게임 수학 강의 노트

강영민

동명대학교

2015년 2학기

강의개요

- 교재명: *게임프로그래머를 위한* 수학과 OpenGL 프로그래밍
- 저자 - 강영민
- 출판사 - 도서출판 GS인터비전.
- 강의방식
  - 강의식 진행
  - 수시고사 10회
  - 프로그래밍 구현 실습

강의 목표

게임과 같은 3차원 콘텐츠를 설계하고 제작하는 데에 필요한 기본적인 수학 지식을 습득한다.

무엇을 다루나

주제	내용
벡터	벡터의 개념과 연산
행렬	행렬의 기학적 개념 이해와 응용
변환	변환의 개념 및 행렬 표현 이해
사원수	사원수를 이용한 변환 개념 이해
카메라	카메라 투영 행렬의 이해
조명과 재질	조명 계산 모델의 이해
충돌	기하 객체의 충돌을 감지하는 방법 이해

마음의 준비

- 수식에 겁 먹지 말자
  - $\frac{1}{n}\sum_{i=1}^{n} x_i$은 그냥 $\frac{x_1+x_2+x_3+x_4+\cdots+x_n}{n}$이다.
- 거의 매주 시험을 친다. 시험 문제는 언제나 미리 알려 준다.
  - 결석은 출석 점수를 못 받아서 무서운 것이 아니라, 시험 문제를 알지 못 하게 되어 무서운 것이다.
- 출석인정에 대하여
  - 사정이 있으면 결석을 하라. 세상에는 수업보다 중요한 일이 많다.
  - 선택의 책임은 본인 것이다. 출석 인정은 없다. 세상은 그런 곳이다.
  - 세상에 안 되는 일은 없다. 필요한 경우 진지하게 상담요청을 하라.

벡터란 무엇인가?

- 벡터의 의미
  - 벡터(vector)는 '나르다'라는 의미의 라틴어 동사 'vehere'에서 유래
  - '무엇인가를 나르는 것'이라는 의미
  - 벡터라는 것은 무엇인가를 옮겨 놓는 역할을 수행한다.
- 수학과 물리학에서의 개념
  - 크기와 방향으로 결정되는 양(量, quantity)
  - 방향량(方向量)이라고도 함.
  - 예: 힘(force)은 크기만으로는 그 성질을 온전히 표현할 수 없고, 방향도 같이 고려해야 하므로 벡터로 표현된다.

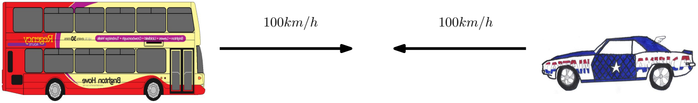
벡터

수학자들은 자연 현상의 많은 것들이 수로 표현될 수 있음을 알았는데, 하나의 숫자로 충분히 표현할 수 있지만 하나 이상의 수가 필요한 경우도 있다. 이러한 양을 벡터(vector)라고 부른다.

벡터의 개념

- 물리적 현상 등을 표현할 때 - 대상을 양(量, quantity)으로 표현
- 이 양은 스칼라(scalar) 혹은 벡터(vector)
  - 스칼라 값은 오로지 크기만으로 완전히 그 양을 표현할 수 것으로서 물체의 질량, 소요된 시간, 길이, 열량 등이 해당
  - 벡터(vector)는 이와 달리 크기와 함께 방향도 같이 존재하는 양으로 힘, 속도, 변위와 같은 양이 바로 여기에 해당

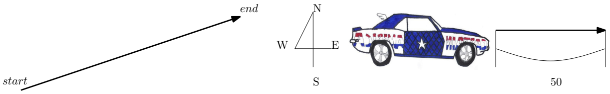
- 속도(速度, velocity)와 속력(速力, speed)
- 속도는 벡터, 속력은 스칼라



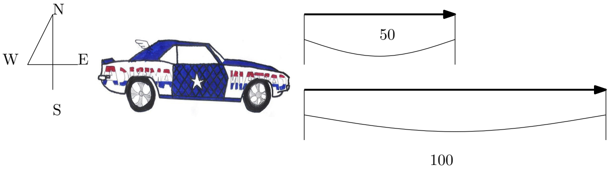
동일한 속력으로 서로 마주 보며 달리는 차량의 속도

화살표를 이용한 벡터 표현

- 벡터를 표현하는 가장 직관적인 방법은 화살표를 이용
- 화살표: 시점(始點)과 종점(終點)으로 구성
- 화살표의 방향은 벡터의 방향을 시각적으로 표현하고, 화살표의 길이는 벡터의 크기를 시각적으로 표현



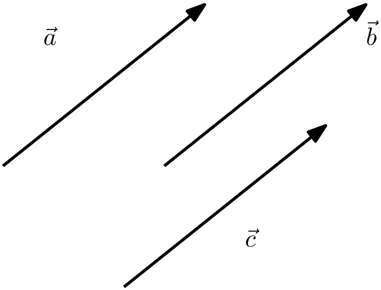
벡터의 시각적 표현과 달리는 자동차 속도 표현의 예



속력이 두 배로 늘어난 자동차의 속도

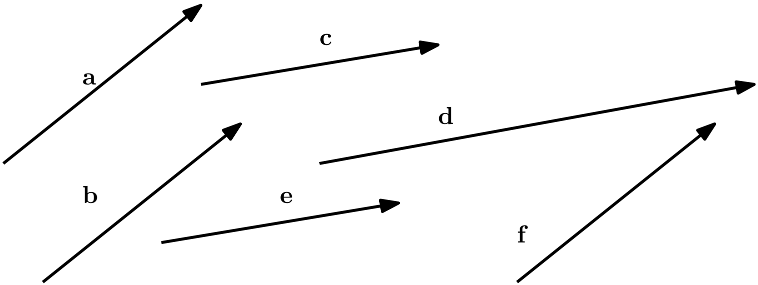
동등벡터(equivalent vector)

- 벡터의 표기법
  - $\vec{a}$, $\mathbf{a}$
- 동등벡터
  - 크기와 방향이 같으면 모두 동등한 벡터로 간주



동등벡터

동등벡터 찾기

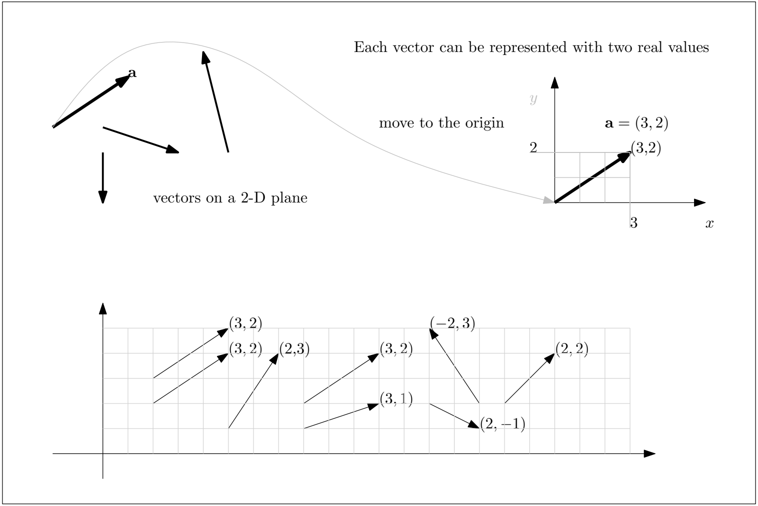


동등벡터 찾기

벡터의 수학적 표현

- 벡터: 화살표가 그려지는 공간의 차원(次元,dimension)에 따라 결정되는 개수의 성분
  - n-튜플(tuple)
  - $\mathbf{v} = (v_1, v_2, v_3, \cdots, v_n)$
  - n 개의 차원을 가진 공간에서 그려지는 화살표 = n차원 벡터

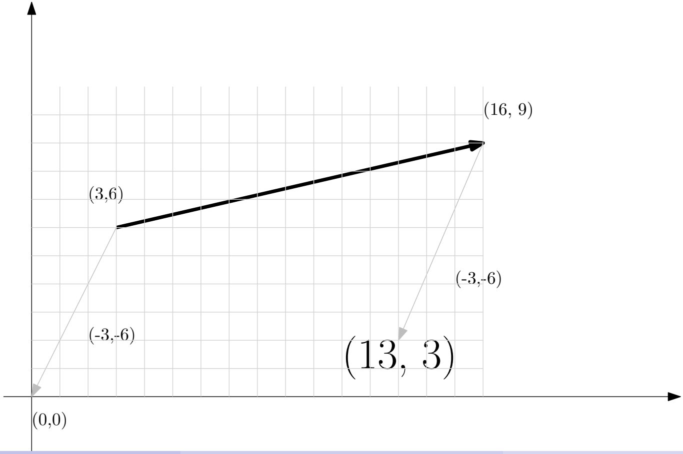
2차원 벡터



Each vector can be represented with two real values
a
move to the origin
$\mathbf{a} = (3, 2)$
y
2
(3,2)
3
x
vectors on a 2-D plane
(3, 2)
(3, 2)
(2,3)
(3, 2)
(−2, 3)
(2, 2)
(3, 1)
(2, −1)

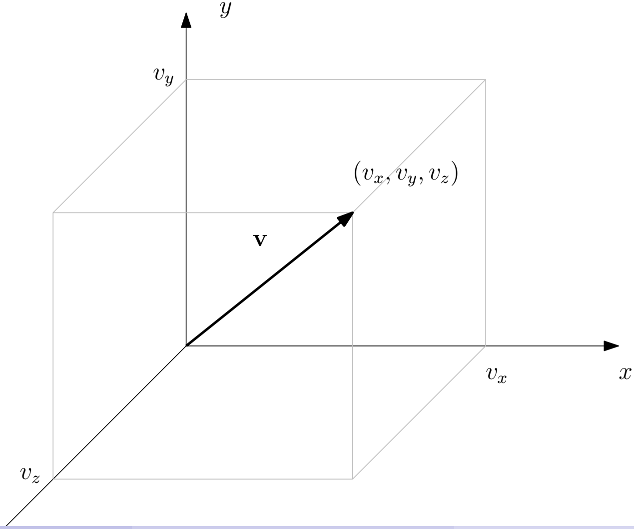
시점을 원점으로 옮기기

시점이 원점이 아닌 벡터는 시점을 원점으로 끌어 오면 된다. 벡터의 시점은 (3,6)의 좌표에 놓여있고, 끝점은 (16,9)이다. 시작점을 원점으로 옮기는 것은 (-3,-6) 만큼의 이동을 하는 것이다. 따라서 끝점은 (13,3)의 위치로 이동하게 된다. 그러므로 이 벡터는 (13,3)으로 표현된다.

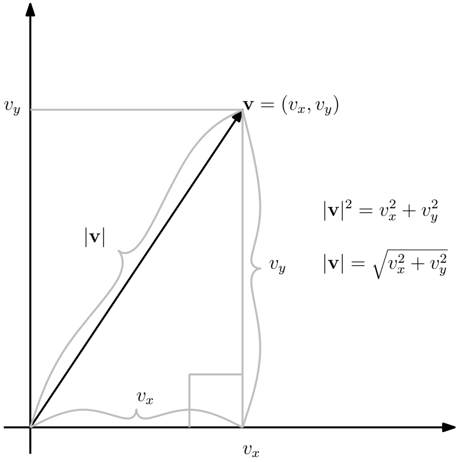


3차원 벡터

3차원 벡터는 지금까지 살펴본 2차원 벡터에 축(軸, axis)을 하나 더하기만 하면 된다.



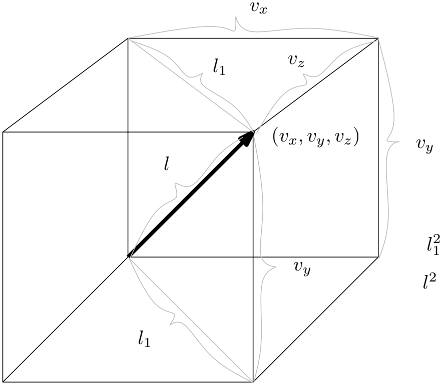
벡터의 크기



어떤 벡터 $\mathbf{v}$가 (v_x, v_y)로 표현될 때 이 벡터의 크기는 v_x와 v_y로 어떻게 구할 수 있을까? 그 값은 '길이'에 대한 상식적 정의에 따라, 스칼라 값이며 양(陽, positive)의 값이 된다. 이렇게 양의 길이(positive length)를 벡터에 할당하는 것을 놈(norm)이라 하며, 어떤 벡터 $\mathbf{v}$의 놈은 $||\mathbf{v}||$로 표현한다. 유클리드 기하에 의해 얻어지는 길이는 Euclidean Norm이라고 한다. 이는 피타고라스의 정리를 이용하여 쉽게 구할 수 있다.

3차원 벡터의 크기

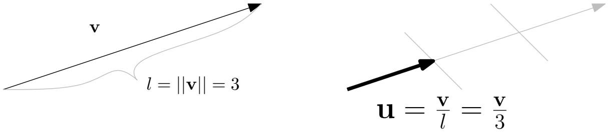
$$\mathbf{v} = (v_x, v_y, v_z)$$
$$||\mathbf{v}|| = \sqrt{v_x^2 + v_y^2 + v_z^2}$$



$$l_1^2 = v_x^2 + v_z^2$$
$$l^2 = l_1^2 + v_y^2$$

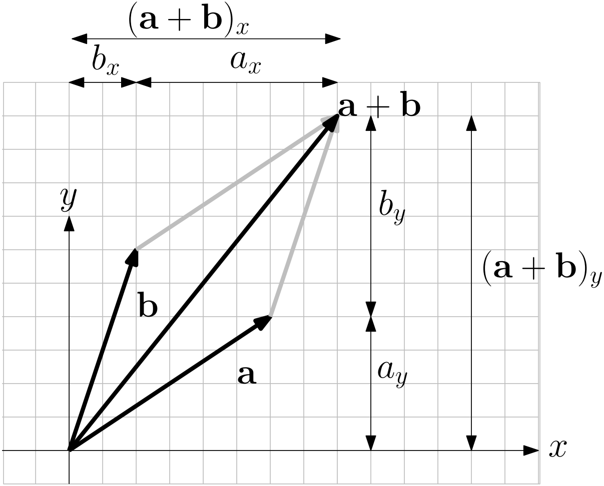
벡터의 정규화

- 정규화
  - 단위 벡터는 길이가 1인 벡터.
  - 어떤 벡터의 방향과 일치하는 단위벡터를 구하는 작업은 종종 많은 응용에서 필요.
  - 이러한 작업은 벡터의 길이를 1로 만드는 것과 같다.
  - 이를 정규화(normalization)이라고 한다.



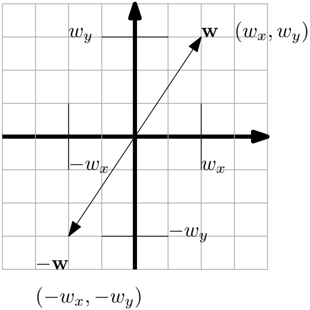
벡터의 덧셈

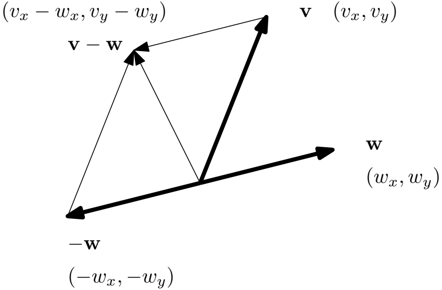
$\mathbf{v} + \mathbf{w} = (v_x + w_x, v_y + w_y, v_z + w_z)$



벡터의 뺄셈

$$\mathbf{v} - \mathbf{w} = (v_x - w_x, v_y - w_y, v_z - w_z)$$





벡터에 스칼라 곱하기

벡터는 크기만을 가진 스칼라와 곱할 수 있다. 어떤 스칼라 값 s가 있다고 하자, 이 스칼라 값과 벡터 $\mathbf{v} = (v_x, v_y, v_z)$를 곱한 $s\mathbf{v}$는 다음과 같다.

$$s\mathbf{v} = (sv_x, sv_y, sv_z)$$

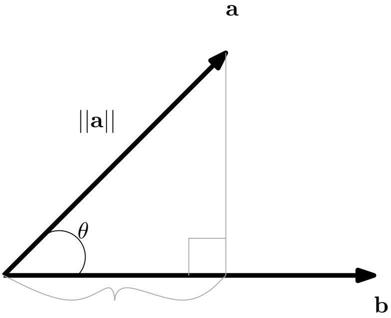
벡터의 기본적인 연산 규칙

$$
\begin{aligned}
\mathbf{a} + \mathbf{b} &= \mathbf{b} + \mathbf{a} \\
(\mathbf{a} + \mathbf{b}) + \mathbf{c} &= \mathbf{a} + (\mathbf{b} + \mathbf{c}) \\
\mathbf{a} + \vec{0} &= \vec{0} + \mathbf{a} = \mathbf{a} \\
\mathbf{a} + (-\mathbf{a}) &= \mathbf{a} - \mathbf{a} = \vec{0} \\
(k + l)\mathbf{a} &= k\mathbf{a} + l\mathbf{a} \\
(kl)\mathbf{a} &= k(l\mathbf{a}) \\
1\mathbf{a} &= \mathbf{a} \\
0\mathbf{a} &= \vec{0} \\
(-1)\mathbf{a} &= -\mathbf{a}
\end{aligned}
$$

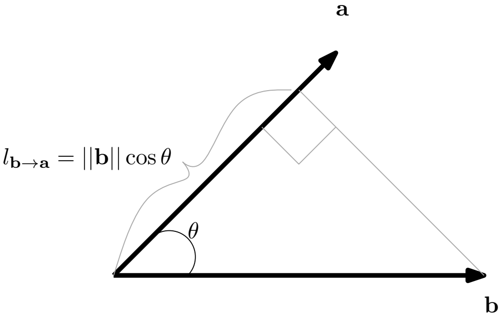
벡터의 스칼라 곱, 혹은 내적(dot product)

- 내적
  - 스칼라 곱(scalar product)라고도 부름
  - 두 개의 벡터를 피연산자(operand)로 하는 이항 연산(binary operator)로서 그 결과가 스칼라 값
  - 두 벡터 $\mathbf{a}$와 $\mathbf{b}$의 내적은 $\mathbf{a} \cdot \mathbf{b}$로 표현
  - 두 벡터가 이루는 사잇각이 θ라고 하며, 내적의 크기는 다음과 같다.
    - $\mathbf{a} \cdot \mathbf{b} = ||\mathbf{a}||||\mathbf{b}|| \cos\theta$
  - 실제 계산 방법
    - $\mathbf{a}, \mathbf{b} \in \mathbb{R}^n$
    - $\mathbf{a} \cdot \mathbf{b} = a_1b_1 + a_2b_2 + \cdots + a_nb_n = \sum_{i=1}^{n} a_ib_i$

벡터 내적의 의미



$l_{\mathbf{a}\rightarrow\mathbf{b}} = ||\mathbf{a}|| \cos\theta$



$$l_{\mathbf{a}\rightarrow\mathbf{b}} = (||\mathbf{a}||||\mathbf{b}|| \cos\theta)/||\mathbf{b}||$$
$$l_{\mathbf{a}\rightarrow\mathbf{b}} = \mathbf{a}\cdot\mathbf{b}/||\mathbf{b}||$$

$$l_{\mathbf{b}\rightarrow\mathbf{a}} = (||\mathbf{a}||||\mathbf{b}|| \cos\theta)/||\mathbf{a}||$$
$$l_{\mathbf{b}\rightarrow\mathbf{a}} = \mathbf{a}\cdot\mathbf{b}/||\mathbf{a}||$$

벡터 내적의 활용

- 코사인 함수의 특성을 통해 간단히 얻어지는 사실
  - $\theta = 0 \Rightarrow \cos\theta = 1, \mathbf{a} \cdot \mathbf{b} = ||\mathbf{a}||||\mathbf{b}||$
  - $\theta = \pi/2 \Rightarrow \cos\theta = 0, \mathbf{a} \cdot \mathbf{b} = 0$
  - $\mathbf{a} \cdot \mathbf{a} = ||\mathbf{a}||^2$
- 벡터를 이용하여 각도를 계산하거나 투영을 계산하는 데에 널리 사용
  - $\mathbf{v} \cdot \mathbf{w} = ||\mathbf{v}||||\mathbf{w}|| \cos\theta$
  - $\cos\theta = \frac{\mathbf{v} \cdot \mathbf{w}}{||\mathbf{v}||||\mathbf{w}||}$
  - $\theta = \cos^{-1} \frac{\mathbf{v} \cdot \mathbf{w}}{||\mathbf{v}||||\mathbf{w}||}$
  - $\theta = \cos^{-1} \frac{v_x w_x + v_y w_y + v_z w_y}{\sqrt{v_x^2 + v_y^2 + v_z^2}\sqrt{w_x^2 + w_y^2 + w_z^2}}$

벡터 내적의 활용

예제

어떤 두 벡터가 각각 (3,2)와 (4,1)이라고 하자. 두 벡터가 이루는 각도를 구하라.

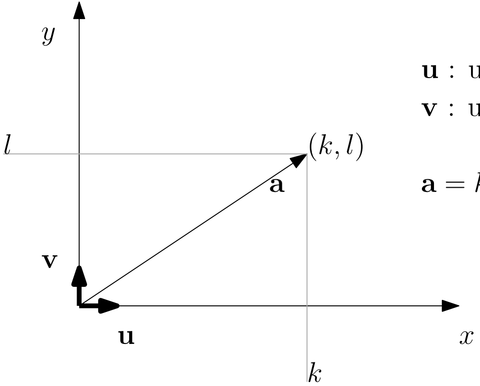
정답

두 벡터를 각각 $\mathbf{v}$와 $\mathbf{w}$로 표현하자. 두 벡터의 내적 $\mathbf{v} \cdot \mathbf{w}$는 $3 \cdot 4 + 2 \cdot 1$, 즉 14이다. 각각의 길이는 $||\mathbf{v}|| = \sqrt{9+4} = \sqrt{13}$과 $||\mathbf{w}|| = \sqrt{16+1} = \sqrt{17}$이다. 따라서 두 벡터의 사이각은 다음과 같다.

$$\theta = \cos^{-1} \frac{14}{\sqrt{13}\sqrt{17}} = \cos^{-1} \frac{14}{\sqrt{221}} \simeq \cos^{-1} 0.94174191159484 \simeq 19.65^\circ$$

좌표축과 좌표 - 1/3 기본 의미

- $\mathbf{a} = (k, l)$로 표현된다는 것은 xy 좌표계에서 기저벡터가 되는 x축 단위벡터를 $\mathbf{u}$와 y축 단위벡터를 $\mathbf{v}$를 다음과 같이 합성한 것



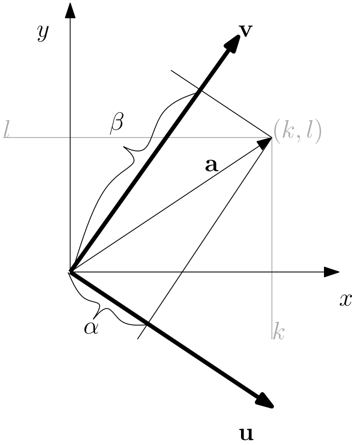
$\mathbf{u}$: unit vector along x axis

$\mathbf{v}$: unit vector along y axis

$$\mathbf{a} = k\mathbf{u} + l\mathbf{v}$$

좌표축과 좌표 - 2/3 새로운 축의 정의

- 새로운 직교 좌표계를 고려해 보자. 여기서는 두 축이 $\mathbf{u}$와 $\mathbf{v}$
- $\mathbf{a}$의 $\mathbf{u}$ 축 투영 길이 α, $\mathbf{v}$ 축 투영 길이 β 계산
- 이 두 축을 기준으로 하는 좌표계에서는 $\mathbf{a}$가 (α, β)의 좌표로 표현됨



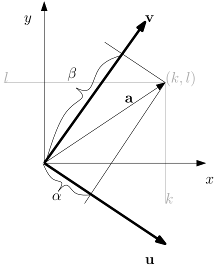
$\mathbf{u}$: an axis

$\mathbf{v}$: another axis perpendicular to $\mathbf{u}$

$\mathbf{a} = \alpha\mathbf{u} + \beta\mathbf{v}$

좌표축과 좌표 - 3/3 내적을 이용한 투영 길이 구하기

- $\mathbf{a}$가 축 $\mathbf{u}$와 $\mathbf{v}$ 방향으로 가지는 길이 α와 β는 어떻게 구하나
  - 내적을 이용
  - α는 $\mathbf{a}$를 $\mathbf{u}$ 방향으로 투영한 그림자의 길이 $= \mathbf{a} \cdot \mathbf{u}/||\mathbf{u}||$
  - 축은 단위 벡터로 표현하므로 $||\mathbf{u}|| = 1$. 따라서 $\alpha = \mathbf{a} \cdot \mathbf{u}$
  - 비슷한 방법으로 $\beta = \mathbf{a} \cdot \mathbf{v}$
  - $\mathbf{u}$와 $\mathbf{v}$를 축으로 하는 좌표계에서 $\mathbf{a}$는 $(\mathbf{a} \cdot \mathbf{u}, \mathbf{a} \cdot \mathbf{v})$

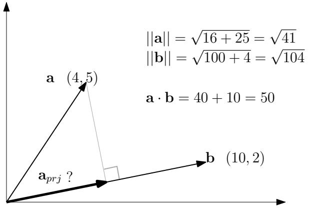


$\mathbf{u}$: an axis
$\mathbf{v}$: another axis perpendicular to $\mathbf{u}$

$\mathbf{a} = \alpha\mathbf{u} + \beta\mathbf{v}$

예제

그림처럼 어떤 벡터 $\mathbf{a}$가 (4,5)이고, 다른 벡터 $\mathbf{b}$는 (10,2)라고 하자. 이때 벡터 $\mathbf{a}$를 $\mathbf{b}$ 위에 수직방향으로 내린 그림자가 되는 벡터 $\mathbf{a}_{prj}$을 구하라.



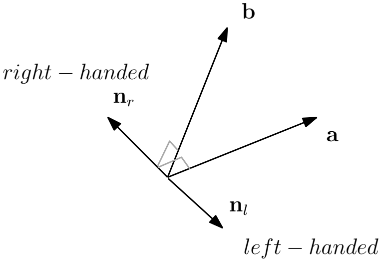
정답

$$l = \mathbf{a} \cdot \mathbf{b} / ||\mathbf{b}||$$

$$\mathbf{a}_{prj} = l\tilde{\mathbf{b}} = \frac{50}{104}(10,2) \simeq (4.8, 0.96)$$

벡터의 외적(外積) - 의미

- 벡터의 외적(cross product)
  - 벡터 곱(vector product): 두 벡터를 피연산자로 하는 이항연산으로 그 결과가 벡터
  - 벡터를 곱해 행렬을 얻는 외적(outer product)과 용어의 혼동이 있음. 여기서는 결과가 벡터인 곱
- 표현
  - 두 벡터 $\mathbf{a}$와 $\mathbf{b}$의 외적은 $\mathbf{a} \times \mathbf{b}$로 표현
  - 그 결과는 벡터이므로 $k\mathbf{n}$ ($\mathbf{n}$은 $\mathbf{a}$와 $\mathbf{b}$에 동시에 수직인 단위벡터)
  - 동시에 수직인 벡터는 두 개가 존재. 좌표계에 의해 결정됨.

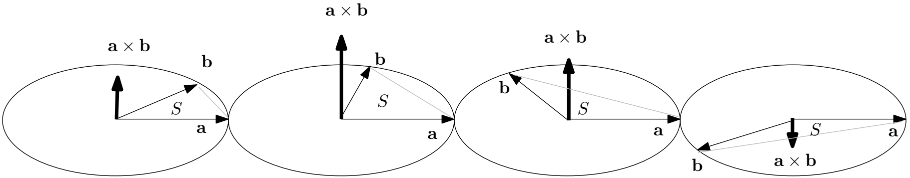


벡터의 외적(外積) 의미의 수학적 표현

- 두 벡터의 외적
  - 외적의 크기 k
    - 두 벡터의 크기와 사잇각의 사인(sine) 값에 비례
    - $k = ||\mathbf{a}||\mathbf{b}|| \sin\theta$
  - 외적의 방향 $\mathbf{n}$
    - $\mathbf{n}$: 두 벡터에 수직인 방향 벡터
- 따라서 두 벡터의 외적은 다음과 같이 표현할 수 있다.
  - $\mathbf{a} \times \mathbf{b} = ||\mathbf{a}||\mathbf{b}|| \sin\theta\mathbf{n}$

외적이 가진 의미

- 외적을 표현하는 식 $\mathbf{a} \times \mathbf{b} = ||\mathbf{a}|||\mathbf{b}|| \sin\theta \mathbf{n}$의 의미
  - 외적은 두 벡터에 동시에 수직한 벡터
  - 크기는 두 벡터가 수직일 때에 최대, 같은 방향이나 반대방향일 때 최소
  - 외적의 크기는 벡터 $\mathbf{a}$와 $\mathbf{b}$의 끝점을 연결한 삼각형의 넓이 S에 비례
    - $||\mathbf{a} \times \mathbf{b}|| = \frac{1}{2}S$



외적의 계산

- 3차원 벡터의 외적을 구하기
  - $\mathbf{a} = (a_1, a_2, a_3), \mathbf{b} = (b_1, b_2, b_3)$
  - 두 벡터의 외적: $\mathbf{a} \times \mathbf{b} = (a_2 b_3 - a_3 b_2, a_3 b_1 - a_1 b_3, a_1 b_2 - a_2 b_1)$
- "행렬"을 이용한 곱셈으로 구하기
  - "반대칭(skew-symmetric 혹은 antisymmetric)" 행렬 이용

$$\mathbf{A}^* = \begin{pmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{pmatrix}$$

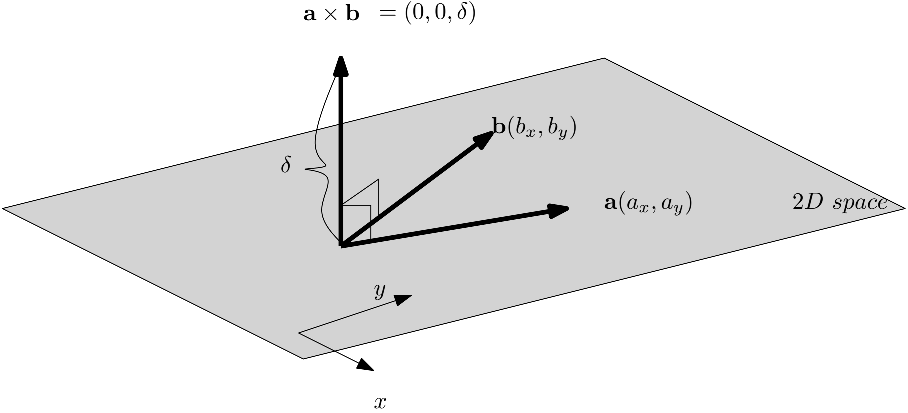
$$\mathbf{A}^*\mathbf{b} = \begin{pmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{pmatrix} \mathbf{b} = (a_2 b_3 - a_3 b_2, a_3 b_1 - a_1 b_3, a_1 b_2 - a_2 b_1)$$

외적의 연산 법칙들

$$\mathbf{a} \times \mathbf{b} = -\mathbf{b} \times \mathbf{a}$$
$$\mathbf{a} \times (\mathbf{b} \times \mathbf{c}) = (\mathbf{a} \times \mathbf{b}) \times \mathbf{c}$$
$$(\mathbf{a} + \mathbf{b}) \times \mathbf{c} = (\mathbf{a} \times \mathbf{c}) + (\mathbf{b} \times \mathbf{c})$$
$$k(\mathbf{a} \times \mathbf{b}) = k\mathbf{a} \times \mathbf{b} = \mathbf{a} \times k\mathbf{b}$$
$$\mathbf{a} \times \vec{0} = \vec{0} \times \mathbf{a} = \vec{0}$$
$$\mathbf{a} \times \mathbf{a} = \vec{0}$$
$$\mathbf{a} \cdot (\mathbf{a} \times \mathbf{b}) = \vec{0}$$
$$\mathbf{b} \cdot (\mathbf{a} \times \mathbf{b}) = \vec{0}$$

2차원 공간에서의 외적과 외적의 응용

- 2차원 공간의 두 벡터 $\mathbf{a} = (a_x, a_y)$와 $\mathbf{b} = (b_x, b_y)$의 외적은?
  - 그림의 회색 면은 2차원 공간의 일부
  - 두 벡터 $\mathbf{a}$와 $\mathbf{b}$의 외적은 2차원 공간 밖에서 정의
  - 축 z가 필요하며, 이 z 축 성분으로만 표현



2차원 공간에서의 외적과 외적의 응용

- 2차원 벡터의 외적이 2차원 공간 밖에 정의가 되고, 이것은 3차원 벡터라고 볼 수 있다.
- 2차원 벡터 $\mathbf{a} = (a_x, a_y)$와 $\mathbf{b} = (b_x, b_y)$를 3차원 벡터로 가정
  - $\mathbf{a} = (a_x, a_y, 0)$
  - $\mathbf{b} = (b_x, b_y, 0)$
  - $\mathbf{a} \times \mathbf{b} = (0, 0, a_x b_y - a_y b_x)$
- z 성분의 값으로 알 수 있는 것들 (이를 δ라고 하자)
  - $\delta > 0$인 경우는 $\mathbf{b}$가 $\mathbf{a}$의 진행 방향을 기준으로 왼쪽에 있음
  - $\delta < 0$인 경우는 오른쪽
  - 절대값은 두 벡터 사이에 만들어지는 삼각형의 크기에 비례

삼각형 넓이 구하기

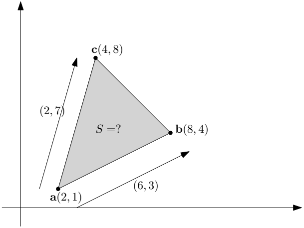
예제

꼭지점 좌표가 (2,1), (8,4), (4,8)인 삼각형의 넓이 S를 구하라.

정답

꼭지점들을 각각 $\mathbf{a}, \mathbf{b}, \mathbf{c}$로 표현하자. 우리는 $\mathbf{a}$에서 $\mathbf{b}$로 가는 벡터를 구할 수 있고, $\mathbf{u} = (6, 3)$가 된다. 비슷한 방식으로 $\mathbf{a}$에서 $\mathbf{c}$로 가는 벡터는 $\mathbf{v} = (2, 7)$이다. 삼각형의 넓이는 이 두 벡터의 외적이 가지는 크기의 반이다.

$$S = \frac{1}{2}||\mathbf{a} \times \mathbf{b}|| = \frac{6 \cdot 7 - 3 \cdot 2}{2} = 18$$



외적과 평면

- 외적의 또 다른 응용
  - 평면 표현
    - 평면은 그 평면 위의 삼각형으로 표현 가능 = 3 개의 점 = 9 개의 원소
    - 좀 더 효율적인 방법 = 법선 벡터를 이용하기
    - 법선벡터 = 평면이 바라보는 방향을 나타내는 벡터
    - 법선벡터가 나타내는 것은 하나의 평면이 아니라 동일한 방향을 쳐다보는 모든 평면
    - 평면의 표현 = (법선벡터, 평면이 지나는 점) : 6 개의 원소로 표현 가능
    - 법선벡터 구하기: 벡터의 외적을 이용

행렬

- 행렬의 역사
  - 1차 방정식의 풀이에 아주 오래 전부터 사용
  - 그 특성이 정확히 파악되지 않고 1800년대까지는 배열(array)이라는 이름으로 알려짐
  - 기원전 10세기에서 기원전 2세기 사이에 여러 세대에 걸쳐 쓰여진 중국의 구장산술(九章算術)에 연립 방정식을 풀기 위해 소개
  - 판별식의 개념 등장
  - 1545년에야 이탈리아 수학자 지롤라모 카르다노(Girolamo Cardano)가 그의 저서 "위대한 기술(Ars Magna)"를 통해 유럽에 전함
  - 오랜 기간 동안 많은 수학자들이 이 행렬을 다루며 다양한 성질을 발견
  - 행렬은 공간을 나루는 데에 필요한 유용한 도구
  - 공간 내의 점들을 이떤 위치에서 다른 위치로 옮겨 놓는 다양한 변환이 행렬을 이용하여 표현됨

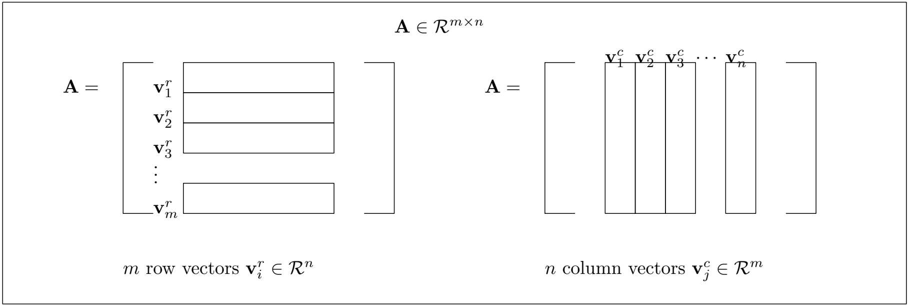
행렬이란 무엇인가

- 행렬은 2차원으로 배열된 수
- 가로 줄을 행(row), 세로 줄을 열(column)
- m 개의 행과 n 개의 열로 이루어진 행렬은 $\mathbf{A}$는 다음의 모양
- $\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1j} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2j} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ a_{i1} & a_{i2} & \cdots & a_{ij} & \cdots & a_{in} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mj} & \cdots & a_{mn} \end{bmatrix}$
- $m \times n$ 행렬이라고 하며 $\mathbf{A} \in \mathbb{R}^{m \times n}$로 표현
- 각 행은 n 개의 원소를 가진 행벡터(row vector)
- 각 열은 m 개의 원소를 가진 열벡터(column vector)

행벡터와 열벡터

$\mathbf{A}$를 행벡터 $\mathbf{v}_i^r$로 표현하면 다음과 같다. $\mathbf{A} = \begin{bmatrix} \mathbf{v}_1^r \\ \mathbf{v}_2^r \\ \vdots \\ \mathbf{v}_m^r \end{bmatrix}$

$\mathbf{A}$를 열벡터 $\mathbf{v}_j^c$로 표현하면 다음과 같다. $\mathbf{A} = \begin{bmatrix} \mathbf{v}_1^c & \mathbf{v}_2^c & \cdots & \mathbf{v}_n^c \end{bmatrix}$



정사각 행렬 - square matrix

- 정방행렬, 혹은 정사각형 행렬은 행과 열의 수가 동일한 행렬.
  - $\mathbf{A} \in \mathbb{R}^{n \times n}$ 인 행렬
  - 정방행렬은 물리 문제에서 운동을 다루거나 그래픽스에서 변환을 다룰 때에 빈번히 나타남
- 다음 행렬 $\mathbf{A}$ 는 3×3 의 정사각 행렬이다.
  - $\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$

전치 행렬 - transpose

- 어떤 행렬 $\mathbf{A}$ 의 전치행렬은 $\mathbf{A}^{\mathrm{T}}$
  - $\mathbf{B} = \mathbf{A}^{\mathrm{T}} \Rightarrow b_{ij} = a_{ji}$
  - 따라서, $m \times n$ 행렬의 전치는 $n \times m$ 행렬
  - $\mathbf{A} \in \mathbb{R}^{m \times n} \Rightarrow \mathbf{B} \in \mathbb{R}^{n \times m}$
- 전치행렬의 성질
  - $(\mathbf{A}^{\mathrm{T}})^{\mathrm{T}} = A$
  - $(\mathbf{A} + \mathbf{B})^{\mathrm{T}} = \mathbf{A}^{\mathrm{T}} + \mathbf{B}^{\mathrm{T}}$
  - $(k\mathbf{A})^{\mathrm{T}} = k\mathbf{A}^{\mathrm{T}}$
  - $(\mathbf{AB})^{\mathrm{T}} = \mathbf{B}^{\mathrm{T}}\mathbf{A}^{\mathrm{T}}$
- 그래픽스(graphics) 분야에서 매우 유용한 사용 방법은 회전 변환 행렬의 역행렬을 구할 때
- 3차원 공간의 회전 변환은 정규직교(orthonormal) 특성
- 정규직교 행렬의 역행렬은 그 행렬의 전치임이 알려져 있음

대각 행렬 - diagonal matrix

- 대각성분은 어떤 행렬 $\mathbf{A}$의 i 행 j 열 성분을 a_{ij}라고 표현할 때, $i = j$인 성분
- 대각행렬은 대각성분을 제외한 다른 모든 성분의 값이 0인 행렬이다
  - $\mathbf{A} = \begin{bmatrix} a_{11} & 0 & 0 \\ 0 & a_{22} & 0 \\ 0 & 0 & a_{33} \end{bmatrix}$
- 대각행렬을 다른 행렬과 곱했을 때의 성질
  - 어떤 3×3 대각행렬을 $\mathbf{D}$, 1,2,3 행 대각성분은 d_1, d_2, d_3
  - 다른 어떤 3×3 행렬 $\mathbf{A}$와 $\mathbf{D}$의 곱
    - $\mathbf{DA} = \begin{bmatrix} d_1a_{11} & d_1a_{12} & d_1a_{13} \\ d_2a_{21} & d_2b_{22} & d_2b_{23} \\ d_3a_{31} & d_3b_{32} & d_3b_{33} \end{bmatrix}$
    -
    - $\mathbf{AD} = \begin{bmatrix} d_1a_{11} & d_2a_{12} & d_3a_{13} \\ d_1a_{21} & d_2b_{22} & d_3b_{23} \\ d_1a_{31} & d_2b_{32} & d_3b_{33} \end{bmatrix}$

대각 행렬과 벡터의 곱 (1/2)

대각행렬 $\mathbf{D}$와 열벡터 $\mathbf{v} \in \mathbb{R}^3$이 $(v_1, v_2, v_3)^{\mathrm{T}}$의 곱

$$\mathbf{Dv} = \begin{bmatrix} a_{11} & 0 & 0 \\ 0 & a_{22} & 0 \\ 0 & 0 & a_{33} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} d_1 v_1 \\ d_2 v_2 \\ d_3 v_3 \end{bmatrix}$$

대각 행렬과 벡터의 곱 (2/2)

행벡터 $\mathbf{v}^{\mathrm{T}}$와 대각행렬의 곱

$$\mathbf{v}^{\mathrm{T}}\mathbf{D} = \begin{bmatrix} v_1 & v_2 & v_3 \end{bmatrix} \begin{bmatrix} a_{11} & 0 & 0 \\ 0 & a_{22} & 0 \\ 0 & 0 & a_{33} \end{bmatrix} = \begin{bmatrix} d_1 v_1 & d_2 v_2 & d_3 v_3 \end{bmatrix}$$

행렬의 덧셈과 뺄셈

두 행렬의 덧셈과 뺄셈은 동일한 크기의 행렬 사이에 정의됨

$$\mathbf{A} \in \mathbb{R}^{m \times n} \wedge \mathbf{A} + \mathbf{B} = \mathbf{C} \Rightarrow \mathbf{B}, \mathbf{C} \in \mathbb{R}^{m \times n}$$

$$\mathbf{A} \in \mathbb{R}^{m \times n} \wedge \mathbf{A} - \mathbf{B} = \mathbf{D} \Rightarrow \mathbf{B}, \mathbf{D} \in \mathbb{R}^{m \times n}$$

행렬과 덧셈과 뺄셈은 동일한 행과 열에 있는 성분을 서로 더하고, 빼서 원래의 자리에 기록

$$\mathbf{A} + \mathbf{B} = \mathbf{C} \Rightarrow c_{ij} = a_{ij} + b_{ij}$$

$$\mathbf{A} - \mathbf{B} = \mathbf{D} \Rightarrow d_{ij} = a_{ij} - b_{ij}$$

행렬의 곱셈

- $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\mathbf{B} \in \mathbb{R}^{n \times x}$
  - $\mathbf{AB} = \mathbf{C} \in \mathbb{R}^{m \times x}$

- $\mathbf{A}$의 앞에 $\mathbf{B}$를 곱할 경우
  - $\mathbf{B} \in \mathbb{R}^{x \times m}$ 이어야 함
  - 그 결과는 $\mathbf{BA} = \mathbf{D} \in \mathbb{R}^{x \times n}$

- $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\mathbf{B} \in \mathbb{R}^{n \times x}$
- $\mathbf{AB} = \mathbf{C} \in \mathbb{R}^{m \times x}$
  - $c_{ij} = a_{i1}b_{1j} + a_{i2}b_{2j} + \cdots + a_{in}b_{nj}$
  - $c_{ij} = \sum_{k=1}^{n} a_{ik}b_{kj}$

행렬 $\mathbf{A}$의 i 번째 행 벡터를 $\mathbf{A}_{i,*}$ 라고 하고, j 번째 열 벡터를 $\mathbf{B}_{*,j}$ 라고 하면, 위의 식은 다음과 같이 다시 쓸 수 있다.

$$c_{ij} = \mathbf{A}_{i,*}^{\mathrm{T}} \cdot \mathbf{B}_{*,j} \quad (1)$$

행렬과 스칼라의 곱

행렬에 스칼라를 곱하는 연산은 해당 스칼라 값을 행렬의 모든 원소에 곱하면 된다.

$$k\mathbf{A} = \mathbf{B} \Rightarrow b_{ij} = ka_{ij}$$

행렬 덧셈과 뺄셈의 연산 법칙

행렬은 다음과 같은 연산 법칙을 가진다.

$$\mathbf{A} + \mathbf{B} = \mathbf{B} + \mathbf{A}$$

$$(\mathbf{A} + \mathbf{B}) + \mathbf{C} = \mathbf{A} + (\mathbf{B} + \mathbf{C})$$

$$\mathbf{A} + 0 = 0 + \mathbf{A} = \mathbf{A}$$

$$\mathbf{A} + (-\mathbf{A}) = 0$$

이때, 0은 $\mathbf{A}$와 같은 차원의 행렬로 모든 원소가 0인 행렬이다.

행렬 스칼라 곱의 연산 법칙

$$(k+l)\mathbf{A} = k\mathbf{A} + l\mathbf{A}$$

$$k(\mathbf{A}+\mathbf{B}) = k\mathbf{A} + k\mathbf{B}$$

$$(kl)\mathbf{A} = k(l\mathbf{A})$$

$$(-1)\mathbf{A} = -\mathbf{A}$$

$$0\mathbf{A} = 0$$

행렬 곱셈의 연산 법칙

$$\mathbf{AB} \neq \mathbf{BA}$$

$$\mathbf{A}(\mathbf{BC}) = (\mathbf{AB})\mathbf{C}$$

$$\mathbf{A}(\mathbf{B} + \mathbf{C}) = \mathbf{AB} + \mathbf{AC}$$

$$(\mathbf{A} + \mathbf{B})\mathbf{C} = \mathbf{AC} + \mathbf{BC}$$

$$\mathbf{AI} = \mathbf{IA} = \mathbf{A}$$

$$k\mathbf{AB} = (k\mathbf{A})\mathbf{B} = \mathbf{A}k\mathbf{B}$$

$\mathbf{I}$는 항등행렬

행렬식 1/2

- 행렬식은 정방행렬에서 정의된다.
- 어떤 행렬 $\mathbf{A}$의 행렬식은 $det\mathbf{A}$, $det(\mathbf{A})$, 또는 $|\mathbf{A}|$로 표현
- 행렬식을 계산하기 위해 필요한 개념
  - 소행렬식(minor)
  - 여인자(cofactor)
- 소행렬식
  - $\mathbf{A} \in \mathbb{R}^{m \times n}$: 이 행렬은 $m \times n$개의 소행렬식(minor) M_{ij}를 가짐
  - 각 M_{ij}는 $\mathbf{A}$ 행렬의 i 행 벡터 전체와 j 열 벡터 전체를 제거하고 얻어지는 행렬($\in \mathbb{R}^{m-1 \times n-1}$)의 행렬식
- 여인자
  - 행렬 $\mathbf{A}$의 여인자는 소행렬식이 구해지는 위치마다 결정
  - 다음과 같이 정의되는 $m \times n$개의 여인자 C_{ij}가 존재
  - $C_{ij} = (-1)^{i+j} M_{ij}$

행렬식 2/2

- 행렬 $\mathbf{A} \in \mathbb{R}^{m \times n}$ 의 i 행, j 열 여인자 C_{ij} 를 이용한 행렬식 계산

$$\begin{aligned} det\mathbf{A} &= |\mathbf{A}| &= \sum_{j=1}^{n} A_{1j}C_{1j} = \sum_{j=1}^{n} A_{2j}C_{2j} = \cdots = \sum_{j=1}^{n} A_{mj}C_{mj} \\ & &= \sum_{i=1}^{m} A_{i1}C_{i1} = \sum_{i=1}^{m} A_{i2}C_{i2} = \cdots = \sum_{j=1}^{n} A_{in}C_{in} \end{aligned} \quad (2)$$

- $\mathbf{A}$ 의 임의의 행 벡터 $\mathbf{A}_{i,*}$ 와 $\mathbf{C}$ 의 동일 위치 행 벡터 $\mathbf{C}_{i,*}$ 의 내적
- $\mathbf{A}$ 의 임의의 열 벡터 $\mathbf{A}_{*,j}$ 와 $\mathbf{C}$ 의 동일 위치 열 벡터 $\mathbf{C}_{*,j}$ 의 내적

$$det\mathbf{A} = |\mathbf{A}| = \mathbf{A}_{i,*}\mathbf{C}_{i,*}^{\mathrm{T}} = \mathbf{A}_{*,j}^{\mathrm{T}}\mathbf{C}_{*,j} \quad (3)$$

예제

예제

다음 행렬의 행렬식을 구하라. $\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$

정답

여인자 M_{ij} 를 구한다.

$$\mathbf{M} = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} = \begin{bmatrix} det\,[A_{22}] & det\,[A_{21}] \\ det\,[A_{12}] & det\,[A_{11}] \end{bmatrix} = \begin{bmatrix} A_{22} & A_{21} \\ A_{12} & A_{11} \end{bmatrix}$$

여인자의 정의에 따라, 여인자로 구성된 행렬 $\mathbf{C}$는 다음과 같다.

$$\mathbf{C} = \begin{bmatrix} (-1)^{1+1}A_{22} & (-1)^{1+2}A_{21} \\ (-1)^{2+1}A_{12} & (-1)^{2+2}A_{11} \end{bmatrix} = \begin{bmatrix} A_{22} & -A_{21} \\ -A_{12} & A_{11} \end{bmatrix}$$

$\mathbf{A}$ 의 임의의 행 벡터를 선택할 수 있으므로 우선 1행을 가지고 오자.
그리고 여인자 행렬 $\mathbf{C}$ 의 1행을 가지고 와서, 두 행 벡터의 내적을 구하면 다음과 같다.

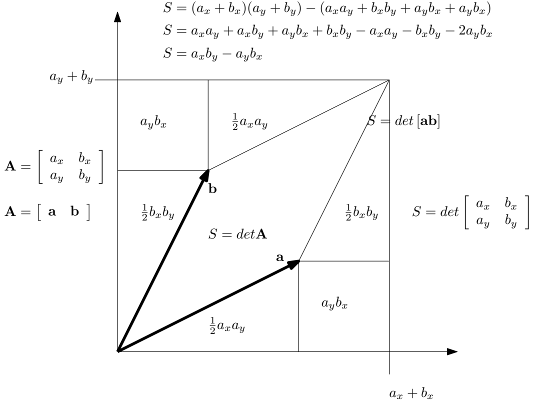
$$det\mathbf{A} = \mathbf{A}_{1,*}\mathbf{C}_{1,*}^{\mathrm{T}} = A_{11}A_{22} + A_{12}(-A_{21})$$

행렬식의 기하적 의미

두 열 벡터 $\mathbf{a} = (a_x a_y)^{\mathrm{T}}$ 와 $\mathbf{b} = (b_x, b_y)^{\mathrm{T}}$ 를 열로 하는 행렬 $\mathbf{A}$

$$\mathbf{A} = \begin{bmatrix} a_x & b_x \\ a_y & b_y \end{bmatrix}$$

이 두 벡터를 두 개의 변으로 하는 평행사변형의 넓이가 행렬 $\mathbf{A}$ 의 행렬식

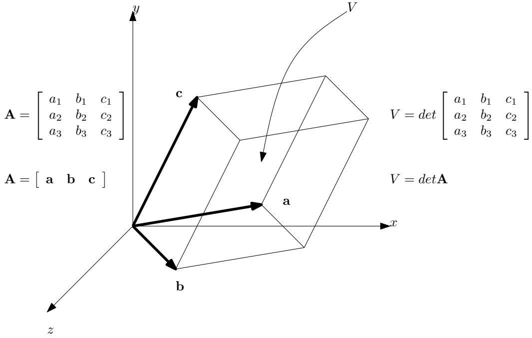


3×3 행렬의 행렬식 의미

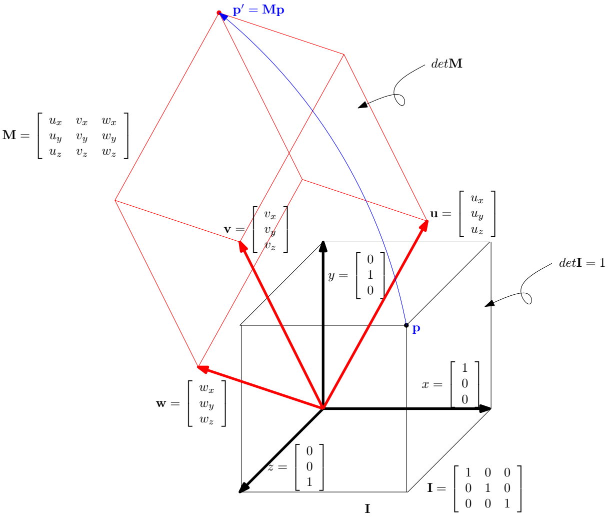
$\mathbf{A} \in \mathbb{R}^{3\times3}$는 세 개의 벡터 $\mathbf{a}, \mathbf{b}, \mathbf{c}$를 포함

$$\mathbf{A} = \begin{bmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{bmatrix} = \begin{bmatrix} \mathbf{a} & \mathbf{b} & \mathbf{c} \end{bmatrix}$$

이 세 개의 벡터들이 만드는 평행육면체의 크기가 세 개의 벡터들로 구성된 행렬의 행렬식



행렬과 행렬식의 기하적 의미



행렬식의 특성

몇 가지 기억해 둘 행렬식의 특성은 다음과 같다.

$$|\mathbf{A}| = |\mathbf{A}^{\mathrm{T}}|$$

$$\mathbf{A} \in \mathbb{R}^{n \times n} \Rightarrow |k\mathbf{A}| = k^n |\mathbf{A}|$$

$$|\mathbf{AB}| = |\mathbf{A}||\mathbf{B}|$$

역행렬

- 역행렬은 정방행렬에만 존재
- $\mathbf{A}$의 역행렬이 존재한다면, 이 역행렬을 $\mathbf{A}^{-1}$로 표현
- 역행렬 $\mathbf{A}^{-1}$은 다음과 같은 조건을 만족
  - $\mathbf{A}\mathbf{A}^{-1} = \mathbf{I}$
  - $\mathbf{A}^{-1}\mathbf{A} = \mathbf{I}$

- 역행렬이 존재하는 행렬을 가역행렬(invertible matrix)
- 역행렬이 존재하지 않는 행렬은 특이행렬(singular matrix)

- 의사 역행렬(pseudo-inverse)
  - 행렬 $\mathbf{A}$가 정방행렬이 아니고 $\mathbb{R}^{m\times n}$에 속한다고 하자. 다른 어떤 행렬 $\mathbf{B}$가 $\mathbb{R}^{n\times m}$에 속하면, 두 행렬의 곱 $\mathbf{AB}$는 $\mathbb{R}^{m\times m}$에 속하는 정방행렬이 된다. 만약 $\mathbf{AB} = \mathbf{I} \in \mathbb{R}^{n\times n}$이라면, $\mathbf{B}$를 $\mathbf{A}$의 의사 역행렬(pseudo-inverse)라고 한다.

역행렬의 계산

- 역행렬의 계산은 수반행렬(adjoint matrix)를 이용하여 쉽게 정의
  - 행렬 $\mathbf{A}$의 수반행렬: 여인자 C_{ij}를 성분으로 하는 행렬 $\mathbf{C}$의 전치(transpose)

$$adj\,\mathbf{A} = \begin{pmatrix} C_{11} & C_{21} & \cdots & C_{n1} \\ C_{12} & C_{22} & \cdots & C_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ C_{1n} & C_{2n} & \cdots & C_{nn} \end{pmatrix} = \mathbf{C}^{\mathrm{T}}$$

- 수반행렬을 행렬의 행렬식으로 나누면 역행렬이 된다.

$$\mathbf{A}^{-1} = \frac{adj\,\mathbf{A}}{|\mathbf{A}|} = \frac{1}{|\mathbf{A}|} \begin{pmatrix} C_{11} & C_{21} & \cdots & C_{n1} \\ C_{21} & C_{22} & \cdots & C_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ C_{n1} & C_{2n} & \cdots & C_{nn} \end{pmatrix} = \mathbf{C}^{\mathrm{T}}$$

식은 간단하지만, 여인자를 구하는 재귀호출이 매우 많은 계산을 요구

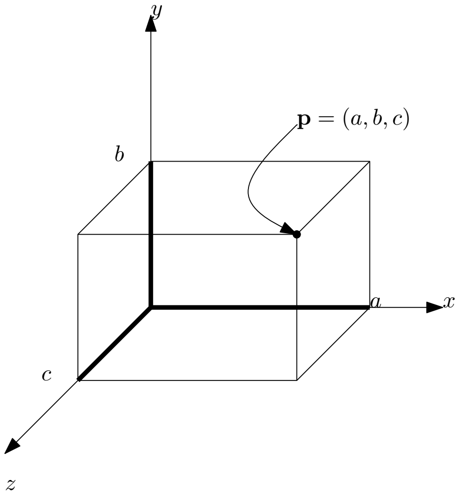
변환이란

수학적 의미에서 변환(transformation)

- 어떤 집합 S를 다른 어떤 집합 S로 대응시키는 함수
- 공간과 점, 그리고 벡터의 문제로 이해할 때, 변환이란 공간 상의 벡터나 점을 다른 벡터나 점으로 바꾸는 연산
- 변환 행렬
  - 어떤 벡터 $\mathbf{a}$가 $\mathbb{R}^n$에 속한다고 할 때, 이 벡터에 행렬 $\mathbf{A} \in \mathbb{R}^{n\times n}$을 곱하면 $\mathbf{a}$와 같은 차원의 벡터 $\mathbf{b}$를 얻는다.
  - $\mathbf{b} = \mathbf{Aa}$ $(\mathbf{a}, \mathbf{b} \in \mathbb{R}^n, \mathbf{A} \in \mathbb{R}^{n\times n})$.
  - 어떤 벡터를 동일한 차원의 다른 벡터로 옮기는 행렬 $\mathbf{A} \in \mathbb{R}^{n\times n}$을 변환행렬(transform matrix)라고 한다.

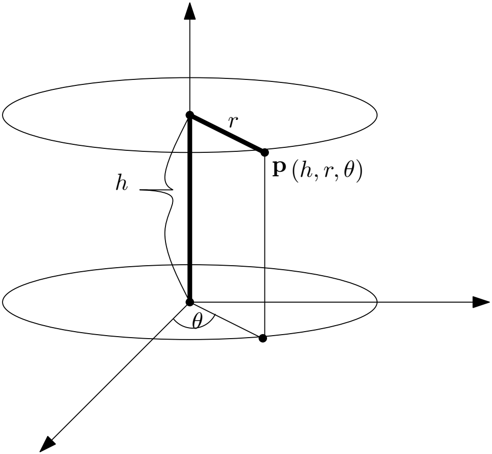
좌표계 - 직교 좌표계

- 일반적으로 가장 익숙한 좌표계
- 데카르트 좌표계(Cartesian coordinate system)



좌표계 - 원기둥 좌표계

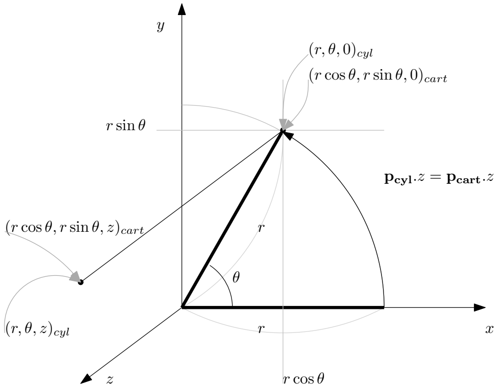
- $\mathbf{p}$는 이러한 높이 h와 반지름 r을 가진 원기둥의 윗쪽 원주에 놓임.
- 원주에서 특정한 위치는 각도 θ로 표현
- 원기둥 좌표: (r, θ, h)



원기둥 좌표를 직교 좌표로 옮기기

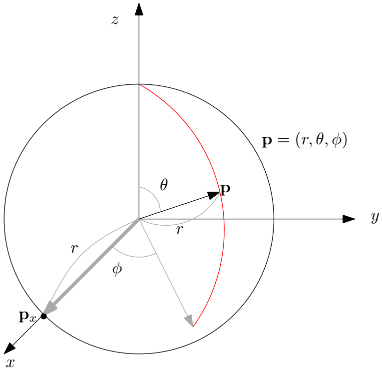
원기둥 좌표계의 좌표를 $\mathbf{p}_{cyl}$, 직교 좌표계의 좌표를 $\mathbf{p}_{cart}$ 으로 표현하면

$$(r, \theta, h)_{cyl} = (r\cos\theta, r\sin\theta, h)_{cart}$$



좌표계 - 구면 좌표계

- $\mathbf{p}$를 지나며 중심이 원점인 구면의 반지름을 r
- 반지름 r인 점 가운데 x 축 위에 있는 점을 $\mathbf{p}_x$
- $\mathbf{p}_x$을 xy 평면 위에서 $\mathbf{p}$와 같은 경도선에 놓는 각도가 ϕ
- 이를 들어 올려 점 $\mathbf{p}$를 지나도록 하는 데에 필요한 각도를 θ
- 구면 좌표 (r, θ, ϕ)



구면 좌표와 직교 좌표

구면 좌표는 일반적으로 다음과 같은 제한을 갖는다.

$$r \geq 0$$

$$0 \leq \theta \leq \pi$$

$$0 \leq \phi \leq 2\pi$$

직교 좌표계의 좌표 (x, y, z)를 구면 좌표계로 옮기기

$$\begin{aligned} r &= \sqrt{x^2 + y^2 + z^2} \\ \theta &= \arccos\left(\frac{z}{\sqrt{x^2+y^2+z^2}}\right) \\ \phi &= \arctan\left(\frac{y}{x}\right) \end{aligned}$$

구면 좌표계의 좌표를 직교 좌표로 옮기기

$$\begin{aligned} x &= r \sin\theta \cos\phi \\ y &= r \sin\theta \sin\phi \\ z &= r \cos\theta \end{aligned}$$

무슨 좌표계를 사용할 것인가

- 공간에 존재하는 점을 다룰 때에는 어떠한 좌표계를 사용해도 무방
- 컴퓨터 그래픽스 분야에서 가장 많이 사용되는 좌표계는 직교 좌표계
- 우리는 직교 좌표계에서 변환에 대해 다룰 예정
- 직교 좌표계를 기본적인 좌표계로 삼고 변환과 관련된 행렬 연산을 살필 것

어파인(affine) 변환

게임을 구현하기 위한 3차원 그래픽스에서 흔히 사용되는 변환

- 이동변환(translation): 주어진 변위 벡터만큼 좌표를 동일하게 옮김
- 회전변환(rotation): 2차원은 기준점, 3차원은 기준축을 중심으로 돌림
- 크기변경(scaling): 각 축 방향으로 주어진 비율에 따라 좌표 값이 커지거나 줄어든다.

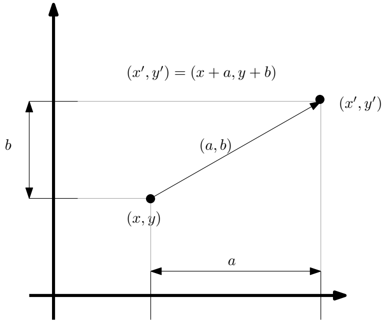
이러한 변환은 어파인 변환(affine transformation)의 일종

- 서로 연결되어 있음을 의미하는 라틴어 'affinis'에서 유래
- 직선 위의 점들을 직선을 유지한 상태로 변환하는 변환
- 직선 위에서의 점들 사이의 거리 비가 변환된 직선 위에서 그대로 유지
- 직선은 직선으로, 평행선은 평행선으로 유지
- 실시간 컴퓨터 그래픽스에서는 여러 가지 효율성의 이유로 어파인 변환을 사용

이동 변환(translation)

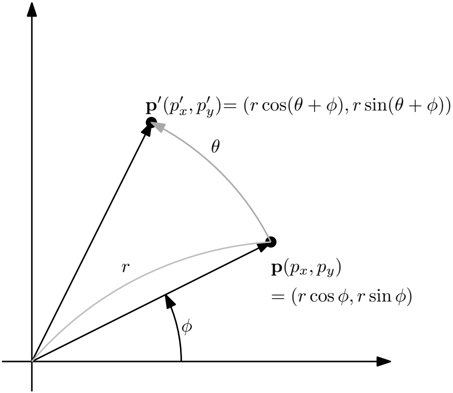
- 2차원: 좌표 (x, y)를 x 축 방향으로 a, y 축 방향으로 b 만큼 옮기기
- $(x', y') = (x, y) + (a, b) = (x + a, y + a)$

- 모든 차원에 대해 어떤 벡터 $\mathbf{a}$를 변위 벡터 $\mathbf{d}$를 이용하여 $\mathbf{x}'$로 옮기는 이동 변환을 다음과 같이 벡터 더하기로 정의할 수 있음
  - $\mathbf{a} \in \mathbb{R}^n, \mathbf{d} \in \mathbb{R}^n$
  - $\mathbf{x}' = \mathbf{a} + \mathbf{d} \quad \mathbf{x}' \in \mathbb{R}^n$

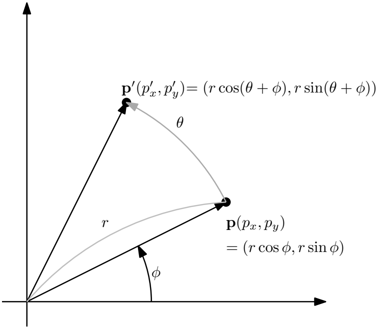


2차원 회전 변환(rotation) - 문제

- 2차원 회전의 중심: 피벗(pivot)
- 기본적인 회전: 피벗이 원점인 경우
  - $\mathbf{p}$를 원점을 중심으로 θ 만큼 회전하여 놓이는 지점 $\mathbf{p}'$를 구하는 문제
  - 원래 좌표 (p_x, p_y)를 θ 만큼 회전하여 얻는 (p'_x, p'_y)를 얻는 문제



2차원 회전 변환(rotation) - 좌표값에 대한 이해



- 원점에서 (p_x, p_y)로 선분: 선분 길이 r과 x축과 이루는 각도 ϕ
- $(p_x, p_y) = (r\cos\phi, r\sin\phi)$
- 이 좌표를 θ만큼 회전하여 얻는 (p'_x, p'_y)
  - $(p'_x, p'_y) = (r\cos(\theta+\phi), r\sin(\theta+\phi))$

2차원 회전 변환(rotation) - 회전결과

- ϕ를 계산하지 않고 답을 얻어야 함
- 참조할 공식
  - $\cos(a+b) = \cos a \cos b - \sin a \sin b$
  - $\sin(a+b) = \sin a \cos b + \cos a \sin b$
- 회전하여 얻는 좌표는 다음과 같이 표현
  - $p'_x = (r\cos\phi)\cos\theta - (r\sin\phi)\sin\theta$
  - $p'_y = (r\cos\phi)\sin\theta + (r\sin\phi)\cos\theta$
- 원래의 좌표 (p_x, p_y)를 이용하여 표현
  - $p'_x = p_x\cos\theta - p_y\sin\theta$
  - $p'_y = p_x\sin\theta + p_y\cos\theta$

2차원 회전 변환(rotation) - 행렬표현

이러한 변환은 다음과 같은 행렬과 벡터의 곱으로 표현할 수 있다.

$$\begin{bmatrix} p'_x \\ p'_y \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} p_x \\ p_y \end{bmatrix}$$

2차원 공간에서 어떤 점 $\mathbf{p}$를 원점 기준으로 θ만큼 회전시켜 $\mathbf{p}'$를 얻는 변환은 회전변환 행렬 $\mathbf{R}(\theta)$을 이용하여 $\mathbf{p}' = \mathbf{R}(\theta)\mathbf{p}$로 표현할 수 있다.

$$\mathbf{R}(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

3차원 회전 - z 축 회전

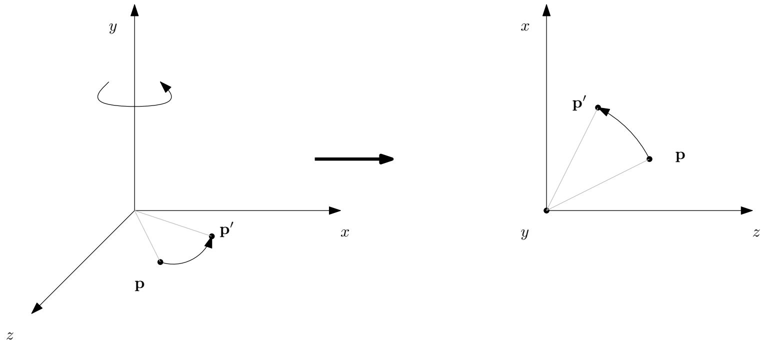
- 2차원 회전을 그대로 3차원에 적용
  - 3차원 좌표 $\mathbf{p} = (p_x, p_y, p_z)$을 z 기준으로 회전
- 이 변환은 2차원 변환에 z 성분만 추가
  - z 축 성분은 그대로 유지된다. $(p'_z = p_z)$
  - p_x, p_y 의 값은 2차원 회전과 동일하게 변환된다.

$$\begin{aligned}
p'_x &= \cos\theta \cdot p_x - \sin\theta \cdot p_y + 0 \cdot p_z \\
p'_y &= \sin\theta \cdot p_x + \cos\theta \cdot p_y + 0 \cdot p_z \\
p'_z &= 0 \cdot p_x + 0 \cdot p_y + 1 \cdot p_z
\end{aligned}$$

이것은 다음과 같은 행렬 표현으로 다시 쓸 수 있다.

$$\begin{bmatrix} p'_x \\ p'_y \\ p'_z \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix}$$

3차원 회전 - y 축 회전 (1/2)



- z 축은 2차원 회전의 x축에 대응
- x 축은 2차원 회전의 y축에 대응

$$
\begin{array}{rcrrr}
p'_z & = & \cos\theta \cdot p_z & -\sin\theta \cdot p_x & +0 \cdot p_y \\
p'_x & = & \sin\theta \cdot p_z & +\cos\theta \cdot p_x & +0 \cdot p_y \\
p'_y & = & 0 \cdot p_z & +0 \cdot p_x & +1 \cdot p_y
\end{array}
$$

3차원 회전 - y 축 회전 (1/2)

순서를 재배열하면 다음과 같은 식을 얻는다.

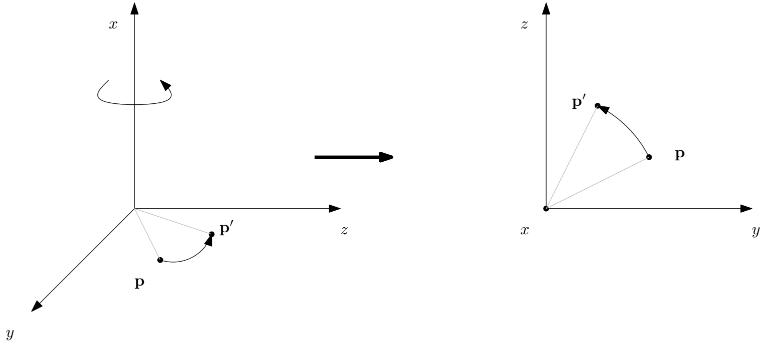
$$\begin{array}{rcrrr} p'_x & = & \cos\theta\cdot p_x & +0\cdot p_y & +\sin\theta\cdot p_z \\ p'_y & = & 0\cdot p_x & +1\cdot p_y & 0\cdot p_z \\ p'_z & = & -\sin\theta\cdot p_x & +0\cdot p_y & +\cos\theta\cdot p_z \end{array}$$

이것도 역시 행렬 표현으로 다시 쓸 수 있다.

$$\begin{bmatrix} p'_x \\ p'_y \\ p'_z \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix}$$

3차원 회전 - x 축 회전 (1/2)

- 2차원 회전에서 x, y의 역할에 y와 z 축이 각각 대응



$$
\begin{array}{rcrrr}
p'_y & = & \cos\theta\cdot p_y & -\sin\theta\cdot p_z & +0\cdot p_x \\
p'_z & = & \sin\theta\cdot p_y & +\cos\theta\cdot p_z & +0\cdot p_x \\
p'_x & = & 0\cdot p_y & +0\cdot p_z & +1\cdot p_x
\end{array}
$$

3차원 회전 - x 축 회전 (2/2)

행렬과 벡터의 곱으로 표현하면 다음과 같다.

$$\begin{bmatrix} p'_x \\ p'_y \\ p'_z \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix}$$

회전 행렬의 역행렬

- 회전행렬은 특별한 특징을 지님 (2차원 회전행렬을 보자)
  - 첫 열 벡터는 길이는 $\sqrt{\cos^2\theta + \sin^2\theta}$이므로 1
  - 두 번째 열의 길이 역시 $\sqrt{\sin^2\theta + \cos^2\theta}$로 1
  - 즉 두 벡터 모두 단위 벡터 (정규)
  - 이 두 벡터를 서로 내적하면 $\cos\theta(-\sin\theta) + \sin\theta\cos\theta$로 0
  - 두 벡터가 서로 수직 (직교)
- 모든 벡터는 단위 벡터이고 서로 직교 = 정규직교(orthonormal)
- 정규직교 행렬의 역행렬은 그 행렬의 전치(transpose)와 같음
- 3차원 회전행렬들도 정규직교임을 쉽게 확인 가능

$$\mathbf{R}_x{}^{-1}(\theta) = \mathbf{R}_x{}^{\mathrm{T}}(\theta)$$
$$\mathbf{R}_y{}^{-1}(\theta) = \mathbf{R}_y{}^{\mathrm{T}}(\theta)$$
$$\mathbf{R}_z{}^{-1}(\theta) = \mathbf{R}_z{}^{\mathrm{T}}(\theta)$$

임의의 축에 대한 회전

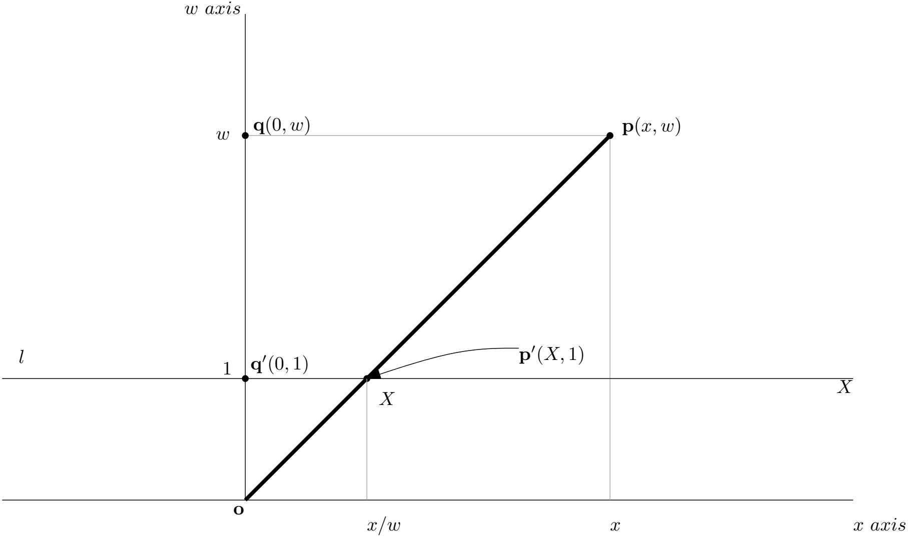
- 임의의 회전축을 기준으로 θ만큼 회전하는 변환은 다음과 같은 절차를 따라 구현할 수 있다.
  1. 회전축이 원점을 지나도록 이동변환 $\mathbf{T}$를 적용한다.
  2. 원점을 지나는 회전축이 xz 평면에 놓이도록 z축 회전 $\mathbf{R}_1$ 적용
  3. 이 회전축이 z축과 동일한 방향이 되도록 y축 회전 $\mathbf{R}_2$를 적용한다.
  4. z축을 기준으로 θ만큼 회전하도록 $\mathbf{R}_z(\theta)$를 적용한다.
  5. ${\mathbf{R}_2}^{-1}$, 즉 ${\mathbf{R}_2}^{\mathrm{T}}$를 적용한다.
  6. ${\mathbf{R}_1}^{-1}$, 즉 ${\mathbf{R}_1}^{\mathrm{T}}$를 적용한다.
  7. $\mathbf{T}^{-1}$, 즉 $-\mathbf{T}$를 적용한다.
- 벡터 더하기(이동변환)와 행렬 곱하기(회전)가 혼재
- 동차좌표계(homogeneous coordinate)을 이용
  - 이동 변환과 회전 변환 모두 4×4 행렬
  - 위의 절차를 모두 누적한 하나의 행렬로 표현 가능
- $\mathbf{R}_{pivot}(\theta) = -\mathbf{T}{\mathbf{R}_1}^{\mathrm{T}}{\mathbf{R}_2}^{\mathrm{T}}\mathbf{R}_z(\theta)\mathbf{R}_2\mathbf{R}_1\mathbf{T}$

동차좌표계(homogeneous coordinate)

- 사영기하학(射影幾何學, projective geometry)에서 사용되는 좌표계
- n 차원의 사영공간을 $n+1$의 좌표로 표현
- 그래픽 API나 라이브러리들은 동차좌표계를 표준적인 좌표계로 채용
  - 3차원 그래픽스에서 정의된 3차원 가상 공간 객체의 2차원 투영 이미지를 얻는 일과 유관
  - 3차원 공간의 어파인(affine) 변환들을 모두 4×4의 행렬로 표현
- 3차원 공간의 좌표를 표현하는 벡터 $[x, y, z]^{\mathrm{T}}$는 동차좌표계에서 $[x, y, z, 1]^{\mathrm{T}}$로 표현할 수 있음
- 보다 일반적인 형태는 마지막 숫자를 1이 아닌 다른 값게 하는 것
- 2차원: $[x, y, w]^{\mathrm{T}}$, 3차원: $[x, y, z, w]^{\mathrm{T}}$

동차좌표계의 시각적 이해 (1/2)

동차좌표계 이해를 위해 Ammeraal이 사용한 그림



동차좌표계의 시각적 이해 (2/2)

- 두개의 축: x 축이고, 다른 하나는 w 축
  - 동차 좌표계에서 마지막 원소 w를 제외한 모든 성분은 이 x 축 값
  - 마지막 원소는 이 w 축 값
- x축 위에 있지 않는 점 $\mathbf{p}$는 중심 사영(central projection) $\mathbf{p}'$를 가짐
  - $w = 1$의 직선 l과 원점 $\mathbf{o}$에서 $\mathbf{p}$를 연결한 직선의 교차점
  - 이때 원점은 사영중심(center of projection)
- w 축과 x축을 모두 포함한 차원의 공간을 이보다 한 차원 낮은 x축 공간으로 떨어뜨리는 것
- 선분 $\overline{\mathbf{op}}$를 지나는 직선 위의 모든 점들이 이 $\mathbf{p}'$로 사영
- (x, w)에 해당하는 $\mathbf{p}'(X, 1)$ 구하기
  - 닮은 삼각형 $\mathbf{opq}$와 $\mathbf{op'q'}$
  - 등비 관계를 이용하여 구한다
  - $X = \frac{X}{1} = \frac{|\mathbf{p'q'}|}{|\mathbf{oq'}|}$
- 이 식은 다음과 같이 바뀐다.
- $X = \frac{X}{1} = \frac{|\mathbf{oq'}|}{|\mathbf{p'q'}|} = \frac{|\mathbf{oq}|}{|\mathbf{pq}|} = \frac{x}{w}$

동차좌표계와 데카르트 좌표계의 관계

w 좌표의 의미

사영기하에서 **op**를 지나는 직선 위의 모든 점들은 (x, w) 형태의 좌표로 표현할 수 있고, 이 모든 점들은 $w = 1$인 평면으로 중심사영을 수행했을 때, w 좌표는 무의미해지면서 (x/w)의 좌표로 바뀌게 된다. 즉, 3차원 공간의 좌표를 표현하기 위해 동차좌표계를 사용한다면 $[x, y, z, w]^{\mathrm{T}}$의 형태가 되며, 이것은 위의 그림에서 w 축을 포함한 공간이 된다. 이를 3차원 데카르트 좌표로 바꾸는 것은 중심사영이 이루어지는 $w = 1$ 평면으로 옮겨 놓는 것이고 이때의 좌표는 $[x/w, y/w, z/w]^{\mathrm{T}}$가 되는 것이다. 그리고 3차원 공간의 측면에서 보면, **op**를 지나는 직선 위의 모든 점들이 동일한 점으로 간주되는 것이다.

동차 좌표계 사용의 이점(利點)

- 3차원 좌표 $[x, y, z]^{\mathrm{T}}$ 를 동차좌표계 좌표로 바꾸는 간단한 방법은 $w = 1$ 평면에서의 좌표인 $[x, y, z, 1]^{\mathrm{T}}$
- 동차좌표계를 쓰면 좋은 점
  - 좌표와 벡터의 구분이 가능
  - $[x, y, z]^{\mathrm{T}}$ 가 3차원 좌표라면 이 좌표로 표현되는 지점은 3차원 공간내 유일
  - 벡터로 해석된다면 그것은 수 많은 동등 벡터를 표현하게 되며, 공간 내의 특별한 지점을 가리키지 않음
  - 이 둘은 분명히 다르지만 단순한 좌표 표현 방식으로는 구분이 불가능
- 동차좌표계에서 좌표와 벡터의 구분
  - 좌표는 $w \neq 0$ 인 $[x, y, z, w]^{\mathrm{T}}$
  - 벡터는 $w = 0$ 인 $[x, y, z, 0]^{\mathrm{T}}$
  - $[x, y, z, 0]^{\mathrm{T}}$ 는 위치를 가진 좌표 $[x, y, z]^{\mathrm{T}}$ 가 아니라 위치가 없는 벡터 $[x, y, z]^{\mathrm{T}}$
- 또다른 잇점
  - 이동변환와 회전변환을 모두 같은 차원의 행렬로 표현

동차 좌표계에서 이동 변환

동차좌표계에서의 이동

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} + \begin{bmatrix} d_x \\ d_y \\ d_z \\ 0 \end{bmatrix} = \begin{bmatrix} x + d_x \\ y + d_y \\ z + d_z \\ 1 \end{bmatrix}$$

행렬 표현

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & d_x \\ 0 & 1 & 0 & d_y \\ 0 & 0 & 1 & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x + d_x \\ y + d_y \\ z + d_z \\ 1 \end{bmatrix}$$

이동 변환 행렬의 역행렬

이제 이동변환을 행렬로 표현할 수 있게 되었다. 변위 벡터 $\mathbf{d}(d_x, d_y, d_z)$ 만큼의 이동을 수행하는 변환행렬을 $\mathbf{T_d}$ 라고 하면 이동 변환은 다음과 같이 표현할 수 있다.

$$\mathbf{p}' = \mathbf{T_d}\mathbf{p}, \quad \mathbf{T_d} \in \mathbb{R}^{4\times 4}$$

이동변환 행렬 $\mathbf{T_d}$ 의 역행렬은 어떻게 될까? 역행렬은 이 행렬이 일으킨 변환을 원래대로 되돌려 놓는 것이므로 $\mathbf{T_{-d}}$ 가 됨을 알 수 있다.

$$\begin{bmatrix} 1 & 0 & 0 & d_x \\ 0 & 1 & 0 & d_y \\ 0 & 0 & 1 & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & 0 & 0 & -d_x \\ 0 & 1 & 0 & -d_y \\ 0 & 0 & 1 & -d_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

동차좌표계에서의 회전 행렬

- 3차원 공간에서 정의되었던 회전 변환을 $\mathbf{R}_{33}$
- $\mathbb{R}^{4\times4}$ 에 속하는 동차좌표계 회전 행렬은 $\mathbf{R}_{44}$
- 원소가 모두 0인 3차원 열벡터를 $\mathbf{O}_3^{col}$, 행벡터를 $\mathbf{O}_3^{row}$

$$\mathbf{R}_{44} = \begin{bmatrix} \mathbf{R}_{33} & \mathbf{O}_3^{col} \\ \mathbf{O}_3^{row} & 1 \end{bmatrix}$$

$$\mathbf{R}_{44}^x = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{R}_{44}^y = \begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{R}_{44}^z = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

복합변환

좌표를 $\mathbf{R}_{44}$ 를 이용하여 회전하고, 이를 $\mathbf{T_d}$ 만큼 이동하는 변환

$$\mathbf{p}' = \mathbf{T_d}\mathbf{R}_{44}\mathbf{p}$$

두 변환을 모두 수행하는 하나의 행렬 구할 수 있음

$$\begin{aligned}\mathbf{p}' &= \begin{bmatrix} \mathbf{I}_{33} & \mathbf{d} \\ \mathbf{O}_3^{row} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_{33} & \mathbf{O}_3^{col} \\ \mathbf{O}_3^{row} & 1 \end{bmatrix} \mathbf{p} \\ &= \begin{bmatrix} \mathbf{R}_{33} & \mathbf{d} \\ \mathbf{O}_3^{row} & 1 \end{bmatrix} \mathbf{p}\end{aligned}$$

- 행렬의 역행렬은?

$$\begin{bmatrix} \mathbf{R}_{33}^{\mathrm{T}} & \mathbf{O}_3^{col} \\ \mathbf{O}_3^{row} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{I}_{33} & -\mathbf{d} \\ \mathbf{O}_3^{row} & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{33}^{\mathrm{T}} & -\mathbf{R}_{33}^{\mathrm{T}}\mathbf{d} \\ \mathbf{O}_3^{row} & 1 \end{bmatrix}$$

복합변환

좌표를 $\mathbf{R}_{44}$ 를 이용하여 회전하고, 이를 $\mathbf{T_d}$ 만큼 이동하는 변환

$$\mathbf{p}' = \mathbf{T_d}\mathbf{R}_{44}\mathbf{p}$$

두 변환을 모두 수행하는 하나의 행렬 구할 수 있음

$$\begin{aligned}\mathbf{p}' &= \begin{bmatrix} \mathbf{I}_{33} & \mathbf{d} \\ \mathbf{O}_3^{row} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_{33} & \mathbf{O}_3^{col} \\ \mathbf{O}_3^{row} & 1 \end{bmatrix} \mathbf{p} \\ &= \begin{bmatrix} \mathbf{R}_{33} & \mathbf{d} \\ \mathbf{O}_3^{row} & 1 \end{bmatrix} \mathbf{p}\end{aligned}$$

- 행렬의 역행렬은?

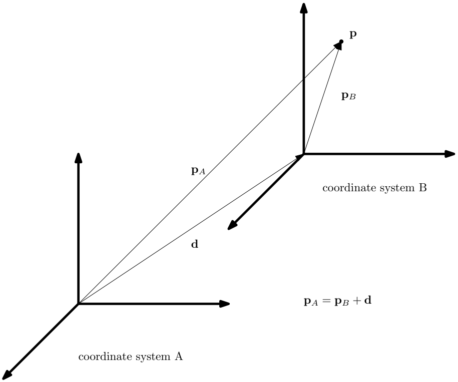
$$\begin{bmatrix} \mathbf{R}_{33}^{\mathrm{T}} & \mathbf{O}_3^{col} \\ \mathbf{O}_3^{row} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{I}_{33} & -\mathbf{d} \\ \mathbf{O}_3^{row} & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{33}^{\mathrm{T}} & -\mathbf{R}_{33}^{\mathrm{T}}\mathbf{d} \\ \mathbf{O}_3^{row} & 1 \end{bmatrix}$$

좌표계의 변환

- 어떤 점이 좌표계 A에 의해 $\mathbf{p}_A$라 표현된다고 가정
- 다른 좌표계 B의 입장에서 보면 다른 좌표 $\mathbf{p}_B$
- 이렇게 좌표계가 달라질 때 바뀐 좌표계에 따라 새로운 좌표를 계산하는 일은 그래픽스에서 매우 빈번히 나타나는 작업
    - 가상 공간 내의 모든 객체의 위치를 하나의 기준으로 정의하는 데에 필요한 전역좌표계(global coordinate system)과 개별 객체 내에 정의된 지역좌표계(local coordinate system)

좌표계의 이동

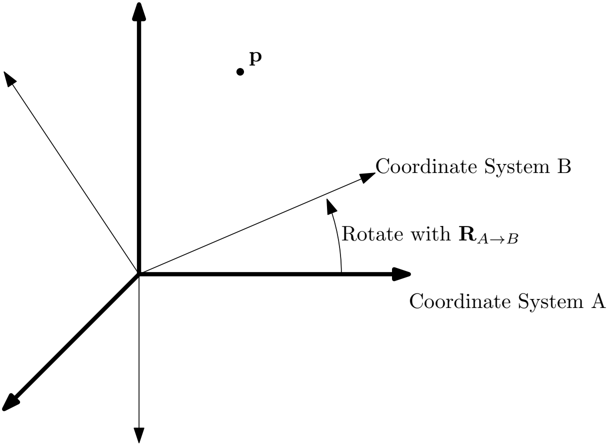
$\mathbf{p}_A = \mathbf{p}_B + \mathbf{d}$



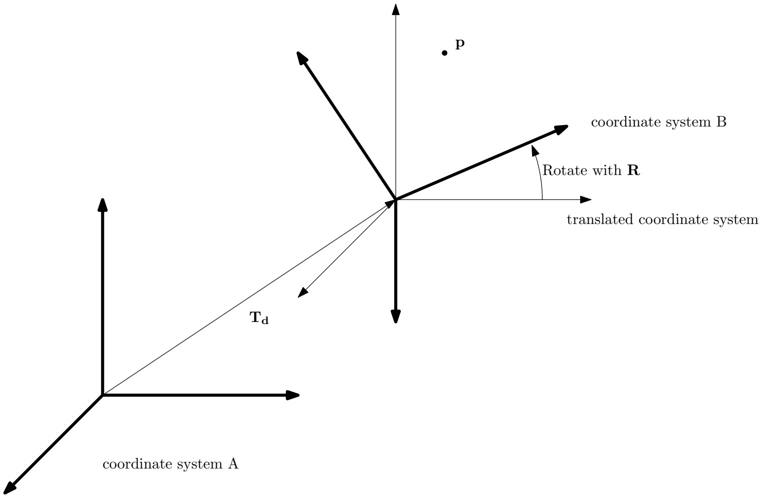
$$\mathbf{T_d} = \begin{bmatrix} 1 & 0 & 0 & d_x \\ 0 & 1 & 0 & d_y \\ 0 & 0 & 1 & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{p}_A = \mathbf{T_d}\mathbf{p}_B, \quad \mathbf{p}_B = \mathbf{T_d}^{-1}\mathbf{p}_A$$

좌표계의 회전

$$\begin{aligned} \mathbf{p}_A &= \mathbf{R}_{A\to B}\mathbf{p}_B \\ \mathbf{p}_B &= \mathbf{R}^{-1}_{A\to B}\mathbf{p}_A &= \mathbf{R}^{\mathrm{T}}_{A\to B}\mathbf{p}_A \end{aligned}$$



회전과 이동이 함께 이뤄진 좌표계 변환 (1/2)



회전과 이동이 함께 이뤄진 좌표계 변환 (1/2)

$$\mathbf{R} = \begin{bmatrix} \mathbf{R}_{33} & 0 \\ \mathbf{O}_3^{row} & 1 \end{bmatrix}, \quad \mathbf{T}_{\mathbf{d}} = \begin{bmatrix} \mathbf{I}_{33} & \mathbf{d} \\ \mathbf{O}_3^{row} & 1 \end{bmatrix}, \quad \mathbf{T}_{\mathbf{d}}\mathbf{R} = \begin{bmatrix} \mathbf{R}_{33} & \mathbf{d} \\ \mathbf{O}_3^{row} & 1 \end{bmatrix}$$

$$\begin{aligned} \mathbf{p}_A &= \begin{bmatrix} \mathbf{R}_{33} & \mathbf{d} \\ \mathbf{O}_3^{row} & 1 \end{bmatrix} \mathbf{p}_B \\ \mathbf{p}_B &= \begin{bmatrix} \mathbf{R}_{33}^{\mathrm{T}} & \mathbf{R}_{33}^{\mathrm{T}}\mathbf{d} \\ \mathbf{O}_3^{row} & 1 \end{bmatrix} \mathbf{p}_A \end{aligned}$$