서 카메라는 그림 1.21-(a)와 동일한 영상을 생성하게 된다.

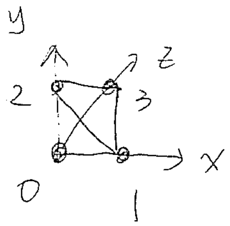
요약하면, 오른손 좌표계와 왼손 좌표계 간 포팅을 위해서는 두 가지 리모델링 작업, 즉 정점 재정렬 및 z 좌표 부호 변경 작업을 수행해야 한다. 이 중 z 좌표 부호 변경은 반드시 수행해야 하지만, 정점 재정렬은 피해 갈 수 있는 방법이 있다. 이에 대해서는 3.4절에서 자세히 다루게 될 것이다.

실시간 그래픽스의 양대 API인 OpenGL과 Direct3D는 서로 다른 좌표계를 사용한다. OpenGL은 오른손 좌표계를, Direct3D는 왼손 좌표계를 사용한다. 얼핏 보면 단순해 보이는 이 차이는 초보 프로그래머에게 상당히 커다란 혼란을 일으키곤 했다. 하지만 'GPU 프로그래밍'이 가능해지면서 이 혼란의 정도는 상당히 누그러졌다. 예를 들어, 이제는 Direct3D 상에서 오른손 좌표계를 사용해도 된다. 이에 대해서는 2장 이후 자세히 다루게 될 것이다.

이 책은 기본적으로 오른손 좌표계를 사용한다. 보통의 독자들이 오른손 좌표계에 더 익숙하기 때문이다. 따라서, 삼각형 정점은 특별한 언급이 없는 한 항상 반시계 방향으로 정렬될 것이다.

연습문제

1. 닫힌 3차원 폴리곤 메쉬 중 가장 간단한 것은 사면체(tetrahedron)이다. 이를 구성하는 네 개의 정점이 (0,0,0), (1,0,0), (0,1,0), (0,0,1)이라고 가정하자. 이 사면체의 삼각형 노멀은 사면체 외부를 향해야 한다.
   (a) 이 사면체가 오른손 좌표계에서 사용된다고 가정하고, 인덱스 삼각형 리스트의 정점 버퍼 및 인덱스 버퍼를 도시하라.
   (b) 이 사면체가 왼손 좌표계에서 사용된다고 가정하고, 인덱스 삼각형 리스트의 정점 버퍼 및 인덱스 버퍼를 도시하라.



2. 닫힌 삼각형 메쉬에서 v, e, f가 각각 메쉬의 정점, 변, 면의 개수를 나타낸다고 하자. [노트: 삼각형 메쉬에서의 정점-삼각형 비율]에서 우리는 $f = 2v - 4$임을 밝혔다. 이와 마찬가지로 v와 e 간 관계를 정의하라.

3. ATVR은 average transform to vertex ratio의 약자로, 정점당 평균 몇 번의 처리 단계를 거치는지 측정한다(이는 ACMR과 다르다).
   (a) 메쉬의 삼각형이 최적으로 정렬되었을 때 ATVR 값은 얼마인가?

$\sqrt{(1-\sqrt{3})^2} = \cancel{1-2\sqrt{3}+3-4}\ 2\sqrt{3}$

$= \sqrt{3}-1$

연습문제

1. 2차원에서 (원점 대신) 특정 점 (a,b)를 중심으로 θ만큼 회전하는 아핀 변환 행렬을 계산하라.

2. 선형대수에서 회전 행렬의 역행렬은 그 전치 행렬과 같다(이는 증명 가능하다). 두 개의 비표준 직교 정규 기저 $\{a,b,c\}$와 $\{d,e,f\}$가 주어졌을 때, $\{a,b,c\}$에서 정의된 벡터를 $\{d,e,f\}$에서 정의된 벡터로 변환하는 3×3 행렬을 계산하라.

3. 세 개의 직교 정규 벡터 a, b, c가 있고, a, b, c 간에는 $a \times b = c$ 관계가 성립한다. 또한, a, b, c 중 어느 것도 표준 기저의 벡터와 일치하는 것은 없다. 벡터 a, b, c를 기준으로 하는 축소확대 행렬을 계산하고자 한다. 각 벡터에 대한 축소확대 인자는 각각 s_a, s_b, s_c로 표기한다. 이 축소확대 행렬은 통상 세 개의 3×3 행렬 곱으로 표현된다. 각각의 3×3 행렬을 계산하라.

4. 뷰 변환은 이동과 회전으로 구성된다. 카메라 파라미터는 다음과 같다.
   EYE$=(0,0,-\sqrt{3})$, **AT**$=(0,0,\cancel{1})$ -1 **UP**$=(-10,0,1)$.
   (a) 이동 행렬을 계산하라.
   (b) 회전 행렬을 계산하라.

5. 우리는 2.4.3절에서 투영 변환 이후의 z 범위가 $[-1,0]$임을 가정하고 투영 행렬을 유도하였다. 하지만, OpenGL에서 z 범위는 $[-1,1]$이다. 식 (2.27)의 OpenGL 투영 행렬을 유도하라.

$$T = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \sqrt{3} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$n = (0, 0, -\sqrt{3}+1)$

$N_s \Rightarrow (0, 0, \cancel{\tfrac{\sqrt{3}}{\sqrt{4-2\sqrt{3}}}})$

$$UP \times n = \begin{vmatrix} i & j & k \\ -10 & 0 & 1 \\ 0 & 0 & -1 \end{vmatrix}$$

$= \cancel{(0, -10, 0)}$

① $N_u \Rightarrow (0, -1, 0)$

⇒ z

$= v'$

(2.35)

(2.36)

변환한다. 즉, 식 (2.36)에 넣

(2.37)

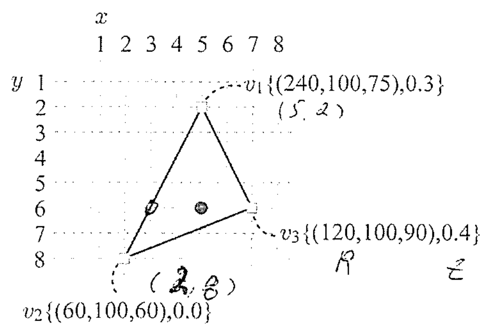
(2.38)

$cot\frac{fovy}{2}$와

연습문제

1. 뷰포트의 모서리 점이 (10,20,1)과 (100,200,2)로 주어졌다. 뷰포트 변환은 반사, 축소확대, 이동 변환으로 구성된다.
   (a) 반사 행렬을 계산하라.
   (b) 축소확대 행렬을 계산하라.
   (c) 이동 행렬을 계산하라.

2. 스크린 공간에 다음과 같은 삼각형이 있다. 정점별 속성은 $\{(R, G, B), z\}$ 이다. 픽셀 좌표 (5,6)에서의 R과 z를 계산하라.



$\Delta y = 6$

$\Delta R = 180$

$\Delta z = -0.3$

$\frac{\Delta R}{\Delta y} = -30$

$\frac{\Delta z}{\Delta y} = -\frac{0.3}{6}$ $R_l = R_1 + 4 \times -30 = 240 - 120$ (120) ($z = 0.1$)

$= -\frac{0.1}{2}$

$= -0.05$

$R = 120$
$z = 0.25$

3. 그림 3.18에 도시된 카메라 공간 광선의 시작점 $(x_c, y_c, -n)$을 다른 방법을 사용해 계산해 보자.
   (a) 뷰포트 변환의 역변환을 (x_s, y_s)에 적용하여 NDC로 표현된 클립 공간에서의 광선 시작점을 계산하라.
   (b) (a)의 결과는 식 (3.4)의 점과 같아야 한다는 사실을 이용하여 x_c와 y_c를 계산하라.

z-컬링은 그 의미가 없어진다. 이런 경우, z-컬링은 자동으로 생략된다. 참고문헌 [12, 13]에서 z-컬링이 생략되는 여러 가지 경우를 확인할 수 있다. Z-컬링을 이용해 렌더링 성능을 향상하기 위해서는 어떤 경우에 z-컬링이 생략되는지, 그리고 생략되지 않더라도 어떤 경우에 z-컬링이 제 성능을 발휘하지 못하는지 이해할 필요가 있다.

4.3.2 응용 사례: 프리-z 패스

Z-컬링 기능을 최대한 활용하고자 하는 목적으로 이른바 **프리-z 패스**(pre-z pass)라 불리는 알고리즘이 제안되었다. 이는 두 번의 렌더링 과정을 거치는 **투패스 알고리즘**(two-pass algorithm)이다. 첫 번째 패스에서는 조명이나 텍스처링을 수행하지 않고 오로지 z-버퍼만 채운다. 컬러 버퍼는 채워지지 않는다. 즉, 모든 픽셀에 대해 카메라로부터 가장 가까운 물체 표면의 깊이 값만을 저장하고 첫 패스는 종료된다.

이렇게 만들어진 z-버퍼를 사용하여 두 번째 패스에서는 통상적인 렌더링을 수행한다. 그러면 z-컬링은 현재 z-버퍼에 의해 가려지는 프래그먼트를 걸러낸다. 따라서 두 번째 패스에서 오버드로우는 1에 접근하게 된다. 만약 프래그먼트 프로그램이 복잡한 연산을 수행한다면, 이같은 프리-z 패스 알고리즘은 전체적인 성능 향상에 큰 도움을 줄 것이다.

그림 4.12의 장면에 대해 프리-z 패스 알고리즘을 수행하는 실험을 하면서, 물체들을 일부러 뒤에서부터 앞으로 정렬했는데도 불구하고, 물체를 앞에서부터 뒤로 정렬한 상태에서 수행한 단일 패스 알고리즘보다 빠른 렌더링 속도를 보였다(물론, 물체를 앞에서부터 뒤로 정렬한 상태에서 프리-z 패스 알고리즘을 수행하면, z-버퍼 쓰기 연산이 줄어들게 되므로 속도는 더 빨라진다). 10.5절은 프리-z 패스 알고리즘과 관련 있는 다른 기법을 소개한다.

연습문제

1. 하나의 픽셀을 놓고 경쟁하는 다섯 개의 삼각형이 이 픽셀 위치에서 서로 다른 깊이 값을 가지고 있다고 하자. 만약 삼각형들이 임의의 순서로 처리된다면, 이 픽셀에 대해 평균 몇 번의 z-버퍼 쓰기 연산이 수행되는가?

2. 하나의 픽셀을 놓고 경쟁하는 세 개의 삼각형이 이 픽셀 위치에서 다음과 같은 RGBA 색상과 z 값을 가지고 있다: {(1,0,0,0.5),0.25}, {(0,1,0,0.5),0.5}, {(0,0,1,1),0.75}. 삼각형은 뒤에서부터 앞으로 차례차례 처리된다. 픽셀의 최종 색상을 계산하라.

A 0 0 1 1

참고문헌 [12,
이용해 렌더링
되지 않더라도

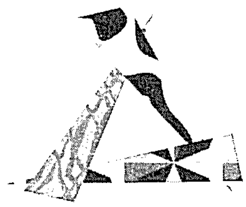
re-z pass)라
패스 알고리즘
하지 않고 오
해 카메라로부

을 수행한다.
라서 두 번째
복잡한 연산을
을 줄 것이다.
서, 물체들을
정렬한 상태
를 앞에서부
산이 줄어들
는 다른 기법

는 다른 깊이
이 픽셀에

같은 RGBA
,1),0.75).
계산하라.

3. 아래 그림과 같이 삼각형 세 개가 겹쳐져 있다.



(a) 삼각형 세 개가 모두 반투명하다고 가정하자. 임의의 삼각형부터 시작해 시계 방향으로 이들을 처리할 경우, 렌더링 결과를 도시하라.

(b) 렌더링 결과에 어떤 문제가 있는가? 그리고 이를 해결하기 위한 방안은 무엇인가?

4. 야외 장면의 현실감을 높이기 위해 안개(fog)를 사용할 때, 종종 다음과 같은 블렌딩 기법을 사용한다.

$$c = fc_f + (1-f)c_o$$

여기에서 c는 최종 색상이고, f는 카메라로부터 거리가 멀어질수록 안개가 짙어지는 정도를 나타내며, c_f는 안개의 색상이고, c_o는 물체의 색상이다. 가장 간단한 선형 안개(linear fog)를 구현하는 방법은, 뷰 프러스텀 내부에 안개가 고루 분포되게 하는 것인데, 전방 평면에 위치한 물체는 선명하게 보이고 후방 평면에 위치한 물체는 완전히 안개에 가려 보이지 않게 할 수 있다. 이 같은 선형 안개 구현을 위한 f값을 원점에서 전방 평면까지의 수직 거리 N과 후방 평면까지의 수직 거리 F의 함수로 정의하라.

$f(N) = 0$ $a\cdot N + b = 0$
$f(F) = 1$ $a\cdot F + b = 1$

$f(z) = \frac{z-N}{F-N}$ $a(F-N) = 1$

$a = \frac{1}{F-N}$

$b = \frac{-N}{F-N}$

$\frac{1}{N-F}z - \frac{F}{N-F}$ $F-z$

$(10-t)^2 + 1 = 4$

$t^2 - 20t + 97 = 0$

표면으로 렌더링하는 작업은 큐브맵 제작 과정과 유사성을 가진다).

래디오시티 알고리즘은 실시간에 구현하기에 연산량이 너무 많다. 따라서, 실시간 그래픽스의 경우, 보통 전처리 단계에서 이를 실행하여 그 결과를 텍스처에 저장한 후, 런타임에 이 텍스처를 사용한다. 이러한 텍스처를 **라이트맵**(light map)이라고 부르는데, 이를 사용한 알고리즘은 10.2.2절에 소개된다.

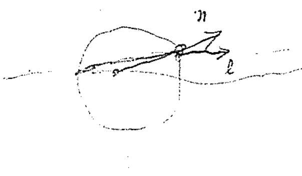
연습문제

1. 식 (5.11)은 하나의 평행 광원을 가정하여 정의되었다.
   (a) 여러 개의 평행 광원을 다룰 수 있도록 식 (5.11)을 수정하라.
   (b) 평행 광원을 점 광원으로 대체할 경우, 식 (5.11)을 어떻게 수정할 것인가?

2. 식 (5.7)과 (5.12)는 작은 차이를 가진다. 무엇 때문에 이 차이가 발생했는가?

   수직 l, I_i 반대방향

3. 반지름이 [illegible]이고 중심이 원점인 구에 광선 추적법을 적용해 보자.
   (a) 광선이 (10,1,0)에서 (−1,0,0) 방향으로 발사되었다. 이 광선을 t에 대한 매개변수 방정식으로 표현하라.
   (b) 구를 음함수로 표현한 후, 광선의 매개변수 방정식을 이용하여 구와 광선의 교차점을 계산하라.
   (c) 반사 벡터를 계산하기 위해서는 교차점에서의 노멀이 필요하다. 본 문제에서는 노멀을 쉽게 계산할 수 있다. 어떻게 계산할 것인가?
   (d) 반사 벡터를 계산하라.



$x^2 + y^2 + z^2 = 4$

$d = (-11, -1, 0)$

$x(t) = S_x + td$

$= 10 - 11t$

$y(t) = 1 - t$

$z(t) = 0$

$(10-11t)^2 + (1-t)^2 = 4$

$121t^2 - 220t + 100$

$+ t^2 - 2t + 1 =$

$122t^2 - 222t + 100$

$122 \quad -100$

$r = 2n(n \cdot l) - l$

교차점 $n = \left(\frac{1220-1100}{122} = \frac{120}{122}, \frac{22}{122}, 0\right)$

가 되는데, v를 $(1-w)$로 대체하면 이는 $(1-w)^2d+2w(1-w)e+w^2f$로 바꿔 쓸 수 있다. 이것은 바로 2차 베지어 곡선의 식으로 그림 6.18-(f)의 아래쪽 모서리를 표현한다. 이 모서리는 오로지 d, e, f에 의해 정의되는 곡선임을 이해하자. 즉 베지어 패치처럼 베지어 삼각형에서도 곡면의 경계는 컨트롤 포인트 망의 경계에 놓인 컨트롤 포인트에 의해서만 정의된다.

3차 베지어 삼각형은 그림 6.19-(a)에 보인 바와 같이, 2차 베지어 삼각형을 위한 컨트롤 포인트 망의 모서리에 하나씩 컨트롤 포인트를 추가하여 정의된다. 한편, 컨트롤 포인트 망 구조를 일관되게 만들기 위해서는 내부에 새로운 컨트롤 포인트 e를 추가해야 한다. 각각의 컨트롤 포인트는 가중치를 할당받는데 이는 u, v, w의 함수로 표현된다. 이 가중치는 그림 6.19-(b)에 도시된 것처럼 반복적 무게중심 보간 기법을 통해 계산된다.

$B_i^n = {}_nC_i\, t^i (1-t)^{n-i}$

$((1-v)^3, 3v(1-v)^2, 3v^2(1-v), v^3)$

연습문제

1. 한 베지어 패치가 u 기준으로는 2차, v 기준으로는 3차로 정의되었다. 컨트롤 포인트 행렬은 다음과 같다.

$$\begin{pmatrix} p_{00} & p_{01} & p_{02} \\ p_{10} & p_{11} & p_{12} \\ p_{20} & p_{21} & p_{22} \\ p_{30} & p_{31} & p_{32} \end{pmatrix} = \begin{pmatrix} (0,0,4) & (0,3,4) & (0,6,4) \\ (3,0,0) & (3,3,0) & (3,6,0) \\ (6,0,0) & (6,3,0) & (6,6,0) \\ (5,0,4) & (5,3,4) & (5,6,4) \end{pmatrix}$$

(a) (u,v)=(0,1)일 때 곡면의 점을 계산하라. $(0\ 0\ 0\ 1)$

(b) (u,v)=(0.5,0.5)일 때 곡면의 점을 계산하라.

$= (0,0,0,1)\begin{pmatrix} p_{00} \\ p_{10} \\ p_{20} \\ p_{30} \end{pmatrix} = p_{30}$

2. 세 개의 점 {(1,0),(0,1),(−1,0)}이 주어졌다.

(a) 이 점 모두를 지나는 2차 베지어 곡선이 있는데, (0,1)에서의 매개변수 t는 0.5이다. 이 베지어 곡선을 계산하라. $(1-t)^2p_0 + 2t(1-t)p_1 + t^2p_2$

(b) 매개변수 t가 0.75가 되는 점의 (x,y) 좌표를 계산하라.

$p_0=(1,0)$ $t=0$

$p_2=(-1,0)$ $t=1$

$\frac{1}{4}(1,0)+\frac{1}{2}p_1+\frac{1}{4}(-1,0)=(0,1)$ $t=\frac{1}{2}$

3. 2차 베지어 곡선 $p(t)$를 따라 이동하는 카메라가 있다. 이 베지어 곡선의 컨트롤 포인트는 p_1, p_2, p_3이다. **EYE**는 곡선 $p(t)$에 놓이고, **AT**은 원점과 p_4를 연결하는 선분을 따라 움직인다. **UP**은 월드 공간의 y축으로 고정되어 있다.

$p_1=(0,2)$

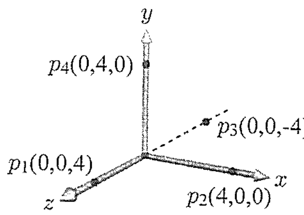
$$u = up \times n = \frac{\begin{vmatrix} i & j & k \\ 0 & 1 & 0 \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 \end{vmatrix}}{\| \quad \|} = \frac{(0,\ 0,\ -\frac{1}{\sqrt{2}})}{\| \quad \|} = (0,\ 0,\ -1)$$

$$n = \frac{EYE - AT}{\| \quad \|} = \frac{(2,-2,0)}{} = \left(\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}, 0\right)$$

$$V = n \times u = \begin{vmatrix} i & j & k \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & -1 \end{vmatrix} = \left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0\right)$$

$$u = \frac{Up \times n}{\| \quad \|} = (0,\ 0,\ -1)$$

$$v = n \times u = \left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0\right)$$



0.5

EYE $p(t) = (2, 0, 0)$

AT $q(0.5) = (0, 2, 0)$

Up $(0, 1, 0)$

(a) $p(t)$와 $q(t)$ 모두 [0,1]의 범위를 가지는 매개변수 t에 의해 정의된다. $t=0.5$ 일 때 $p(t)$와 $q(t)$를 계산하라.

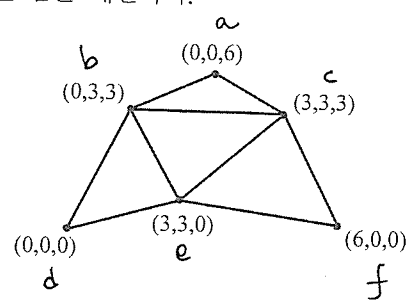
(b) $t=0.5$일 때 카메라 공간의 uvn축을 계산하라.

(c) $t=0.5$일 때의 뷰 행렬을 구성하는 4×4 이동 및 회전 행렬을 계산하라.

이동 $\begin{pmatrix} 1 & 0 & 0 & -2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$ 회전 $\begin{pmatrix} 0 & 0 & -1 & 0 \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 & 0 \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$

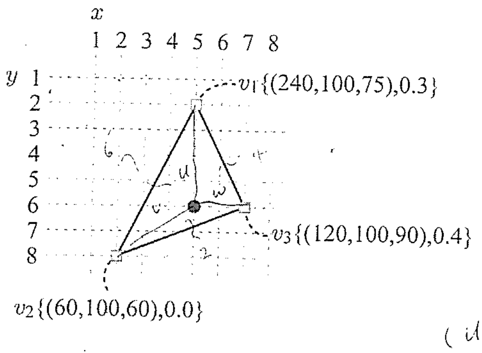
4. 아래와 같은 컨트롤 포인트 망을 가지는 2차 베지어 삼각형에서, $(u,v) = \left(\frac{1}{3}, \frac{1}{3}\right)$ 일 때 곡면상의 3차원 점을 계산하라.

$u + v + w = 1$



$S = u^2 a + 2uvb + 2wuc$
$+ v^2 d + 2vwe + w^2 f$

5. 아래 그림과 같은 스크린 공간 삼각형의 각 정점은 $\{(R, G, B), z\}$ 속성을 가진다.



$u = \frac{area(p, v_2 v_3)}{area(v_1 v_2 v_3)}$

$= \frac{2}{2+4+6} = \frac{1}{6}$

$v = \frac{4}{12} = \frac{2}{6}$

$w = \frac{6}{12} = \frac{3}{6}$

(u, v, w)

(a) 삼각형 내부의 점 (5,6)이 v_1, v_2, v_3에 대해 가지는 무게중심 좌표를 계산하라.

(b) 이 무게중심 좌표를 사용하여 (5,6)에서의 R과 z를 계산하라.

$P(R) = \frac{1}{6} \times 240 + \frac{2}{6} \times 60 + \frac{3}{6} \times 120$

$= 40 + 20 + 60$

$= 120$

3+12 15

그림 8.10-(a)에서와 같이 l_x가 l_y보다 클 때, 등방형 발자국의 샘플 점은 비등방형 발자국의 x 방향을 따라 일정 거리만큼 떨어져 분포한다. 등방형 발자국의 필터링 함수를 $\tau(t_x, t_y)$로 표기하면, 비등방형 필터링 결과는 다음과 같다.

$$\frac{1}{N}\sum_{i=1}^{N}\tau\left(t_x^0 + \frac{\partial t_x}{\partial x}\left(\frac{i}{N+1} - \frac{1}{2}\right), t_y^0 + \frac{\partial t_y}{\partial x}\left(\frac{i}{N+1} - \frac{1}{2}\right)\right) \tag{8.3}$$

여기에서 (t_x^0, t_y^0)는 픽셀 중심의 텍셀 주소를 나타낸다. 그림 8.10의 예에서 (t_x^0, t_y^0)는 $(3.5, 3.5)$이고, 0번 레벨에서의 세 샘플 점 좌표는 $\{(2.0, 2.75), (3.5, 3.5), (5.0, 4.25)\}$이며, τ는 이 세 점 각각에 대해 삼선형 보간을 수행한다.

만약 l_x가 l_y보다 작다면 식 (8.3)에서 ∂x는 ∂y로 대체되고, 이에 따라 우리는 다음과 같은 식을 얻게 된다.

$$\frac{1}{N}\sum_{i=1}^{N}\tau\left(t_x^0 + \frac{\partial t_x}{\partial y}\left(\frac{i}{N+1} - \frac{1}{2}\right), t_y^0 + \frac{\partial t_y}{\partial y}\left(\frac{i}{N+1} - \frac{1}{2}\right)\right)$$

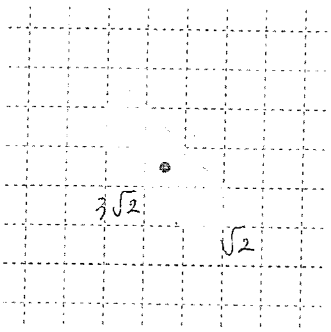
마지막으로 이미지 텍스처링이 조명(lighting)과 어떻게 결합되는지 살펴보자. 조명을 위해 다음과 같은 퐁 모델이 사용된다고 하자(이는 식 (5.11)과 동일하다).

$$max(n \cdot l, 0)s_d \otimes m_d + (max(r \cdot v, 0))^{sh} s_s \otimes m_s + s_a \otimes m_a + m_e$$

이 식에서 디퓨즈 계수 m_d는 이미지 텍스처에서 가져온 색상 값이 된다. 5.1.2절에서 논한 바와 같이, 스페큘러 계수 m_s는 별도의 텍스처에서 읽어올 수 있는데, 이 텍스처는 대개 회색조(gray-scale)로 구성된다.

연습문제

1. 다음 페이지 그림과 같은 8×8 크기의 텍스처에 픽셀이 투영되었다고 하자. 왼쪽 그림은 0번 레벨 텍스처에서의 픽셀 발자국과 그 중심을 보여준다.
   - (a) 이미지 텍스처가 회색조라고 가정하자. 오른쪽 그림은 밉맵의 0번과 1번 레벨 텍스처를 보여준다. 이와 같은 방식으로 2번과 3번 레벨 텍스처를 도시하라.
   - (b) 삼선형 보간 기법으로 밉맵을 필터링하고, λ 계산을 위해 픽셀 발자국의 긴 변 길이를 사용한다고 가정하자. 이 예에서는 어떤 레벨이 선택되는가? 각 레벨에서의 필터링 결과를 계산하라.



0번 레벨

9	9	9	9	9	7	0	1
9	9	9	9	7	9	8	3
6	5	8	9	2	3	2	1
5	8	8	7	7	0	4	1
7	8	1	1	1	1	3	3
8	5	1	5	1	1	0	2
4	9	9	9	1	2	4	8
8	3	9	9	4	1	0	0

0번 레벨

9	9	8	3
6	8	3	2
7	2	1	2
6	9	2	3

1번 레벨

8	7
6	2

2

5 3

$\log_2(3\sqrt{2})$
$= \log_2(3 \times 1.414)$
$= \log_2(4.2\cdots)$
$= 2.xx$

(c) 비등방형 필터링이 사용되고, λ 계산을 위해 픽셀 발자국의 짧은 변 길이를 사용한다고 가정하자. 어떤 레벨이 선택되는가? 왼쪽 그림의 발자국 중심점을 샘플 점으로 취급하여, 각 레벨에서의 필터링 결과를 계산하라(각 레벨에서는 겹선형 보간 기법을 사용하라).

$\log_2\sqrt{2} = 0.5$ 0레벨 4 1레벨 2 $\frac{7}{2} = 3.5$

2. 텍스처 좌표 u가 $[0,1]$의 범위를 벗어난다고 가정하자. 텍스처 어드레싱 모드가 '반복 모드'로 설정된 경우, 주어진 u를 $[0,1]$ 범위 안으로 전환하는 식을 정의하라.

$u' = u - \lfloor u \rfloor$ 1.1 → 0.1
1.2 → 0.2

3. 의료 영상과 같은 응용 분야에서는 종종 3차원 텍스처링이 필요하다. $2^l \times 2^l \times 2^l$ 해상도를 가지는 3차원 이미지를 생각해보자. 3차원 텍스처 좌표 (u,v,w)는 텍셀 주소 (t_x,t_y,t_z)로 매핑되었다고 가정하자.

(a) 2차원 이미지 텍스처가 주어졌을 때, 우리는 통상 근접점 샘플링 혹은 겹선형 보간 기법 중 하나를 택한다. 근접점 샘플링을 3차원 텍스처에 적용하기 위해, (t_x,t_y,t_z)를 어떻게 변환하는지 기술하라.

(b) 겹선형 보간 기법을 3차원 텍스처에 적용하기 위해, (t_x,t_y,t_z)를 어떻게 변환하는지 기술하라.

(c) 3차원 이미지 텍스처로 밉맵을 만들어 보자. 다운샘플링을 어떻게 수행할 것인지 기술하라. 밉맵의 최상위 노드의 해상도는 얼마인가? 밉맵은 총 몇 개의 레벨로 구성되는가?

연습문제

1. 그림 9.6과 달리, 한 점을 공유하는 삼각형들의 노멀을 사용해 그 점의 노멀을 계산하는 알고리즘을 기술하라. p17 1장 그림 1.19 모든 삼각형 노멀들의 평균

2. 그림 4.5-(a)에 보인 원통 및 그 파라미터화 결과를 보자. 원통의 축이 좌표계의 y축과 같다는 가정하에 원통의 각 정점 (x, y, z)에 대해 어떻게 탄젠트 공간을 정의할 것인지 기술하라. 단, 한 정점의 탄젠트 공간 계산 시 주위 정점 정보를 사용할 수 없다.

3. 이 장에서 기술한 패럴랙스 매핑 알고리즘은 일정 간격으로 광선을 샘플링한다. 하지만 이러한 이산적 광선 추적 기법은 종종 광선과 하이트 필드 간 교차점을 놓칠 수 있다. 언제 그러한 경우가 발생하는지, 그리고 이를 해결하는 방법은 어떤 것이 있는지 기술하라.

	R16	G16	B16	A16
RT0	발산광 색상			
RT1	노멀			깊이 값
RT2	디퓨즈 색상			AO
RT3	스페큘러 색상			

[그림 10.24] 스타크래프트 2에서 사용된 G-버퍼는 네 개의 렌더 타겟으로 구성된다[35]. 정적인 환경에서 계산된 앰비언트 오클루전은 AO로 표기되었는데, SSAO가 사용될 경우 이는 사용되지 않는다.

미뤄진 쉐이딩은 많은 대작 게임에서 성공적으로 구현되었다. 그림 10.24는 스타크래프트 2에서 사용된 G-버퍼의 구성을 보여주는데, 각 텍스처는 네 개의 채널을 가지고 있고 각 채널은 16비트 부동 소수점 형식을 가진다.

연습문제

1. 쉐이더를 사용하여 동적으로 큐브맵을 생성하는 것을 고려하자. 장면의 프리미티브들은 기하 쉐이더에 의해 복사되고 변환된다. 기하 쉐이더는 for 루프를 여섯 번 순환하는데 각 순환 단계에서 뷰 및 투영 변환을 프리미티브에 적용한다.
   (a) 각 순환 단계마다 뷰 변환은 같은가 다른가? 그 이유를 설명하라. EYE AT UP
   (b) 각 순환 단계마다 투영 변환은 같은가 다른가? 그 이유를 설명하라.
   p236 아래부분 해답 fovy aspect n, f

2. 그림 10.7-(c)에 보인 바와 같이 라이트 매핑은 대개 이미지 텍스처링과 결합된다.
   (a) 그런데 이러한 결합 결과 자체를 라이트맵에 저장하면 런타임 성능을 향상할 수도 있을 것이다. 하지만, 이는 몇 가지 문제점을 가지고 있으므로 실제 사용되지 않는다. 어떤 문제점이 있는가? (n·l)sd ⊗ md 라이트맵 이미지 텍스쳐 라이트유무 표현 X
   (b) 라이트 매핑은 종종 **다크 매핑**(dark mapping)이라고 풍자되는데, 이는 라이트 매핑을 수행한 후 이미지 텍스처 색상이 더 어두워졌기 때문이다. 왜 이런 문제가 발생하는가? 이 문제를 어떻게 해결할 수 있을까?
   투명컬러 grayscale 값 [0, 1]

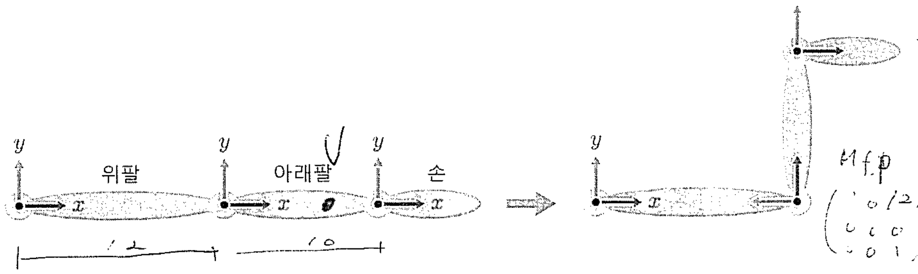
3. 쉐도우 매핑을 부적절하게 수행했을 때 부딪힐 수 있는 문제 중 하나는 이른바 **피터 팬 효과**(Peter Panning)라는 것인데, 렌더링 결과 생성된 영상에서 그림자를 만드는 물체가 그림자와 붙어 있지 않고 분리되어 나타나는 것을 말한다. 즉, 물체가 피터팬처럼 날아다니는 느낌을 준다. 왜 이런 문제가 발생하는가?
   depth offset 적용 (buffer precision 늘리)

범위는 넓다. 캐릭터가 이리저리 움직이는 물체를 응시하도록 할 때에는 IK가 캐릭터의 머리 뼈에 적용된다. 그림 11.22에서는 팔뿐 아니라 머리에도 IK가 적용되었고, 이에 따라 군인이 날아가는 공을 응시하게 된 것이다. 슈팅게임에서 군인이 적에게 총을 겨누도록 하는데 역시 IK가 사용된다.

$\begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$ $M_{f,l}$ $90°$ $\begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$ $M_{h,l}$ $-90°$ $\begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$

연습문제

1. 아래에서 왼쪽 그림은 팔의 초기 자세를 보여준다. 문제를 간단히 하기 위해, 위팔의 뼈 공간을 월드 공간이라 부르자. 아래팔과 손의 뼈 공간 원점은 월드 공간 기준으로 각각 (12,0)과 (22,0)이다. 드레스 포즈의 아래팔이 90°, 손이 −90° 회전하여 오른쪽에 보인 애니메이션 포즈를 만들었다.



$M_{f,p}$ $\begin{pmatrix} 1 & 0 & 12 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$ $M_{h,p}$ $\begin{pmatrix} 1 & 0 & 10 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$

(a) 드레스 포즈에서 아래팔과 손의 부모 변환 행렬 $M_{f,p}$와 $M_{h,p}$를 계산하라.

(b) 드레스 포즈에서 아래팔과 손의 월드 변환 행렬 $M_{f,d}$와 $M_{h,d}$를 계산하라. ↳ $(M_{u,d} M_{f,p})$ $(M_{f,d} M_{h,p})$ (3) (11, 20)

$M_{u,w} = M_{u,d} = I$

(c) $M_{f,d}^{-1}$과 $M_{h,d}^{-1}$을 계산하라. $(M_{f,p}^{-1} M_{u,d}^{-1})$ $(M_{h,p}^{-1} M_{f,d}^{-1})$

(d) 아래팔과 손의 애니메이션을 묘사하는 지역 변환 행렬 $M_{f,l}$과 $M_{h,l}$을 계산하라.

(e) 애니메이션 포즈에서 아래팔과 손의 월드 변환 행렬 $M_{f,w}$와 $M_{h,w}$를 계산하라. 식 (11, 24)

$M_{f,w} = M_{u,w} M_{f,p} M_{f,l}$ $M_{h,w} = M_{f,w} M_{h,p} M_{h,l}$

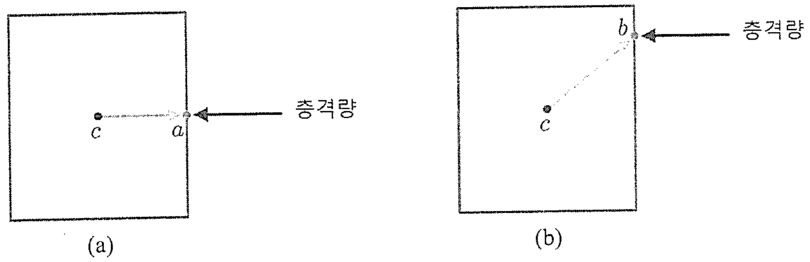
(f) 아래팔의 뼈 공간에서 (8,0)의 좌표를 가지는 정점 v를 생각해보자. 이는 아래팔과 손에 의해 영향받는다. 두 뼈의 블렌딩 가중치는 동일하다. 스키닝 알고리즘을 사용하여 애니메이션 포즈에서 v의 월드 공간 좌표를 계산하라.

$0.5 M_{f,w} v_f + 0.5 M_{h,w} v_h$ $v_f = (8, 0)$ $v_h = (-2, 0)$ 식 (11.27) 적용

2. 그림 11.22에서 날아가는 공을 캐릭터가 응시하도록 하기 위해 머리뼈와 목뼈를 잇는 관절을 사용하도록 하자. yaw pitch roll

(a) 이 관절의 자유도는 몇인가? 3 자유도

(b) 머리뼈를 회전시켜 공을 응시하도록 하는 해석적 기법의 알고리즘을 기술하라.



[그림 12.14] 무게 중심은 c로 표기되었고, a와 b는 접촉점을 나타낸다. (a) 각가속도가 생성되지 않았으므로, 물체는 충돌 이후에 회전하지 않는다. (b) 각가속도가 생성되어 물체는 충돌 이후에 회전한다.

도는 다시 적분되어 물체 **방향**(orientation)의 변화량을 결정한다. 따라서 그림 12.14-(b)의 사각형은 그림 12.14-(a)의 경우와 달리 충돌 후 회전하게 된다. 그림 12.8에서도 충돌 해결 모듈은 각가속도를 계산하였고, 그 결과 충돌 이후의 물체는 회전하면서 멀어지게 되었던 것이다.

B_1이 (8,0) 인경우

식(12.15) $J=(-6m_1, 2\sqrt{3}m_1)$

식(12.17)

식(12.14) $V_1 = ((8,0)\cdot(\frac{\sqrt{3}}{2}, -\frac{1}{2}))(\frac{\sqrt{3}}{2}, -\frac{1}{2}) = (6, -2\sqrt{3})$

$P_1' = (8m_1, 0) + (-6m_1, 2\sqrt{3}m_1) = (2m_1, 2\sqrt{3}m_1)$

연습문제

1. 그림 12.6에서 ~~B_2가 충돌 전 (8,0)의 속도로 움직일 경우~~, B_1과 B_2의 충돌 이후 속도를 계산하라.

B_1 충돌후속도 $(2, 2\sqrt{3})$ B_2 충돌후 속도 $(6, -2\sqrt{3})$

B_1으로 변경

B_2의 충격량 $= -J = (6m_2, -2\sqrt{3}m_2) = P_2'$

2. 한 장면에 m개의 움직이는 물체와 n개의 고정된 물체가 있다. 얼마나 많은 쌍에 대해 충돌 검사를 수행해야 하는가?

움직이는 물체 끼리 충돌가능. 움직이고 고정된 물체 충돌가능

$${}_mC_2 + mn = \frac{m(m-1)}{2} + mn$$

3. 그림 12.13에서 p와 q가 계산되면 이들이 실제로 삼각형의 내부에 위치하는지 여부를 가리기 위해 후처리를 해야 한다.
   (a) p 또는 q가 삼각형의 내부에 있는지 어떻게 검사할 것인가?
   (b) p가 삼각형의 내부에 있고 q가 외부에 있을 때 접촉점을 어떻게 구할 수 있는가?
   (c) p와 q가 모두 삼각형 바깥에 있을 때 이들을 연결한 선분은 삼각형과 교차할 수도 있고 교차하지 않을 수도 있다. 이 두 경우를 어떻게 구분할 수 있는가?