다는 평이지만, Alpha와의 경계선 처리에 문제가 있으며, 텍스처 관련 추가 작업이 필요.(POT).
- ASTC 압축의 경우 현재 나온 압축 방식 중 가장 좋지만, 기기 제약이 많이 따름.

● 작성자 의견

- ASTC 8x8이나 ASTC 6x6 사용을 하고, ASTC를 지원하지 않는 기기에 대해서는 텍스처 표현 방식(Unity 내부에 Quality Setting)으로 최적화