

Real-time skin rendering on graphics hardware

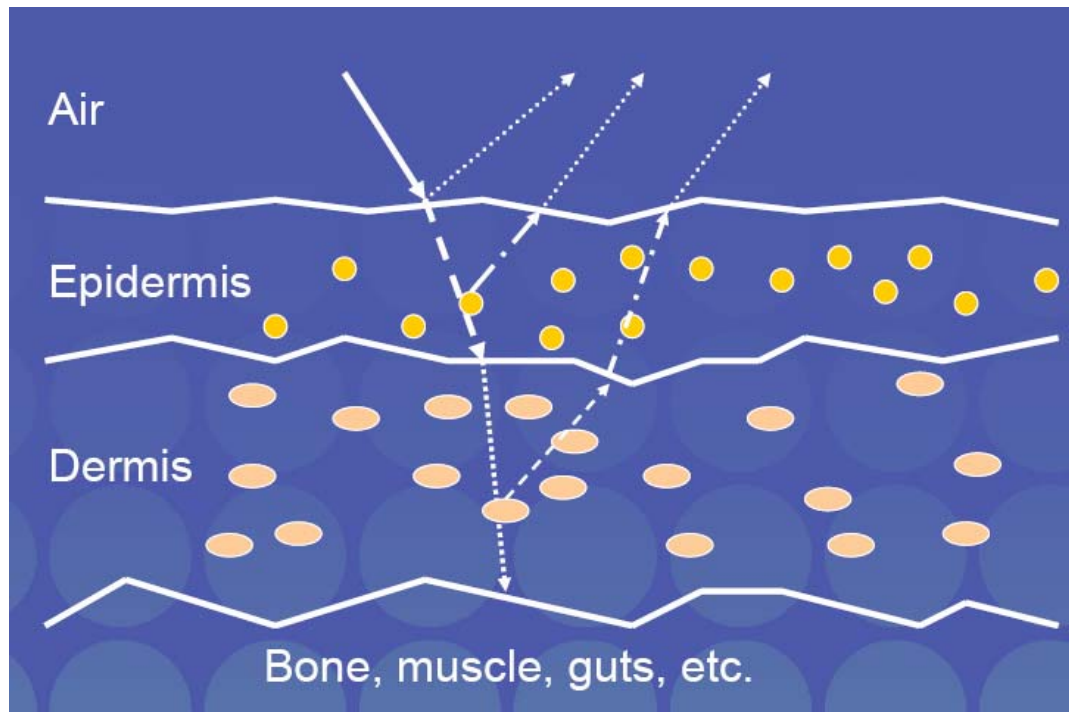
Pedro V. Sander
David Gosselin
Jason L. Mitchell

요약:김성익
2007.1.28

Spirit of Flame
3D RealTime Graphics Programming Study
Kasa

스킨 셰이딩

- 기존 라이트 모델은 딱딱한 표면에 적합하며 피부에는 맞지 않음



목표

- 간단하고 효과적인 스킨 렌더링
- 빠르다
- Subsurface Scattering 과 근접한 결과

텍스처 공간 라이팅

- 오브젝트의 텍스처 좌표로 offscreen 텍스처에 렌더링
- offscreen 텍스처 블러(blur)
- offscreen 텍스처를 읽어서 스펙큘러와 함께 렌더링
- 범프맵은 스펙큘러에만 적용

- 매트릭스 예



From *Matrix: Reloaded* sketch



- 결과



Current skin in Real Time



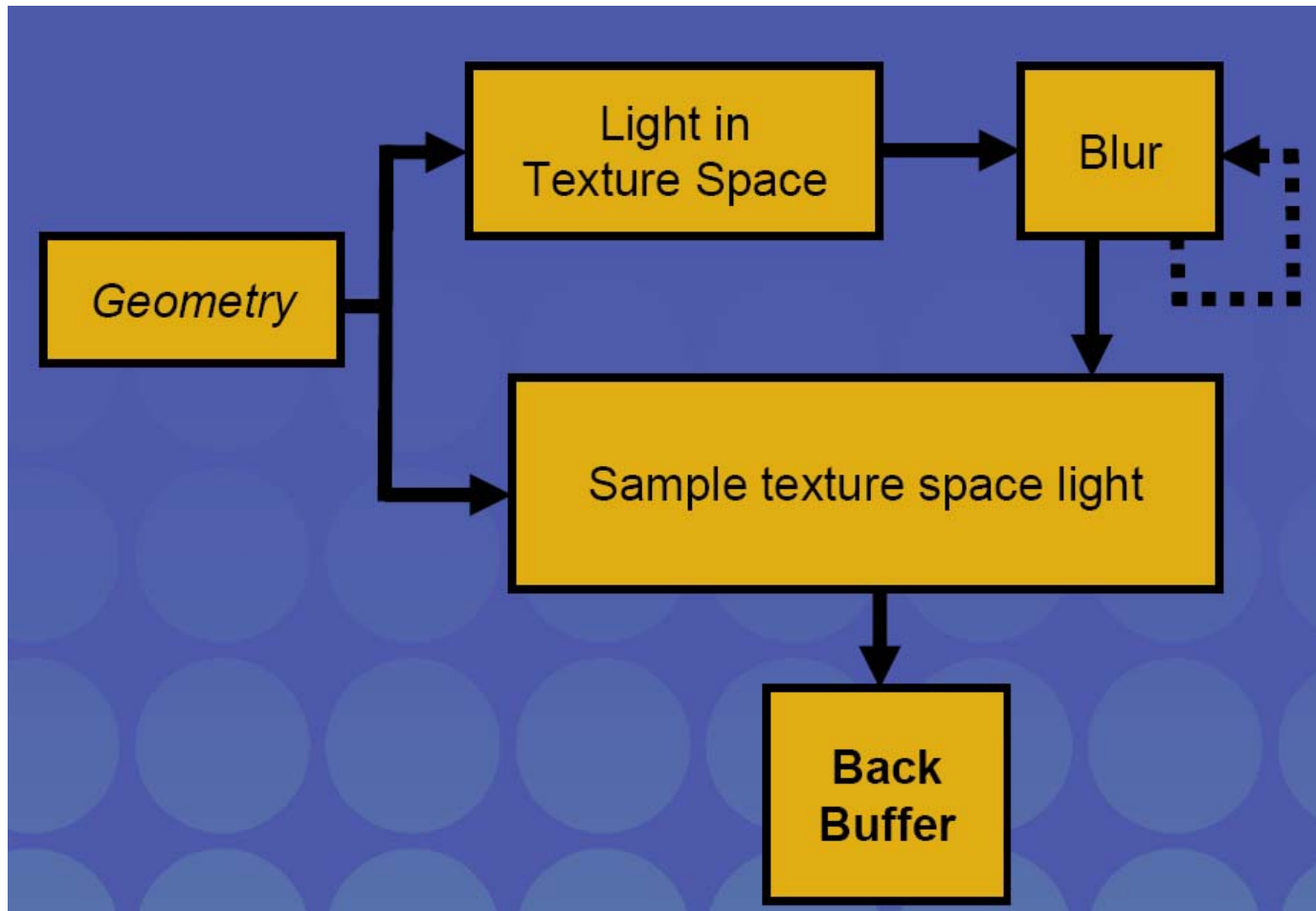
Spirit of Flame
3D RealTime Graphics Programming Study

Kasa

가이드

- 알고리즘
- 텍스처 공간 라이팅
블러(Blur) / 팽창(Dilation)
- 그림자
부드러운 그림자 / 투명 그림자
- 가속 효과
Early-Z Culling

기본 접근



일반적인 라이트 모델

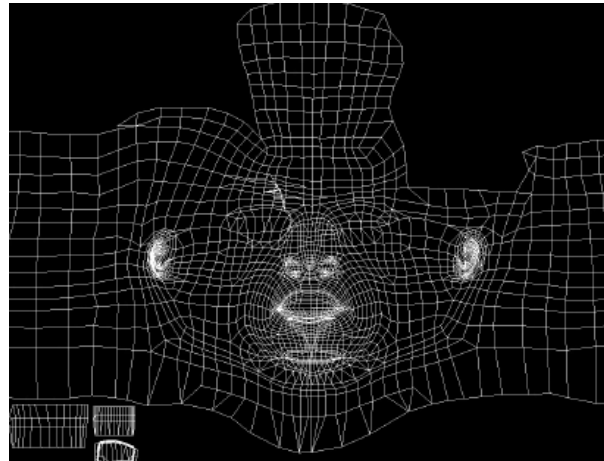
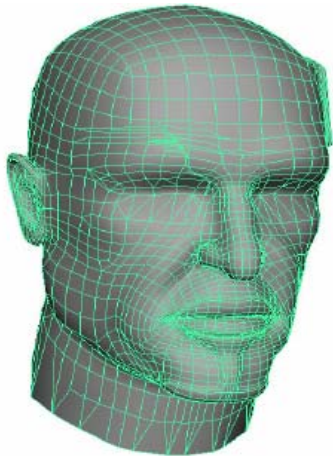


블러된 라이트 모델



텍스처에 렌더링

- 3d 모델처럼 라이팅한다.
- 텍스처에 그린다
 - 텍스처 좌표를 위치로
 - Unwap



spirit of Flame
3D RealTime Graphics Programming Study

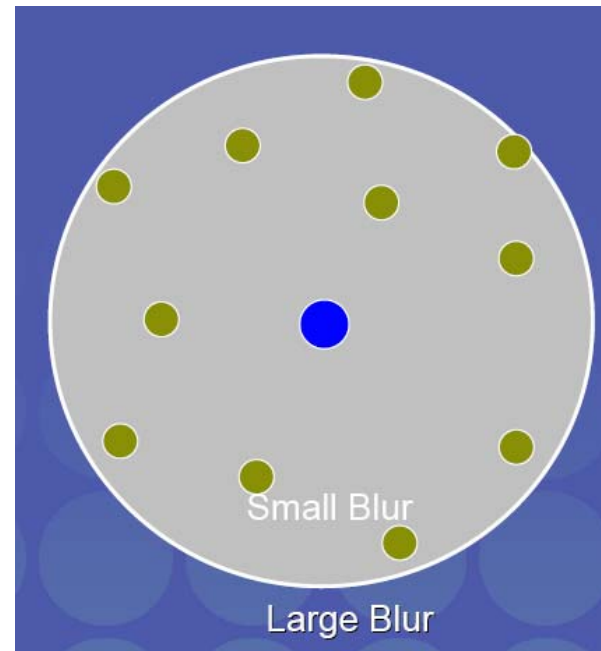
Kasa

블러

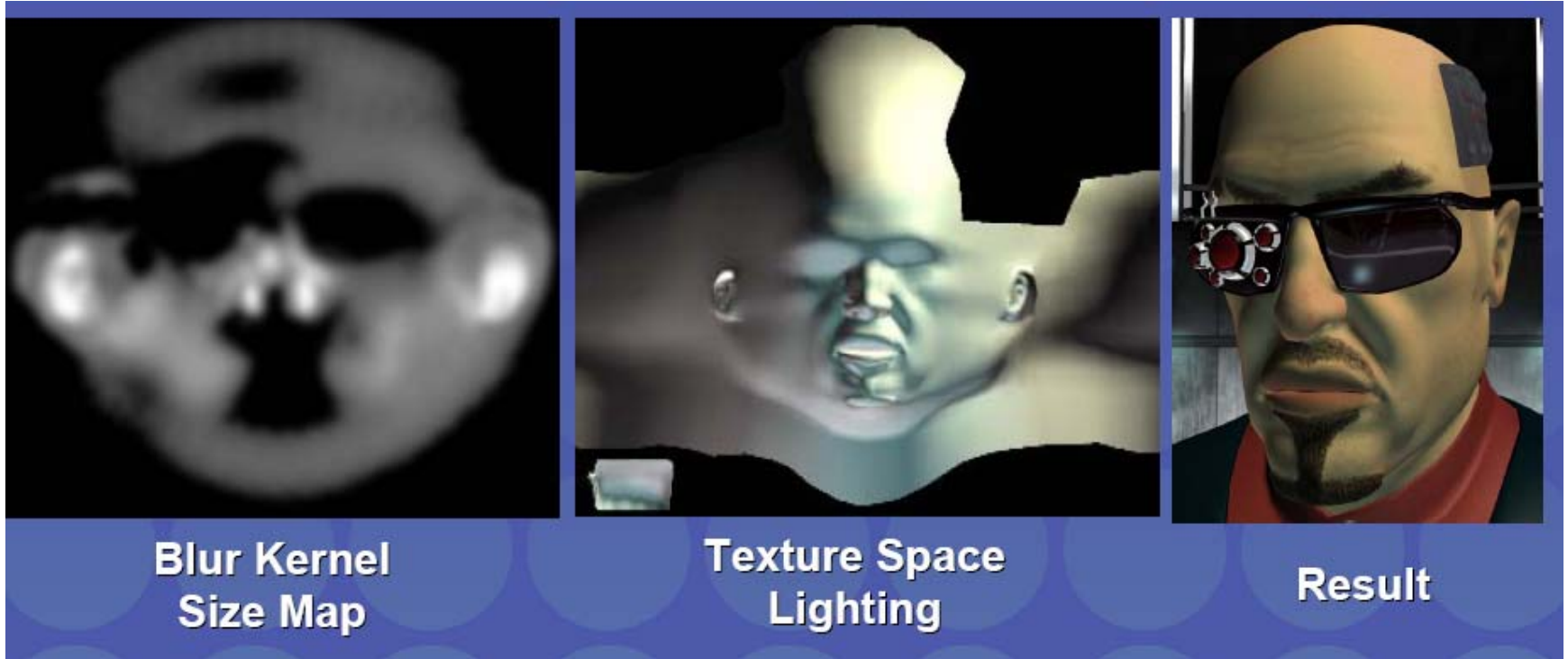
- 스킨 라이트의 표면 정보를 훔내
- 블러의 크기 조절(kernel) 가능한 필터
- 크기는 텍스처에서 읽는다
- 다양한 표면 효과
 - 귀, 눈에서 큰 값
 - 턱에서 작은 값

필터 Kernel

- 확률적인 샘플링
- 푸아송 분포 (독립성, 비집락성, 항시성)
- 중심에서의 2d 오프셋위치로 저장



Kernel 크기 맵과 텍스처

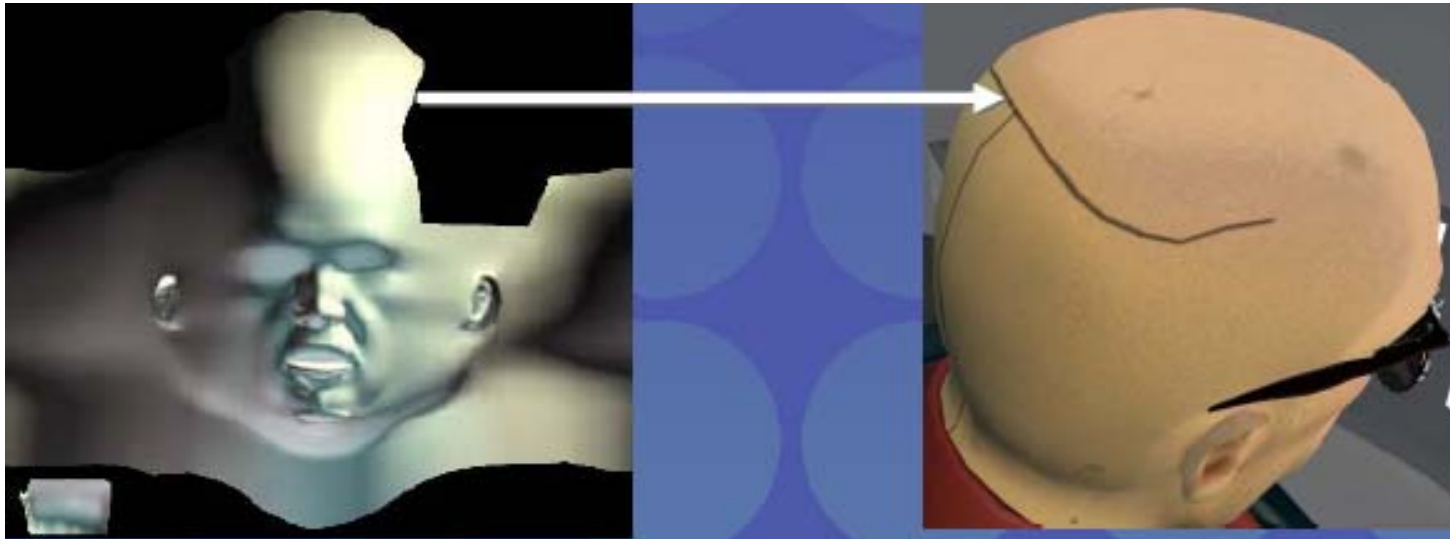


Spirit of Flame
3D RealTime Graphics Programming Study

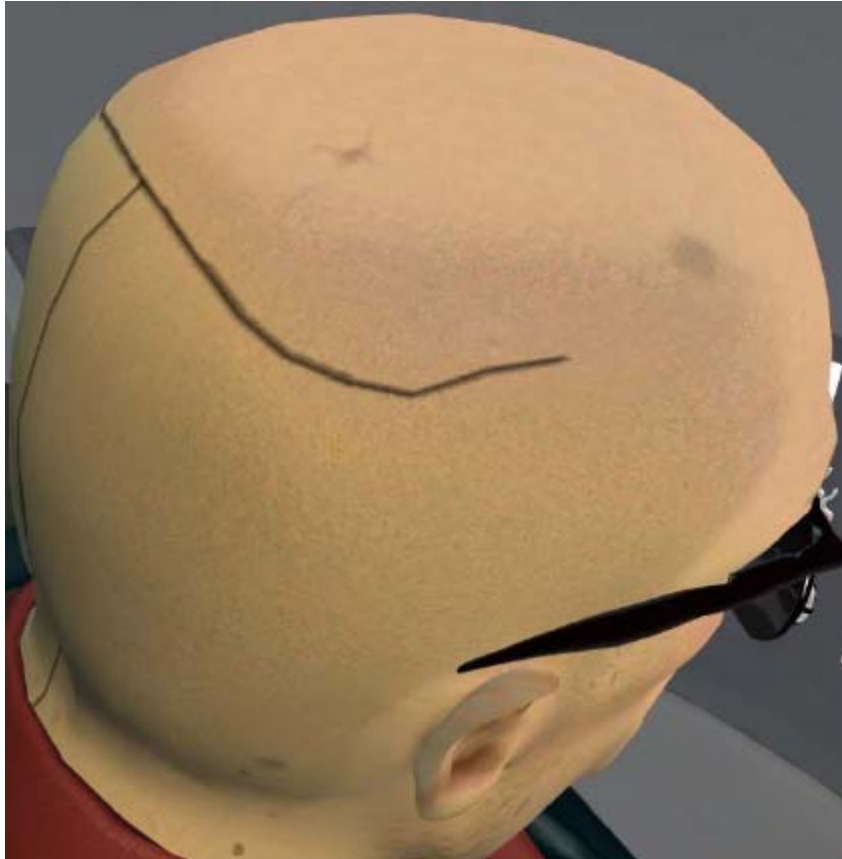
Kasa

팽창 (Dilation)

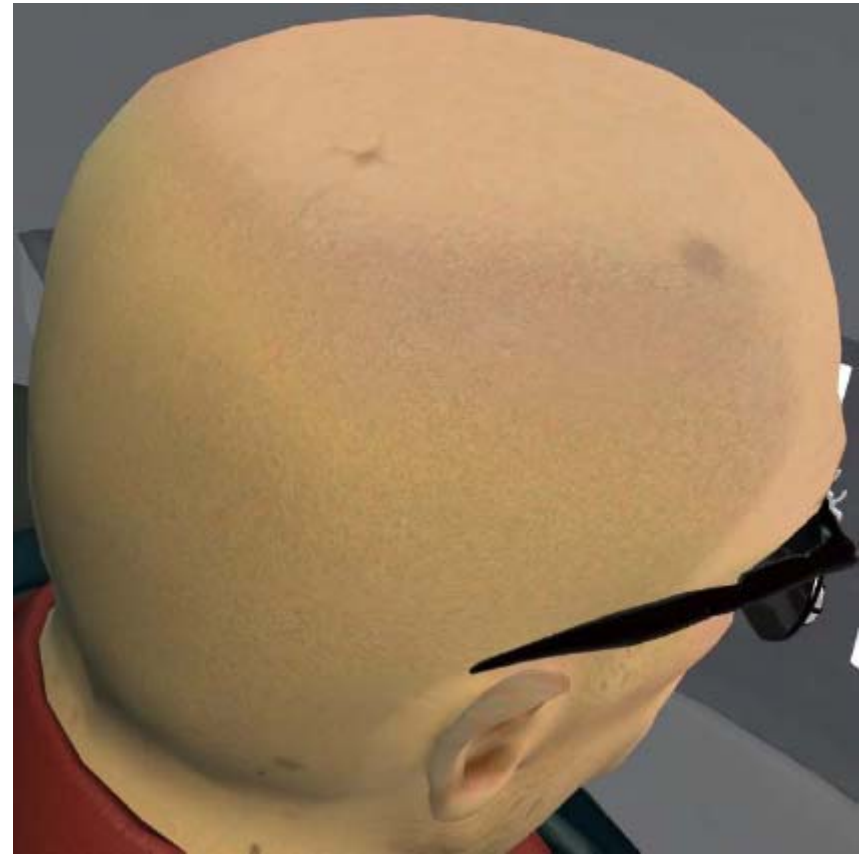
- 텍스처 틈이 생긴
(사용 안 하는 텍셀, 바이 리니어 필터링)
- 블러 pass 에서 Dilation 필요
- Offscreen 의 알파 채널 이동
- 값이 1.0이면 작은 값에 복사



Dilation 결과



Dilation 전



Dilation 후

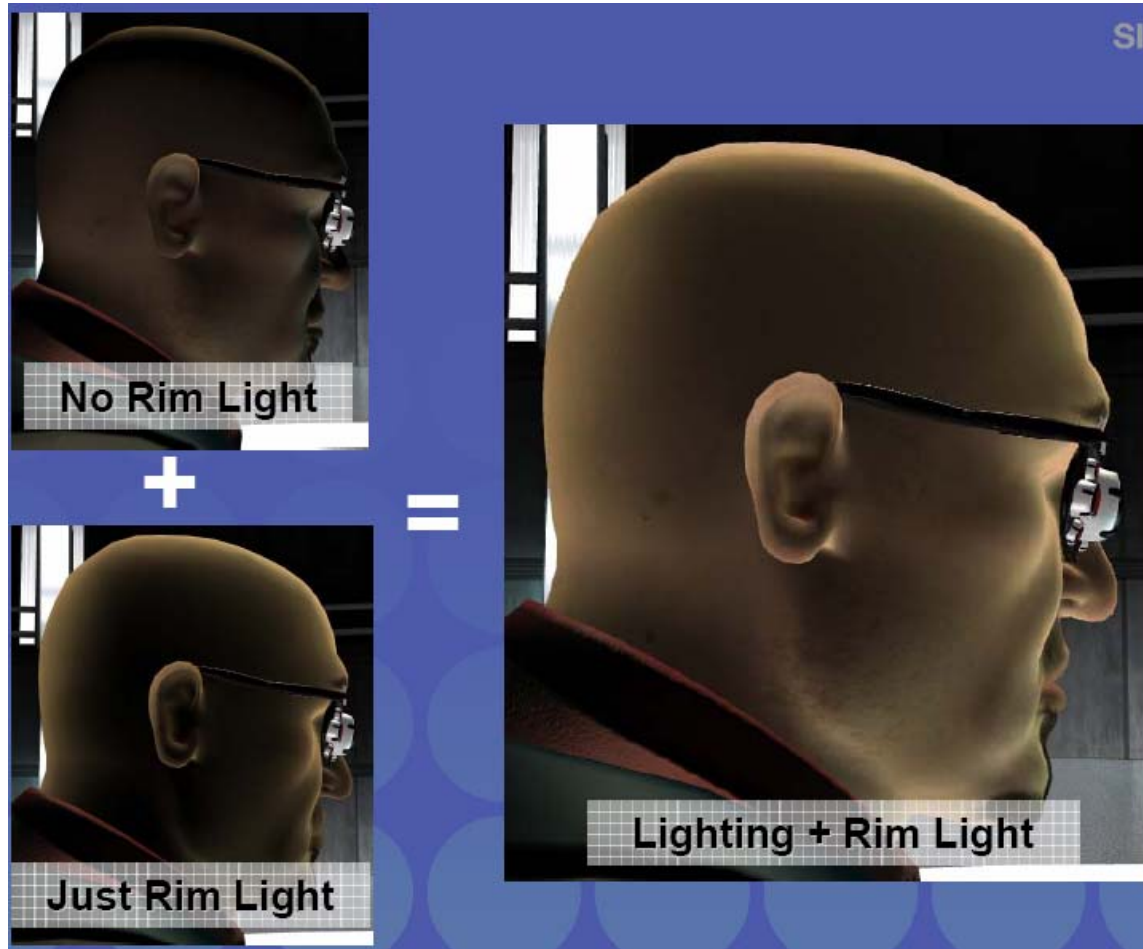
Spirit of Flame
3D RealTime Graphics Programming Study

Kasa

테두리(Rim) 라이트

- 백 라이트일때 피부에 라이트가 배어 나오길 원한다
- 뷰 벡터와 라이트 벡터를 반전한 값의 도트프로덕트 해서 프레넬 항에 곱한다
- 라이트가 오브젝트 뒤에 있을 때 보여준다

Rim 라이트 적용 결과



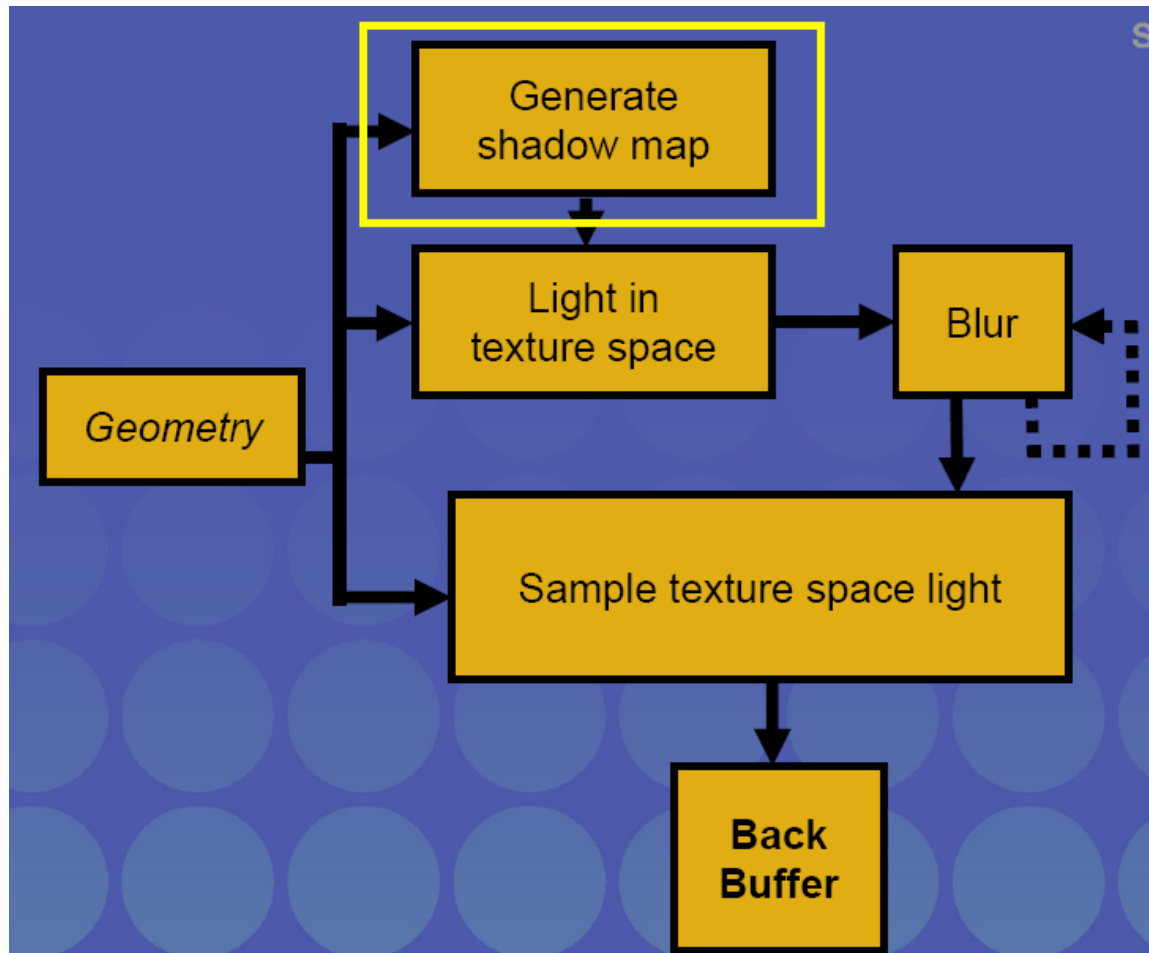
그림자

- 그림자 맵 사용
 - 텍스처 라이팅시 그림자 적용
 - 블러(blur)가 자유로움
 - 자연스럽고 부드러운 그림자
- 키 라이트 하나에 대해서 가능함

그림자 맵

- 라이트 뷰에서 그림자 맵에 렌더링
 - 바운드를 이용하면 효과적으로 공간 할당 가능
- 그림자 맵 렌더링 pass: 깊이 값을 렌더링
- 텍스처 라이팅 패스 : 픽셀 세이더에서 깊이 값과 비교

라이팅이 추가된 다이얼그램

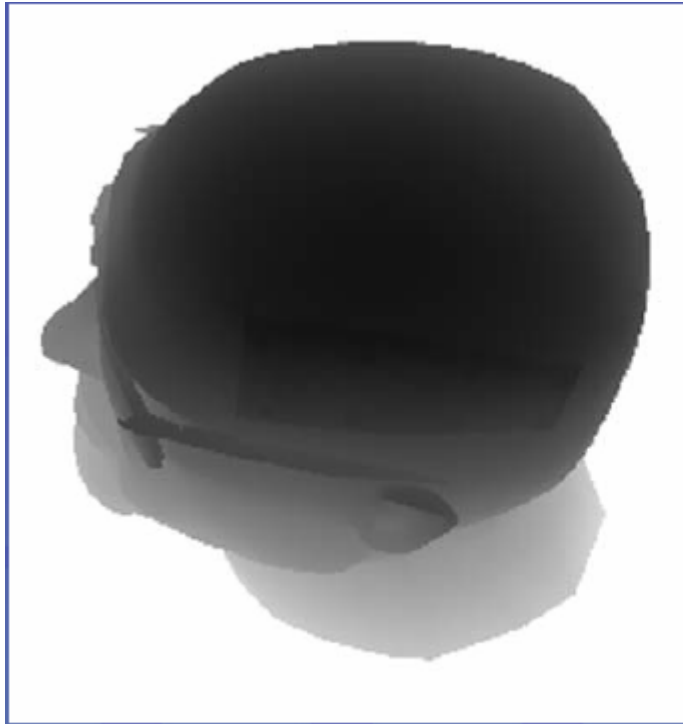


Flame

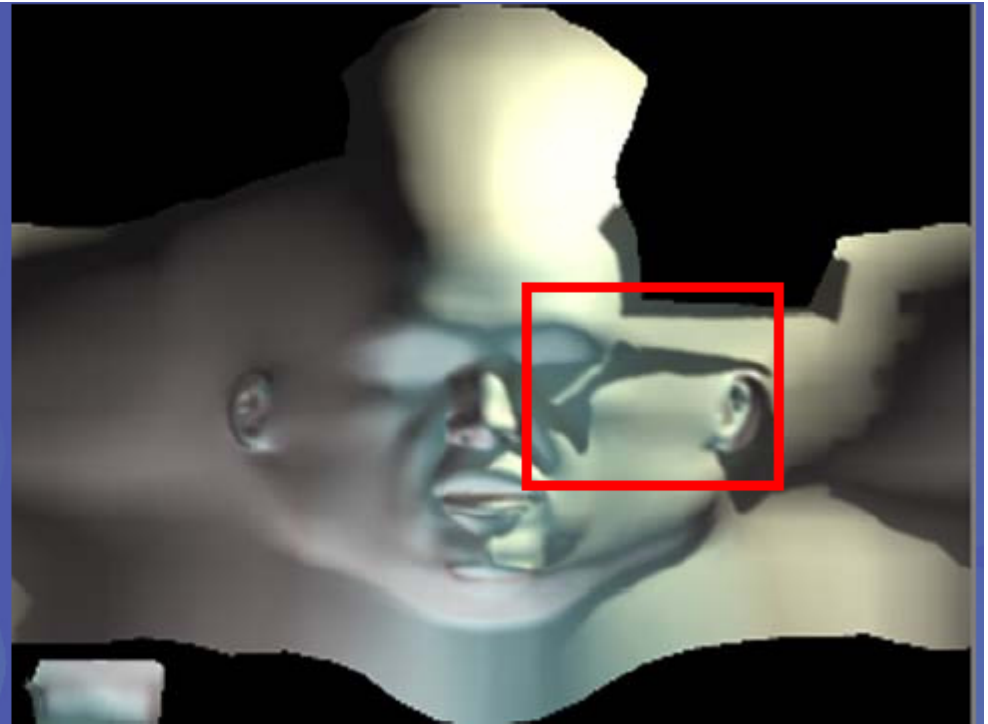
3D Realtime Graphics Programming Study

Kasa

그림자 맵과 그림자가 적용된 라이 트 텍스처



Shadow Map (depth)



Shadows in Texture Space

Spirit of Flame
3D RealTime Graphics Programming Study

Kasa

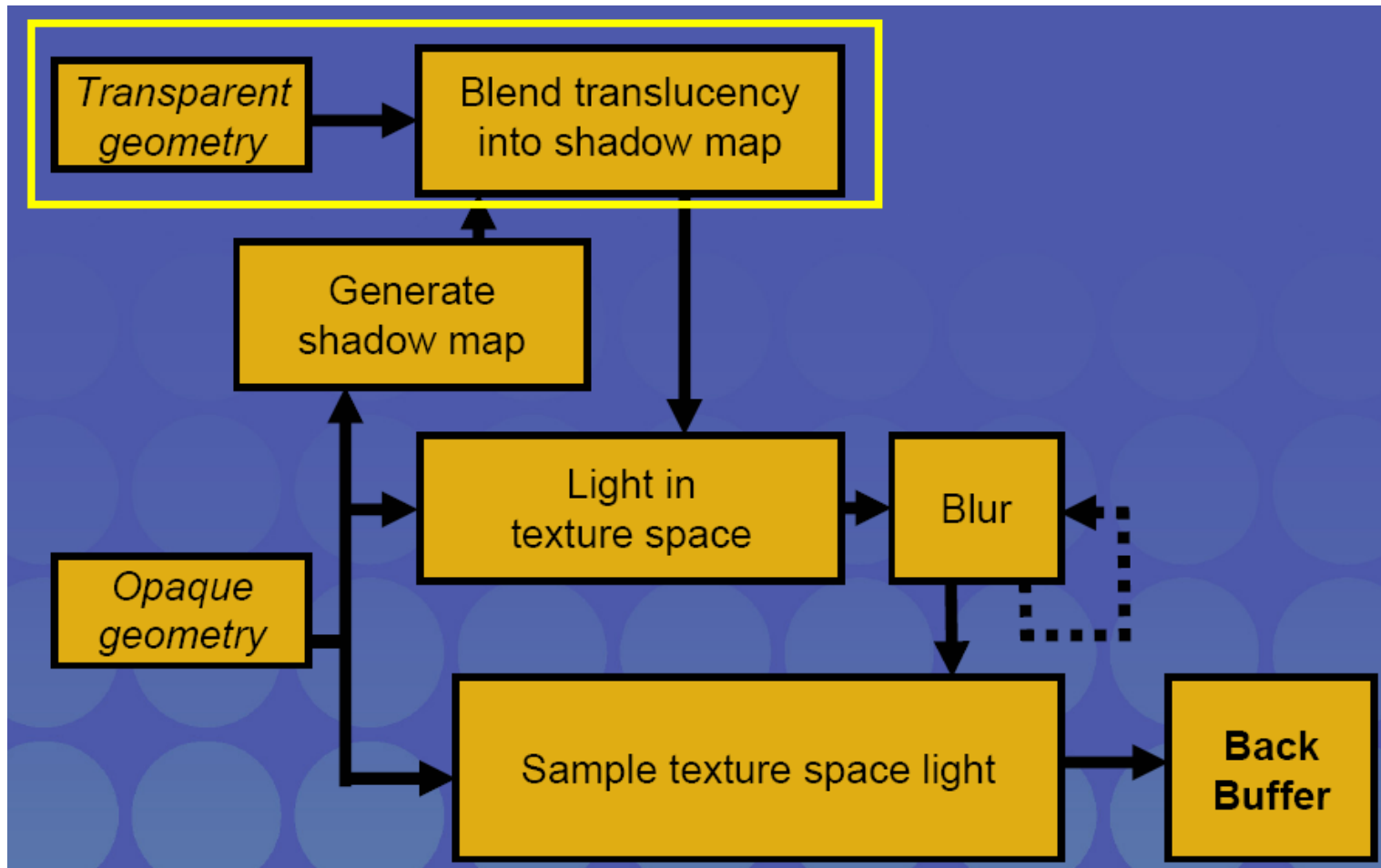
그림자 적용 결과



반투명한 오브젝트의 그림자

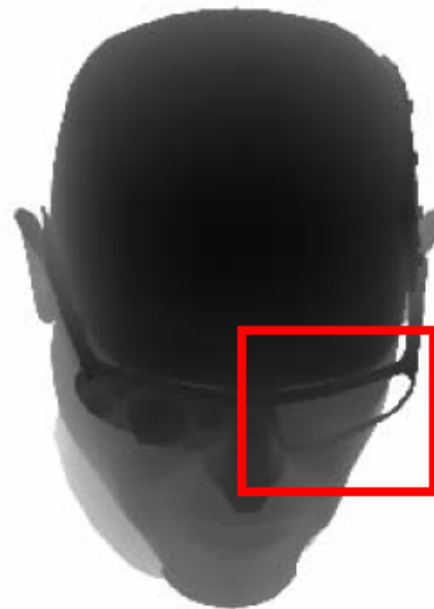
- 알고리즘
 - 그림자 버퍼의 알파값 사용
 - 그림자 버퍼 RGB채널에 투명한 그림자 렌더링
 - 깊이 테스트 on, 깊이값 쓰기 off
- - 텍스처 라이팅 픽셀 셰이더
 - 그림자가 아닌 픽셀은 그림자 버퍼의 RGB값을 곱해줌

다이어그램





Opaque shadow map



Translucent shadow map



RGB



Opaque Shadows



Translucent Shadows

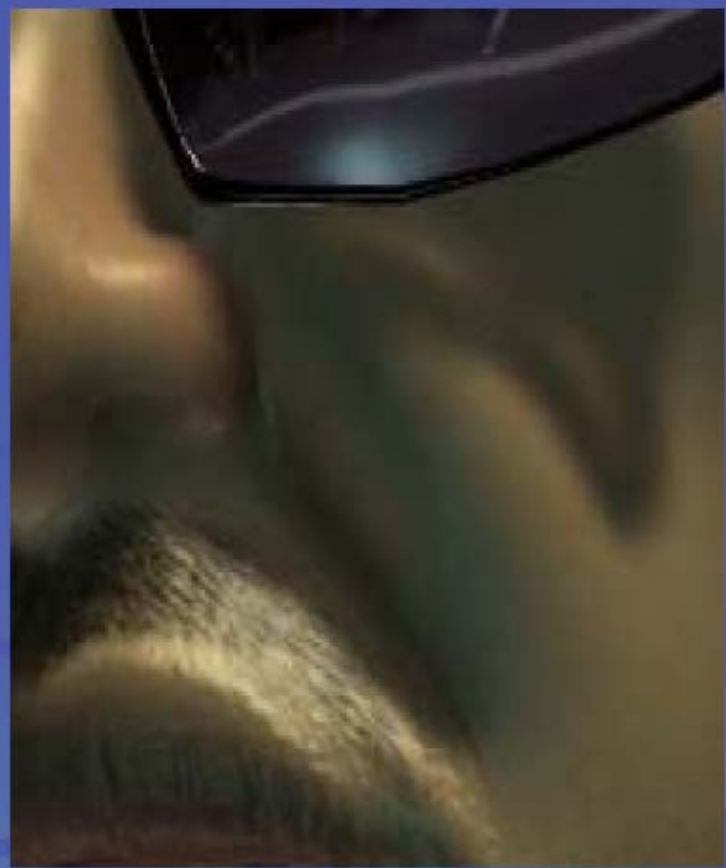
스펙큘러

- 범프맵 사용
- 퍼-픽셀 계산
- 그림자에는 스펙큘러가 안생김
라이트 텍스처 사용
 - 스펙큘러에 텍스처 공간의 라이트값사용

그림자가 흐린 결과물



Specular without shadows

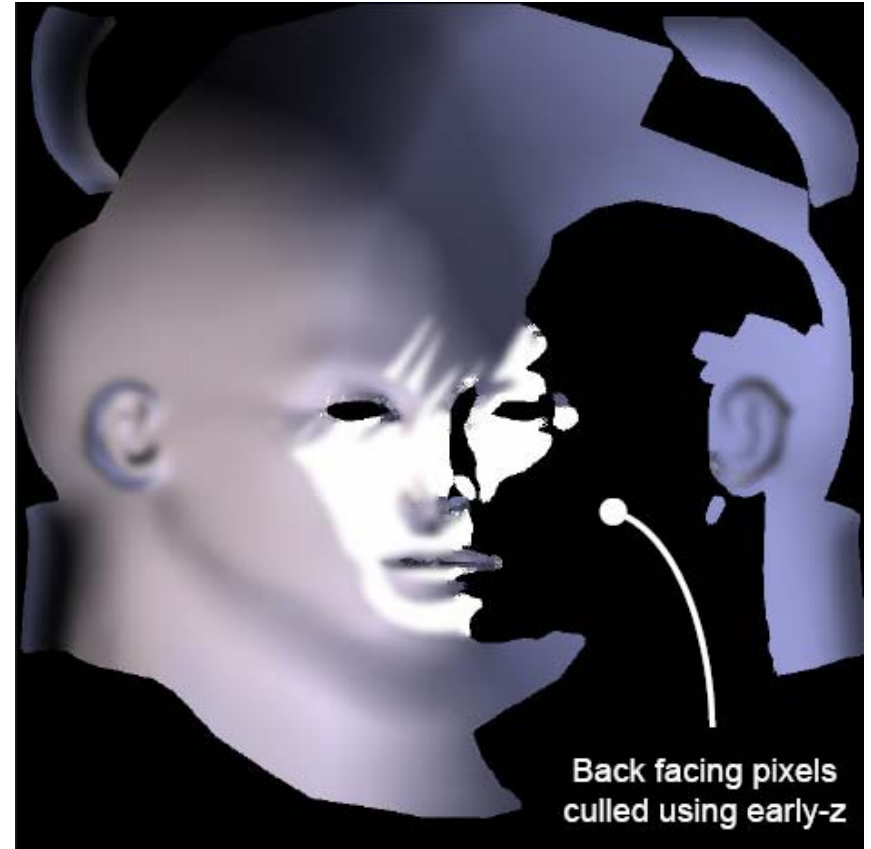


Specular with shadows

Early-Z 컬링

- 깊이 검사가 픽셀세이더 전 과정
 - 픽셀세이더의 비싼 연산을 줄여줌
 - 픽셀세이더가 z값을 출력하지 않을 때만 적용 가능
- 텍스처 공간에선 z버퍼 필요 없음
- Z버퍼엔 아무 값이나 저장 가능
- Early-Z 를 사용
 - 뒷면 제거, 거리에 따라 제거
- Z 테스트는 화면상 안 보이는 폴리곤 처리를 생략 가능하게 해줌

Backface culling



Back facing pixels
culled using early-z

Spirit of Flame

3D RealTime Graphics Programming Study

Kasa

요약

- 간단하고 빠른 스킨 렌더링
- 텍스처 공간 / 블러
- 부드러운 그림자
- 반투명한 그림자
- Early-Z컬링을 이용한 최적화

끝

- 원문 :
- http://ati.de/developer/siggraph04/Sander_SkinSketch.pdf
- 참고영상 :
- http://ati.de/developer/siggraph04/Sander_SkinSketchVideo.mov