1. 3D 그래픽 기초(3D Graphics Fundamentals)

(4) 행렬(Matrix)의 이해

① 행렬(Matrix)

행렬은 행(Row)과 열(Column)로 배열된 실수 값들의 나열(직사각형 표)이다(C 프로그래밍 언어의 배열과 유사하다). m개의 행과 n개의 열을 가진 행렬은 $(m \times n)$행렬이라고 하며 $(m \times n)$을 행렬의 차원이라 한다. 행렬은 $\boldsymbol{M}$과 같이 대문자를 사용하여 표기한다. 행렬 $\boldsymbol{M}$의 원소는 m_{ij} 또는 $m[i][j]$ 형태로 나타낸다. 이때 i는 행 번호이고 j는 열 번호를 나타내며 m_{ij}는 행렬 $\boldsymbol{M}$의 i-번째 행과 j-번째 열의 원소(Element)라고 한다. 예를 들어, m_{12}는 행렬 $\boldsymbol{M}$의 첫 번째 행과 두 번째 열의 원소를 나타낸다. 행렬 $\boldsymbol{M}$을 $[m_{ij}]$로 표기하기도 한다.

다음은 (4×4) 행렬의 예이다.

$$\boldsymbol{M} = \begin{bmatrix} 1 & 3 & 5 & 7 \\ 9 & 8 & 6 & 4 \\ 9 & 7 & 6 & 6 \\ 3 & 1 & 0 & 2 \end{bmatrix}$$

$(m \times n)$ 행렬 $\boldsymbol{M}$은 m개의 행을 가지고 있으므로 행렬 $\boldsymbol{M}$은 m개의 $(1 \times n)$ 행렬을 가지는 것과 같다. 이때 $(1 \times n)$ 행렬을 n-차원 행벡터라고 한다. 행렬 $\boldsymbol{M}$에서 i-번째 n-차원 행벡터를 $m_{i\,\bullet}$로 표기한다. $(m \times n)$ 행렬 $\boldsymbol{M}$은 n개의 열을 가지고 있으므로 행렬 $\boldsymbol{M}$은 n개의 $(m \times 1)$ 행렬을 가지는 것과 같다. 이때 $(m \times 1)$ 행렬을 m-차원 열벡터라고 한다. 행렬 $\boldsymbol{M}$에서 j-번째 m-차원 열벡터를 $m_{\bullet\, j}$로 표기한다.

다음은 $(n \times m)$ 행렬이다.

$$\boldsymbol{M} = \begin{bmatrix} m_{11} & m_{12} & \cdots & m_{1m} \\ m_{21} & m_{22} & \cdots & m_{2m} \\ \cdots & \cdots & \cdots & \cdots \\ m_{n1} & m_{n2} & \cdots & m_{nm} \end{bmatrix} = [m_{ij}]$$

다음은 행렬 $\boldsymbol{M}$의 i-번째 행벡터이다.

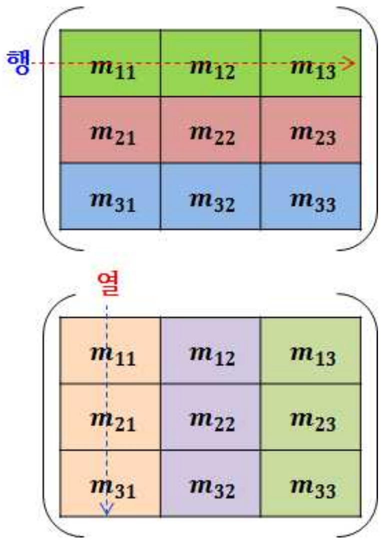
$$m_{i*} = [\, m_{i1}\; m_{i2}\; \cdots\; m_{im} \,]$$

다음은 행렬 $\boldsymbol{M}$의 j-번째 열벡터이다.

$$m_{*j} = \begin{bmatrix} m_{1j} \\ m_{2j} \\ \cdots \\ m_{nj} \end{bmatrix}$$

(1×3) 행렬을 3-차원 행벡터(Row Vector)로 취급할 수 있으며 (3×1) 행렬을 3-차원 열벡터(Column Vector)로 취급할 수 있다. 특별히 명시하지 않으면 3-차원 벡터는 (1×3) 행렬 또는 3-차원 행벡터를 의미하는 것으로 가정한다. 다음 그림은 (3×3) 행렬의 행과 열을 나타내고 있다.

ㅁ (3×3) 행렬의 행과 열



영행렬(Zero matrix)은 행렬의 모든 원소가 0인 행렬이며 **0**으로 표기한다. 행렬 $[m_{ij}]$가 영행렬이면 $\forall x \forall y (m_{ij} = 0)$이다.

$$\boldsymbol{M} = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & 0 \end{bmatrix}$$

n개의 행과 n개의 열을 가진 $(n \times n)$ 행렬을 n-차원 정사각형 행렬(Square matrix)이라고 한다. 다음 행렬 $\boldsymbol{M}$는 $(n \times n)$ 정사각형 행렬이다.

$$\boldsymbol{M} = \begin{bmatrix} m_{11} & m_{12} & \cdots & m_{1n} \\ m_{21} & m_{22} & \cdots & m_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ m_{n1} & m_{n2} & \cdots & m_{nn} \end{bmatrix}$$

$(n \times n)$ 정사각형 행렬 $[m_{ij}]$이 $\forall x \forall y \{(i \neq j) \Rightarrow (m_{ij} = 0)\}$를 만족하면 행렬 $[m_{ij}]$를 대각행렬(Diagonal matrix)이라고 한다. 다음 행렬 $\boldsymbol{M}$는 $(n \times n)$ 대각행렬이다.

$$\boldsymbol{M} = \begin{bmatrix} m_{11} & 0 & \cdots & 0 \\ 0 & m_{22} & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & m_{nn} \end{bmatrix}$$

$(n \times n)$ 정사각형 행렬 $[m_{ij}]$이 $\forall x \forall y \{((i=j) \Rightarrow (m_{ij}=1)) \wedge (i \neq j) \Rightarrow (m_{ij}=0)\}$를 만족하면 행렬 $[m_{ij}]$를 단위행렬(Identity matrix)이라고 하며 $\boldsymbol{I}$또는 $\boldsymbol{E}$로 표기한다. 즉, 단위행렬은 대각행렬에서 대각 원소가 모두 1인 행렬이다.

$$\boldsymbol{I} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}$$

다음 행렬은 (3×3) 단위행렬이다.

$$\boldsymbol{I} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$(m \times n)$ 행렬 $\boldsymbol{M}$에서 모든 행과 열을 서로 바꾼 $(n \times m)$ 행렬을 행렬 $\boldsymbol{M}$의 전치행렬(Transpose matrix)이라고 하며 $\boldsymbol{M}^T$로 표기한다.

$$\boldsymbol{M} = \begin{bmatrix} m_{11} & m_{12} & \cdots & m_{1m} \\ m_{21} & m_{22} & \cdots & m_{2m} \\ \cdots & \cdots & \cdots & \cdots \\ m_{n1} & m_{n2} & \cdots & m_{nm} \end{bmatrix} = [m_{ij}]$$

$$\boldsymbol{M}^T = \begin{bmatrix} m_{11} & m_{21} & \cdots & m_{n1} \\ m_{12} & m_{22} & \cdots & m_{n2} \\ \cdots & \cdots & \cdots & \cdots \\ m_{1m} & m_{2m} & \cdots & m_{nm} \end{bmatrix} = [m_{ij}]^T$$

$(n \times n)$ 정사각형 행렬 $[m_{ij}]$에서 $\forall i \forall j \{m_{ij} = m_{ji}\}$이면 행렬 $\boldsymbol{M}$을 대칭행렬(Symmetric matrix)이라고 한다. 행렬 $\boldsymbol{M}$이 대칭행렬이면 $\boldsymbol{M}^T = \boldsymbol{M}$이다.

$(n \times m)$ 행렬 $\boldsymbol{A}$와 $(n \times m)$ 행렬 $\boldsymbol{B}$의 모든 원소가 서로 같으면 행렬 $\boldsymbol{A}$와 행렬 $\boldsymbol{B}$는 같다고 하며 $\boldsymbol{A} = \boldsymbol{B}$로 표기한다. 즉, 행렬 $\boldsymbol{A}$와 행렬 $\boldsymbol{B}$의 차원이 같고 $\forall i \forall j \{a_{ij} = b_{ij}\}$이면 $\boldsymbol{A} = \boldsymbol{B}$이다.

② 행렬의 연산(Matrix Operation)

행렬에 대한 기본적인 연산은 행렬의 합(덧셈), 행렬의 차(뺄셈), 행렬의 스칼라 곱이다. 행렬의 합(덧셈), 행렬의 차(뺄셈), 행렬의 스칼라 곱 연산의 결과는 같은 차원의 행렬이 된다. 즉, 행렬에 대하여 벡터의 합(덧셈), 벡터의 차(뺄셈), 스칼라 곱 연산을 하면 새로운 행렬을 생성할 수 있다. 행렬의 합(덧셈)과 행렬의 차(뺄셈) 연산을 할 때 두 행렬의 차원이 같아야 하며 요소별 연산(Per-Component Operation)을 수행한다.

$(n \times m)$ 행렬 $\boldsymbol{A}$와 $\boldsymbol{B}$의 덧셈과 뺄셈은 다음을 만족하는 $(n \times m)$ 행렬 $\boldsymbol{C}$이다.

$$C = A \pm B$$

$$c_{ij} = a_{ij} \pm b_{ij}$$

$$A = \begin{bmatrix} a_{11}\, a_{12} \cdots a_{1m} \\ a_{21}\, a_{22} \cdots a_{2m} \\ \cdots \cdots \cdots \cdots \\ a_{n1}\, a_{n2} \cdots a_{nm} \end{bmatrix} = [a_{ij}] \qquad B = \begin{bmatrix} b_{11}\, b_{12} \cdots b_{1m} \\ b_{21}\, b_{22} \cdots b_{2m} \\ \cdots \cdots \cdots \cdots \\ b_{n1}\, b_{n2} \cdots b_{nm} \end{bmatrix} = [b_{ij}]$$

$$C = A + B = \begin{bmatrix} a_{11}+b_{11}\; a_{12}+b_{12} \;\cdots\; a_{1m}+b_{1m} \\ a_{21}+b_{21}\; a_{22}+b_{22} \;\cdots\; a_{2m}+b_{2m} \\ \cdots \qquad \cdots \qquad \cdots \qquad \cdots \\ a_{n1}+b_{n1}\; a_{n2}+b_{n2} \cdots a_{nm}+b_{nm} \end{bmatrix} \qquad [c_{ij}] = [a_{ij} + b_{ij}]$$

$$C = A - B = \begin{bmatrix} a_{11}-b_{11}\; a_{12}-b_{12} \;\cdots\; a_{1m}-b_{1m} \\ a_{21}-b_{21}\; a_{22}-b_{22} \;\cdots\; a_{2m}-b_{2m} \\ \cdots \qquad \cdots \qquad \cdots \qquad \cdots \\ a_{n1}-b_{n1}\; a_{n2}-b_{n2} \cdots a_{nm}-b_{nm} \end{bmatrix} \qquad [c_{ij}] = [a_{ij} - b_{ij}]$$

$(n \times m)$ 행렬 A와 실수(스칼라) k의 곱 kA는 다음을 만족하는 $(n \times m)$ 행렬 C이다.

$$C = kA = Ak$$

$$c_{ij} = k\, a_{ij}$$

$$C = kA = k\begin{bmatrix} a_{11}\, a_{12} \cdots a_{1m} \\ a_{21}\, a_{22} \cdots a_{2m} \\ \cdots \cdots \cdots \cdots \\ a_{n1}\, a_{n2} \cdots a_{nm} \end{bmatrix} = \begin{bmatrix} ka_{11}\, ka_{12} \cdots ka_{1m} \\ ka_{21}\, ka_{22} \cdots ka_{2m} \\ \cdots \cdots \cdots \cdots \\ ka_{n1}\, ka_{n2} \cdots ka_{nm} \end{bmatrix} \qquad [c_{ij}] = [ka_{ij}]$$

③ 행렬의 곱셈(Multiplication)

행렬 A와 B가 주어질 때 행렬 A의 열의 개수와 행렬 B의 행의 개수가 같을 때만 행렬의 곱을 할 수 있다(Inner Dimension Rule). $(m \times r)$ 행렬 A와 $(r \times n)$ 행렬 B가 주어질 때, 행렬의 곱의 결과는, $(1 \le i \le m)$이고 $(1 \le j \le n)$일 때, 다음 조건을 만족하는 $(m \times n)$ 행렬 C이다.

$$C = AB$$

$$c_{ij} = a_{i*} \bullet b_{*j}$$

즉, $(m \times r)$ 행렬 A와 $(r \times n)$ 행렬 B의 곱의 결과 행렬 C는 c_{ij}가 i-번째 행벡터와 j-번째 열벡터의 내적과 같다.

$$C = AB = [c_{ij}]$$

$$c_{ij} = a_{i1}b_{1j} + a_{i2}b_{2j} + \cdots + a_{ir}b_{rj} = \sum_{k=1}^{r} a_{ik}b_{kj}$$

$$AB = \begin{bmatrix} a_{11}\, a_{12} \cdots a_{1r} \\ a_{21}\, a_{22} \cdots a_{2r} \\ \cdots \cdots \cdots \cdots \\ a_{m1}\, a_{m2} \cdots a_{mr} \end{bmatrix} \begin{bmatrix} b_{11}\, b_{12} \cdots b_{1n} \\ b_{21}\, b_{22} \cdots b_{2n} \\ \cdots \cdots \cdots \cdots \\ b_{r1}\, b_{r2} \cdots b_{rn} \end{bmatrix} = \begin{bmatrix} c_{11}\, c_{12} \cdots c_{1n} \\ c_{21}\, c_{22} \cdots c_{2n} \\ \cdots \cdots \cdots \cdots \\ c_{m1}\, c_{m2} \cdots c_{mn} \end{bmatrix}$$

다음 그림은 (3×3) 행렬 A와 (3×3) 행렬 B의 행렬의 곱(AB)을 보여준다. 행렬의 곱의 결과는 (3×3) 행렬이다.

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} * \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{pmatrix}$$

$$= \begin{pmatrix} a_{11}b_{11}+a_{12}b_{21}+a_{13}b_{31} & a_{11}b_{12}+a_{12}b_{22}+a_{13}b_{32} & a_{11}b_{13}+a_{12}b_{23}+a_{13}b_{33} \\ a_{21}b_{11}+a_{22}b_{21}+a_{23}b_{31} & a_{21}b_{12}+a_{22}b_{22}+a_{23}b_{32} & a_{21}b_{13}+a_{22}b_{23}+a_{23}b_{33} \\ a_{31}b_{11}+a_{32}b_{21}+a_{33}b_{31} & a_{31}b_{12}+a_{32}b_{22}+a_{33}b_{32} & a_{31}b_{13}+a_{32}b_{23}+a_{33}b_{33} \end{pmatrix}$$

다음은 (4×4) 행렬을 표현하는 구조체이다.

```
struct Matrix
{
    float _11; float _12; float _13; float _14;
    float _21; float _22; float _23; float _24;
    float _31; float _32; float _33; float _34;
    float _41; float _42; float _43; float _44;
};
```

다음은 두 개의 (4×4) 행렬에 대한 행렬의 곱을 계산하는 함수이다.

```
Matrix MatrixMultiply(Matrix a, Matrix b)
{
    Matrix r;
    r._11 = a._11*b._11 + a._12*b._21 + a._13*b._31 + a._14*b._41;
    r._12 = a._11*b._12 + a._12*b._22 + a._13*b._32 + a._14*b._42;
    r._13 = a._11*b._13 + a._12*b._23 + a._13*b._33 + a._14*b._43;
    r._14 = a._11*b._14 + a._12*b._24 + a._13*b._34 + a._14*b._44;
    r._21 = a._21*b._11 + a._22*b._21 + a._23*b._31 + a._24*b._41;
    r._22 = a._21*b._12 + a._22*b._22 + a._23*b._32 + a._24*b._42;
    r._23 = a._21*b._13 + a._22*b._23 + a._23*b._33 + a._24*b._43;
    r._24 = a._21*b._14 + a._22*b._24 + a._23*b._34 + a._24*b._44;
    r._31 = a._31*b._11 + a._32*b._21 + a._33*b._31 + a._34*b._41;
    r._32 = a._31*b._12 + a._32*b._22 + a._33*b._32 + a._34*b._42;
    r._33 = a._31*b._13 + a._32*b._23 + a._33*b._33 + a._34*b._43;
    r._34 = a._31*b._14 + a._32*b._24 + a._33*b._34 + a._34*b._44;
    r._41 = a._41*b._11 + a._42*b._21 + a._43*b._31 + a._44*b._41;
    r._42 = a._41*b._12 + a._42*b._22 + a._43*b._32 + a._44*b._42;
    r._43 = a._41*b._13 + a._42*b._23 + a._43*b._33 + a._44*b._43;
    r._44 = a._41*b._14 + a._42*b._24 + a._43*b._34 + a._44*b._44;

    return(r);
```

}

행렬의 곱은 교환법칙이 성립하지 않음에 유의하라. 즉, 행렬 A와 B의 곱과 행렬 B와 A의 곱이 항상 같지 않다.

- $AB \neq BA$

행렬의 곱은 결합법칙이 성립한다.

- $A(BC) = (AB)C$

행렬의 곱에 대한 항등원은 단위행렬이다. 즉, $(m \times n)$ 행렬 A와 $(n \times n)$ 단위행렬 I에 대하여 다음이 성립한다.

- $AI = IA = A$

$(n \times n)$ 행렬 A, B, C와 실수 k, p, q에 대하여 다음의 행렬의 연산의 성질이 성립한다.

- $A + B = B + A$
- $A + (B + C) = (A + B) + C$
- $A + 0 = 0 + A = A$
- $(-1)A = -A$
- $A + (-1)A = (-1)A + A = A - A = -A + A = 0$
- $k(A + B) = kA + kB$
- $(p + q)A = pA + qA$
- $(pq)A = p(qA)$
- $A(B + C) = AB + AC$
- $(B + C)A = BA + CA$
- $k(AB) = k(A)B = A(kB)$
- $(A^T)^T = A$
- $(kA)^T = kA^T$
- $(A + B)^T = A^T + B^T$
- $(AB)^T = B^T A^T$

$(1 \times n)$ 벡터 a, b에 대하여 다음이 성립한다. 즉, 벡터의 내적은 $(1 \times n)$ 행렬과 $(n \times 1)$ 행렬의 곱으로 표현할 수 있다.

- $a \cdot b = ab^T$

정사각형 행렬 A에 대하여 $AB = BA = I$를 만족하는 행렬 B를 행렬 A의 역행렬(Inverse matrix)이라고 하며 A^{-1}로 표기한다. 역행렬이 존재하는 행렬을 가역 행렬

(Invertible matrix 또는 Nonsingular matrix)이라고 하고 역행렬이 존재하지 않는 행렬을 특이 행렬(Singular matrix)이라고 한다.

- $AA^{-1} = A^{-1}A = I$

역행렬에 대하여 다음의 성질이 성립한다.

- $(A^{-1})^{-1} = A$
- $(AB)^{-1} = B^{-1}A^{-1}$

직교행렬(Orthogonal matrix)은 단위 벡터인 행벡터들이 서로 직교하는 행렬이다. 직교행렬 A의 역행렬은 전치행렬이다.

- $A^{-1} = A^T$

다음은 (3×3) 직교행렬의 예이다.

$$R(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$R^{-1}(\theta) = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} = R^T(\theta)$$

$$R(-\theta) = R^{-1}(\theta)$$

④ 선형 변환(Linear Transformations)

변환 T가 m-차원 벡터 공간 V에서 n-차원 벡터 공간 W로의 선형 변환일 때, 벡터 공간 V의 벡터 x에 대하여 다음이 성립한다. 이것은 변환 T가 벡터 공간 V의 벡터 x를 벡터 공간 W로의 벡터로 변환하는 것의 의미한다.

$\{v_1, v_2, \cdots, v_m\}$가 벡터 공간 V의 기저 벡터들의 집합이면, 벡터 공간 V에서의 임의의 벡터 x는 다음과 같이 기저 벡터들의 선형결합으로 표현할 수 있다.

$$x = x_1 v_1 + x_2 v_2 + \cdots + x_m v_m$$

$$T : V \rightarrow W$$

$$T(x) = T(x_1 v_1 + x_2 v_2 + \cdots + x_m v_m) = x_1 T(v_1) + x_2 T(v_2) + \cdots + x_m T(v_m)$$

$\{w_1, w_2, \cdots, w_n\}$가 벡터 공간 W의 기저 벡터들의 집합이면, 벡터 공간 W에서의 임의의 벡터 b는 다음과 같이 기저 벡터들의 선형결합으로 표현할 수 있다.

$$b = b_1 w_1 + b_2 w_2 + \cdots + b_n w_n$$

$T(v_j)$는 벡터 공간 W에서의 벡터이므로, $T(v_j)$는 벡터 공간 W의 기저 벡터들과 n개

의 실수 a_{ij}들의 곱들의 합으로 표현할 수 있다.

$$T(\boldsymbol{v}_j) = a_{1j}\boldsymbol{w}_1 + a_{2j}\boldsymbol{w}_2 + \cdots + a_{nj}\boldsymbol{w}_n = (a_{1j}, a_{2j}, \cdots, a_{nj})$$

위의 식에서 $(a_{1j}, a_{2j}, \cdots, a_{nj})$는 벡터 공간 $\boldsymbol{W}$의 벡터이고, 벡터 공간 $\boldsymbol{V}$에서의 기저 벡터 $\boldsymbol{v}_j$가 변환 $\boldsymbol{T}$에 의하여 변환된 결과를 나타낸다.

이제 다음은 벡터 공간 $\boldsymbol{V}$의 벡터 $\boldsymbol{x}$는 벡터 공간 $\boldsymbol{W}$로의 벡터 $\boldsymbol{b}$로 변환되는 것을 표현한다.

$$\boldsymbol{T}(\boldsymbol{x}) = \boldsymbol{T}(x_1\boldsymbol{v}_1 + x_2\boldsymbol{v}_2 + \cdots + x_m\boldsymbol{v}_m) = x_1\boldsymbol{T}(\boldsymbol{v}_1) + x_2\boldsymbol{T}(\boldsymbol{v}_2) + \cdots + x_m\boldsymbol{T}(\boldsymbol{v}_m)$$

$$\boldsymbol{T}(\boldsymbol{x}) = x_1(a_{11}, a_{21}, \cdots, a_{n1}) + x_2(a_{12}, a_{22}, \cdots, a_{n2}) + \cdots + x_m(a_{1m}, a_{2m}, \cdots, a_{nm}) = (b_1, b_2, \cdots, b_n)$$

$$\boldsymbol{b} = (b_1, b_2, \cdots, b_n)$$

$$\boldsymbol{T}(\boldsymbol{x}) = \boldsymbol{b}$$

위의 식에서 다음이 성립한다.

$$b_1 = a_{11}x_1 + a_{12}x_2 + \cdots + a_{1m}x_m$$

$$b_2 = a_{21}x_1 + a_{22}x_2 + \cdots + a_{2m}x_m$$

$$b_i = a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{im}x_m$$

$$b_n = a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nm}x_m$$

위의 식을 행렬을 사용하여 표현하면 다음과 같다.

$$(b_1, b_2, \cdots, b_{n-1}, b_n) = (x_1, x_2, \cdots, x_{m-1}, x_m)\begin{bmatrix} a_{11} & a_{21} & \cdots & a_{n1} \\ a_{12} & a_{22} & \cdots & a_{n2} \\ \cdots & \cdots & \cdots & \cdots \\ a_{1m} & a_{2m} & \cdots & a_{nm} \end{bmatrix}$$

$$\boldsymbol{A} = \begin{bmatrix} a_{11} & a_{21} & \cdots & a_{n1} \\ a_{12} & a_{22} & \cdots & a_{n2} \\ \cdots & \cdots & \cdots & \cdots \\ a_{1m} & a_{2m} & \cdots & a_{nm} \end{bmatrix} \qquad \boldsymbol{b} = (b_1, b_2, \cdots, b_n)$$

$$\boldsymbol{b} = \boldsymbol{x}\boldsymbol{A}$$

위의 식은 벡터 공간 $\boldsymbol{V}$의 m-차원 벡터 $\boldsymbol{x}$를 벡터 공간 $\boldsymbol{W}$의 n-차원 벡터 $\boldsymbol{b}$로의 변환을 $(n \times m)$ 행렬 $\boldsymbol{A}$로 표현할 수 있음을 보여준다. 그리고 행렬 $\boldsymbol{A}$의 j-번째 행벡터는 벡터 공간 $\boldsymbol{V}$의 기저 벡터 $\boldsymbol{v}_j$가 변환 $\boldsymbol{T}$에 의하여 변환된 결과임을 보여준다. 즉, 어떤 선형 변환을 행렬로 표현하기 위하여 벡터 공간 $\boldsymbol{V}$의 기저 벡터 $\boldsymbol{v}_j$를 변환 $\boldsymbol{T}$로 변환한 결과를 행렬의 j-번째 행벡터로 만들면 된다는 것이다. 행렬 $\boldsymbol{A}$를 선형 변환 행렬이라고 한다.

주의 할 것은 벡터를 행벡터가 아닌 열벡터로 표현하면 $\boldsymbol{b} = \boldsymbol{x}\boldsymbol{A}$는 $\boldsymbol{b}^T = (\boldsymbol{x}\boldsymbol{A})^T = \boldsymbol{A}^T\boldsymbol{x}^T$로 표현된다는 것이다. 이때 변환 행렬은 위에서 표현한 행렬과 행과 열이 바뀐 것이다.

즉, 어떤 선형 변환을 행렬로 표현하기 위하여 벡터 공간 $\boldsymbol{V}$의 기저 벡터 $\boldsymbol{v}_j$를 변환 $\boldsymbol{T}$로 변환한 결과를 행렬의 j-번째 열벡터로 만들면 된다. 일반적으로 수학에서는 n-차원 벡터를 $(n \times 1)$ 열벡터로 표현하지만 이 과정에서는 n-차원 벡터를 $(1 \times n)$ 행벡터로 표현한다.

$$\begin{bmatrix} b_1 \\ b_2 \\ \cdots \\ b_{n-1} \\ b_n \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \cdots & \cdots & \cdots & \cdots \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \cdots \\ x_{m-1} \\ x_m \end{bmatrix}$$

$$\boldsymbol{B} = \boldsymbol{A}^T = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \cdots & \cdots & \cdots & \cdots \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{bmatrix}$$

$$\boldsymbol{b} = \begin{bmatrix} b_1 \\ b_2 \\ \cdots \\ b_{n-1} \\ b_n \end{bmatrix} \quad \boldsymbol{B} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \cdots & \cdots & \cdots & \cdots \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{bmatrix} \quad \boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ \cdots \\ x_{m-1} \\ x_m \end{bmatrix}$$

$$\boldsymbol{b} = \boldsymbol{B}\boldsymbol{x}$$

- 선형 변환의 예

  대표적인 선형 변환의 예는 크기(Scaling) 변환, 회전(Rotation) 변환, 반사(Reflection) 변환 등이다.

  - 크기(Scaling) 변환

    다음은 3-차원 좌표계에서 x-축으로 S_x배, y-축으로 S_y배, z-축으로 S_z배로 크기를 바꾸는 크기 변환 행렬이다. 크기 변환(확대 또는 축소)은 원점을 중심으로 변환을 하며 길이와 각도를 보존한다. 길이와 각도를 보존하지 못하는 변환은 변형(Deformation)이라고 한다.

$$\boldsymbol{S} = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & s_z \end{bmatrix}$$

$$(x, y, z)\boldsymbol{S} = (S_x x, S_y y, S_z z)$$

$$\boldsymbol{S}^{-1} = \begin{bmatrix} \frac{1}{S_x} & 0 & 0 \\ 0 & \frac{1}{S_y} & 0 \\ 0 & 0 & \frac{1}{S_z} \end{bmatrix}$$

  - 회전(Rotation) 변환

    z-축을 중심으로 θ만큼 회전을 하는 변환 행렬 $\boldsymbol{R}_z(\theta)$은 다음과 같다.

$$\boldsymbol{R}_z(\theta) = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

이것은 3-차원 좌표계의 x-축$(1,0,0)$, y-축$(0,1,0)$, z-축$(0,0,1)$을 z-축을 중심으로 θ만큼 회전한 결과를 행벡터로 구성한 것과 같다. x-축$(1,0,0)$을 z-축을 중심으로 θ만큼 회전한 결과는 다음과 같다.

$$x_2 = 1 \times \cos(\theta) - 0 \times \sin(\theta) = \cos(\theta)$$

$$y_2 = 1 \times \sin(\theta) + 0 \times \cos(\theta) = \sin(\theta)$$

$$z_2 = 0$$

y-축$(0,1,0)$을 z-축을 중심으로 θ만큼 회전한 결과는 다음과 같다.

$$x_2 = 0 \times \cos(\theta) - 1 \times \sin(\theta) = -\sin(\theta)$$

$$y_2 = 0 \times \sin(\theta) + 1 \times \cos(\theta) = \cos(\theta)$$

$$z_2 = 0$$

z-축$(0,0,1)$을 z-축을 중심으로 θ만큼 회전한 결과는 다음과 같다.

$$x_2 = 0 \times \cos(\theta) - 0 \times \sin(\theta) = 0$$

$$y_2 = 0 \times \sin(\theta) + 0 \times \cos(\theta) = 0$$

$$z_2 = 1$$

벡터 (x_1, y_1, z_1)을 z-축을 중심으로 θ만큼 회전한 결과는 다음과 같다.

$$\begin{aligned}(x_1, y_1, z_1)\boldsymbol{R}_z(\theta) &= (x_1, y_1, z_1)\begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ &= (x_1\cos(\theta) - y_1\sin(\theta), x_1\sin(\theta) + y_1\cos(\theta), z_1)\end{aligned}$$

3-차원 좌표계에서 벡터 (x_1, y_1, z_1)을 x-축을 중심으로 θ만큼 회전한 결과는 다음과 같다.

$$x_2 = x_1$$

$$y_2 = y_1\cos(\theta) - z_1\sin(\theta)$$

$$z_2 = y_1\sin(\theta) + z_1\cos(\theta)$$

그러므로 x-축을 중심으로 θ만큼 회전을 하는 변환 행렬 $\boldsymbol{R}_x(\theta)$은 다음과 같다.

$$\boldsymbol{R}_x(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & \sin\theta \\ 0 & -\sin\theta & \cos\theta \end{bmatrix}$$

3-차원 좌표계에서 벡터(점) (x_1, y_1, z_1)을 y-축을 중심으로 θ만큼 회전한 결과는 다음과 같다.

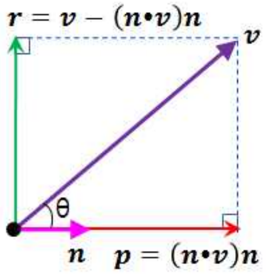
$$x_2 = x_1\cos(\theta) + z_1\sin(\theta)$$
$$y_2 = y_1$$
$$z_2 = -x_1\sin(\theta) + z_1\cos(\theta)$$

그러므로 y-축을 중심으로 θ만큼 회전을 하는 변환 행렬 $\boldsymbol{R}_y(\theta)$은 다음과 같다.

$$\boldsymbol{R}_y(\theta) = \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix}$$

임의의 회전축 $\boldsymbol{n}$을 중심으로 벡터 $\boldsymbol{v}$를 θ만큼 회전(변환)한 결과 벡터 $\boldsymbol{R}_n(\theta)$는 다음과 같이 구할 수 있다. 회전축을 나타내는 벡터 $\boldsymbol{n}$이 단위 벡터일 때, 다음과 같이 벡터 $\boldsymbol{v}$를 $\boldsymbol{n}$에 평행한 벡터 $\boldsymbol{p}$와 수직인 벡터 $\boldsymbol{r}$로 분해(투영: Projection)할 수 있다.

$$\boldsymbol{p} = (\boldsymbol{v} \bullet \boldsymbol{n})\boldsymbol{n}$$
$$\boldsymbol{r} = \boldsymbol{v} - \boldsymbol{p} = \boldsymbol{v} - (\boldsymbol{v} \bullet \boldsymbol{n})\boldsymbol{n}$$

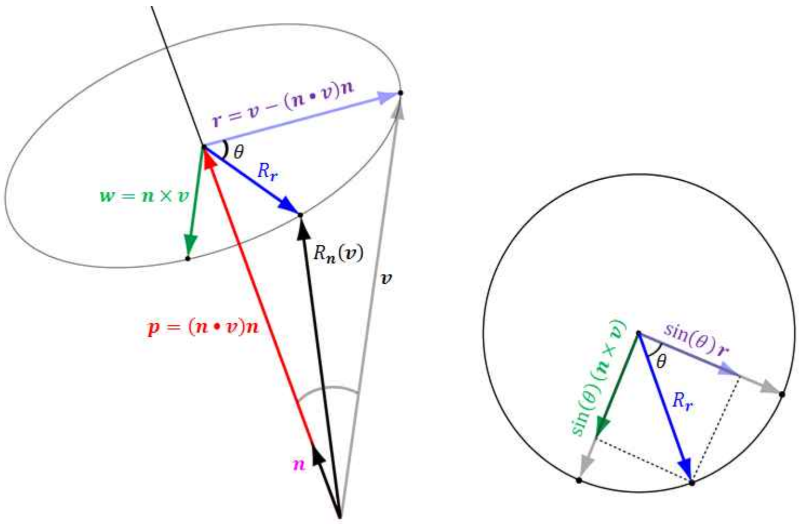


다음 그림에서 다음이 성립한다.

$$\boldsymbol{R}_n(\theta) = \boldsymbol{p} + \boldsymbol{R}_r$$
$$\boldsymbol{w} = \boldsymbol{n} \times \boldsymbol{r} = \boldsymbol{n} \times \boldsymbol{v}$$

벡터 $\boldsymbol{w}$와 벡터 $\boldsymbol{r}$이 수직이고 벡터 $\boldsymbol{w}$와 벡터 $\boldsymbol{p}$가 수직이므로 다음이 성립한다.

$$\boldsymbol{R}_r = \cos(\theta)\boldsymbol{r} + \sin(\theta)\boldsymbol{w}$$
$$\boldsymbol{R}_n(\theta) = \boldsymbol{p} + \boldsymbol{R}_r = \boldsymbol{p} + \cos(\theta)\boldsymbol{r} + \sin(\theta)\boldsymbol{w}$$
$$\boldsymbol{R}_n(\theta) = (\boldsymbol{v} \bullet \boldsymbol{n})\boldsymbol{n} + \cos(\theta)\{\boldsymbol{v} - (\boldsymbol{v} \bullet \boldsymbol{n})\boldsymbol{n}\} + \sin(\theta)(\boldsymbol{n} \times \boldsymbol{v})$$
$$\boldsymbol{R}_n(\theta) = \cos(\theta)\boldsymbol{v} + (1 - \cos(\theta))(\boldsymbol{v} \bullet \boldsymbol{n})\boldsymbol{n} + \sin(\theta)(\boldsymbol{n} \times \boldsymbol{v})$$



위의 식 $R_n(\theta)$는 회전 축 n을 중심으로 벡터 v를 θ만큼 회전한 결과를 나타내는 벡터이다. 이 식은 로드리게스 회전 공식(Rodrigues rotation formula)이라고 한다.

왼손좌표계에서 $n = (x, y, z)$일 때, 로드리게스 회전 공식 $R_n(\theta)$를 행렬을 사용하여 표현하면 다음과 같다. 다음 행렬 R은 회전축 (x, y, z)를 중심으로 θ만큼 회전하는 변환 행렬이다.

$$R_n(\theta) = v\,R$$

$$R = \begin{bmatrix} (1-\cos(\theta))x^2+\cos(\theta) & (1-\cos(\theta))xy+\sin(\theta)z & (1-\cos(\theta))xz-\sin(\theta)y \\ (1-\cos(\theta))xy-\sin(\theta)z & (1-\cos(\theta))y^2+\cos(\theta) & (1-\cos(\theta))yz+\sin(\theta)x \\ (1-\cos(\theta))xz+\sin(\theta)y & (1-\cos(\theta))yz-\sin(\theta)x & (1-\cos(\theta))z^2+\cos(\theta) \end{bmatrix}$$

⑤ 아핀 변환(Affine Transformation)

아핀 공간(Affine Space)은 점들의 집합 W와 벡터 공간 V로 구성된다. 점과 벡터의 관계는 다음의 두 연산으로 정의할 수 있다. P와 Q가 집합 W의 점일 때 벡터 공간 V의 다음과 같은 벡터 v는 유일하다. 이것은 점들에 대한 뺄셈 연산의 결과는 벡터임을 보여준다.

$$v = Q - P$$

집합 W의 모든 점 P와 벡터 공간 V의 모든 벡터 v에 대하여 다음과 같은 점 Q는 유일하다.

$$Q = P + v$$

집합 W에서 고정된 한 점을 원점(Origin) O으로 정의할 수 있다. 그러면 집합 W의 모든 점 P를 다음과 같이 표현할 수 있다.

$$P = O + v$$

n-차원 아핀 공간의 점들의 집합에 대하여 수행할 수 있는 다음과 같은 덧셈 연산을 아핀 결합(Affine combination)이라고 한다.

$$a_1\boldsymbol{P}_1 + a_2\boldsymbol{P}_2 + \cdots + a_n\boldsymbol{P}_n \qquad (a_1 + a_2 + \cdots + a_n = 1)$$

아핀 결합은 다음과 같이 점과 벡터들의 선형 결합으로 표현할 수 있다.

$$a_1 = 1 - (a_2 + \cdots + a_n) = 1 - a_2 - \cdots - a_n$$

$$a_1\boldsymbol{P}_1 + a_2\boldsymbol{P}_2 + \cdots + a_n\boldsymbol{P}_n = (1 - a_2 - \cdots - a_n)\boldsymbol{P}_1 + a_2\boldsymbol{P}_2 + \cdots + a_n\boldsymbol{P}_n$$

$$= \boldsymbol{P}_1 - a_2\boldsymbol{P}_1 - \cdots - a_n\boldsymbol{P}_1 + a_2\boldsymbol{P}_2 + \cdots + a_n\boldsymbol{P}_n$$

$$= \boldsymbol{P}_1 + a_2(\boldsymbol{P}_2 - \boldsymbol{P}_1) + a_3(\boldsymbol{P}_3 - \boldsymbol{P}_1) + \cdots + a_n(\boldsymbol{P}_n - \boldsymbol{P}_1)$$

$$(\boldsymbol{P}_2 - \boldsymbol{P}_1) = \boldsymbol{v}_2,\ (\boldsymbol{P}_3 - \boldsymbol{P}_1) = \boldsymbol{v}_3,\ \cdots,\ (\boldsymbol{P}_n - \boldsymbol{P}_1) = \boldsymbol{v}_n$$

$$a_1\boldsymbol{P}_1 + a_2\boldsymbol{P}_2 + \cdots + a_n\boldsymbol{P}_n = \boldsymbol{P}_1 + a_2\boldsymbol{v}_2 + a_3\boldsymbol{v}_3 + \cdots + a_n\boldsymbol{v}_n$$

아핀 변환은 아핀 공간에서 아핀 공간으로의 대응(함수)이며 다음과 같이 아핀 결합을 보존한다.

$$\boldsymbol{T}(a_1\boldsymbol{P}_1 + a_2\boldsymbol{P}_2 + \cdots + a_n\boldsymbol{P}_n) = a_1\boldsymbol{T}(\boldsymbol{P}_1) + a_2\boldsymbol{T}(\boldsymbol{P}_2) + \cdots + a_n\boldsymbol{T}(\boldsymbol{P}_n)$$

$$a_1 + a_2 + \cdots + a_n = 1$$

m-차원 아핀 공간 $\boldsymbol{V}$에서 n-차원 아핀 공간 $\boldsymbol{W}$로의 모든 아핀 변환 $\boldsymbol{T}$: $\boldsymbol{V} \rightarrow \boldsymbol{W}$은 다음과 같은 형태이다.

$$\boldsymbol{T}(\boldsymbol{x}) = \boldsymbol{x}\boldsymbol{A} + \boldsymbol{y}$$

$\boldsymbol{x}$는 벡터 공간 $\boldsymbol{V}$의 m-차원 벡터이고 $\boldsymbol{A}$는 $(n \times m)$ 선형 변환 행렬이며, $\boldsymbol{y}$는 아핀 공간 $\boldsymbol{W}$의 n-차원 벡터이다. 아핀 변환 $\boldsymbol{T}$를 다음과 같이 블럭 행렬(Block matrix) $\boldsymbol{M}$으로 표현할 수 있다. 행렬 $\boldsymbol{M}$을 아핀 변환 행렬이라고 한다.

$$\boldsymbol{M} = \begin{bmatrix} \boldsymbol{A} & 0 \\ \boldsymbol{y} & 1 \end{bmatrix}$$

$$[\boldsymbol{x} \quad 1]\boldsymbol{M} = [\boldsymbol{x} \quad 1]\begin{bmatrix} \boldsymbol{A} & 0 \\ \boldsymbol{y} & 1 \end{bmatrix} = [\boldsymbol{x}\boldsymbol{A} + \boldsymbol{y} \quad 1]$$

선형 변환은 다음과 같이 블럭 행렬로 표현할 수 있다.

$$[\boldsymbol{x} \quad 0]\begin{bmatrix} \boldsymbol{A} & 0 \\ 0 & 1 \end{bmatrix} = [\boldsymbol{x}\boldsymbol{A} \quad 0]$$

아핀 변환 행렬 $\boldsymbol{M}$의 역행렬 $\boldsymbol{M}^{-1}$은 다음과 같다.

$$\boldsymbol{M}\boldsymbol{M}^{-1} = \begin{bmatrix} \boldsymbol{A} & 0 \\ \boldsymbol{y} & 1 \end{bmatrix}\begin{bmatrix} \boldsymbol{A} & 0 \\ \boldsymbol{y} & 1 \end{bmatrix}^{-1} = \boldsymbol{I} = \begin{bmatrix} \boldsymbol{I} & 0 \\ 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} \boldsymbol{A} & 0 \\ \boldsymbol{y} & 1 \end{bmatrix}\begin{bmatrix} \boldsymbol{A} & 0 \\ \boldsymbol{y} & 1 \end{bmatrix}^{-1} = \begin{bmatrix} \boldsymbol{I} & 0 \\ 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} I & 0 \\ -y & 1 \end{bmatrix}\begin{bmatrix} A & 0 \\ y & 1 \end{bmatrix}\begin{bmatrix} A & 0 \\ y & 1 \end{bmatrix}^{-1} = \begin{bmatrix} I & 0 \\ -y & 1 \end{bmatrix}\begin{bmatrix} I & 0 \\ 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} A & 0 \\ -yA+y & 1 \end{bmatrix}\begin{bmatrix} A & 0 \\ y & 1 \end{bmatrix}^{-1} = \begin{bmatrix} I & 0 \\ -y & 1 \end{bmatrix}$$

$$\begin{bmatrix} A^{-1} & 0 \\ -yA^{-1}+y & 1 \end{bmatrix}\begin{bmatrix} A & 0 \\ -yA+y & 1 \end{bmatrix}\begin{bmatrix} A & 0 \\ y & 1 \end{bmatrix}^{-1} = \begin{bmatrix} A^{-1} & 0 \\ -yA^{-1}+y & 1 \end{bmatrix}\begin{bmatrix} I & 0 \\ -y & 1 \end{bmatrix}$$

$$\begin{bmatrix} I & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} A & 0 \\ y & 1 \end{bmatrix}^{-1} = \begin{bmatrix} A^{-1} & 0 \\ -yA^{-1}+y & 1 \end{bmatrix}\begin{bmatrix} I & 0 \\ -y & 1 \end{bmatrix}$$

$$\begin{bmatrix} A & 0 \\ y & 1 \end{bmatrix}^{-1} = \begin{bmatrix} A^{-1} & 0 \\ -yA^{-1}+y & 1 \end{bmatrix}\begin{bmatrix} I & 0 \\ -y & 1 \end{bmatrix}$$

$$M^{-1} = \begin{bmatrix} A & 0 \\ y & 1 \end{bmatrix}^{-1} = \begin{bmatrix} A^{-1} & 0 \\ -yA^{-1} & 1 \end{bmatrix}$$

벡터 공간 V의 m-차원 벡터 x는 아핀 공간에서 $(m+1)$-차원 벡터 $[x\ 0]$으로 표현할 수 있고 m-차원 점은 아핀 공간에서 m-차원 벡터 $[x\ 1]$로 표현할 수 있다(동차 좌표계: Homogenious coordinates). 아핀 공간에서 $(m+1)$-차원 점 $P_0 = [x_0\ 1]$와 $P_1 = [x_1\ 1]$의 뺄셈은 다음과 같이 아핀 공간에서 $(m+1)$-차원 벡터가 된다.

$$P_0 - P_1 = [x_0\ 1] - [x_1\ 1] = [x_0 - x_1\ 0]$$

$$T(P_0 - P_1) = T(P_0) - T(P_1)$$

아핀 공간의 $(m+1)$-차원 벡터 $[x\ 0]$와 아핀 변환 행렬 M의 곱은 다음과 같다. 이것은 m-차원 벡터 공간에서의 선형 변환과 같다. 즉, 아핀 변환 행렬을 벡터에 대한 선형 변환 행렬로 사용할 수 있음을 의미한다.

$$[x\ \ 0]M = [x\ \ 0]\begin{bmatrix} A & 0 \\ 0 & 1 \end{bmatrix} = [xA\ \ 0]$$

m-차원 벡터 공간에서의 선형 변환 행렬은 $(m \times m)$ 행렬이다. m-차원 아핀 공간에서의 아핀 변환 행렬은 $(m+1)\times(m+1)$ 행렬이다. 3-차원 공간에서의 선형 변환 행렬은 (3×3) 행렬이다. 3-차원 공간에서의 아핀 변환 행렬은 (4×4) 행렬이다.

아핀 변환은 공선형성(Collinearity)을 보존한다. 예를 들어, 한 직선 위의 두 점 P_0와 P_1을 아핀 변환을 하면, 변환된 결과(점) $T(P_0)$와 $T(P_1)$도 같은 직선 위에 있다.

$$L(t) = (1-t)P_0 + tP_1$$

$$T(L(t)) = T((1-t)P_0 + tP_1) = (1-t)T(P_0) + tT(P_1)$$

그리고 같은 평면 위의 세 점 P_0, P_1, P_2을 아핀 변환을 하면 변환된 결과(점) $T(P_0)$, $T(P_1)$, $T(P_2)$도 같은 평면 위에 있다.

$$P(s,t) = (1-s-t)P_0 + sP_1 + tP_2$$

$$T(P(s,t)) = T((1-s-t)P_0 + sP_1 + tP_2)$$
$$T(P(s,t)) = (1-s-t)T(P_0) + sT(P_1) + tT(P_2)$$

아핀 변환은 강체 변환(상대적인 거리 비율)을 보존한다. 두 점 P_0와 P_1에서 거리 d인 점은 $T(P_0)$와 $T(P_1)$에서 거리 d인 점으로 변환된다. 이것은 아핀 변환에 의하여 물체의 모양이 변형되지는 않는다는 것을 의미한다.

$\{v_1, v_2, \cdots, v_m\}$는 m-차원 아핀 공간 V의 기저 벡터(Basis Vector)들의 집합이고 O_V는 아핀 공간 V의 원점이다. m-차원 아핀 공간 V에서의 임의의 점 P는 다음과 같이 기저 벡터들과 원점의 합으로 표현할 수 있다.

$$P = x_1v_1 + x_2v_2 + \cdots + x_mv_m + O_V$$

점 P를 순서쌍으로 표현하면 $P = (x_1, x_2, \cdots, x_m)$이다.

m-차원 아핀 공간 V의 점 $P = (x_1, x_2, \cdots, x_m)$에서 n-차원 아핀 공간 W로의 아핀 변환 T는 다음과 같이 표현할 수 있다.

$$T : V \rightarrow W$$
$$T(P) = T(x_1v_1 + x_2v_2 + \cdots + x_mv_m + O_V) = x_1T(v_1) + x_2T(v_2) + \cdots + x_mT(v_m) + T(O_V)$$

$\{w_1, w_2, \cdots, w_n\}$는 n-차원 아핀 공간 W의 기저 벡터(Basis Vector)들의 집합이고 O_W는 아핀 공간 W의 원점이다. n-차원 아핀 공간 W의 임의의 점 Q는 다음과 같이 기저 벡터들의 선형결합과 원점의 합으로 표현할 수 있다. 점 Q를 순서쌍으로 표현하면 $Q = (b_1, b_2, \cdots, b_n)$이다.

$$Q = b_1w_1 + b_2w_2 + \cdots + b_nw_n + O_W$$

$T(v_j)$는 아핀 공간 W의 벡터이므로 $T(v_j)$는 다음과 같은 아핀 결합으로 표현할 수 있다. $(a_{1j}, a_{2j}, \cdots, a_{nj})$는 아핀 공간 W의 벡터이고 아핀 공간 V에서의 기저 벡터 v_j가 변환 T에 의하여 변환된 결과를 나타낸다.

$$T(v_j) = a_{1j}w_1 + a_{2j}w_2 + \cdots + a_{nj}w_n = (a_{1j}, a_{2j}, \cdots, a_{nj})$$

$T(O_V)$는 아핀 공간 W의 점이므로 $T(O_V)$는 다음과 같은 아핀 결합으로 표현할 수 있다.

$$T(O_V) = y_1w_1 + y_2w_2 + \cdots + y_nw_n + O_W = (y_1, y_2, \cdots, y_n)$$

다음은 아핀 공간 V의 점 P가 아핀 공간 W의 점 Q로 변환되는 것을 표현한다.

$$T(P) = T(x_1v_1 + x_2v_2 + \cdots + x_mv_m + O_V) = x_1T(v_1) + x_2T(v_2) + \cdots + x_mT(v_m) + T(O_V)$$

$$T(P) = x_1(a_{11}, a_{21}, \cdots, a_{n1}) + x_2(a_{12}, a_{22}, \cdots, a_{n2}) + \cdots + x_m(a_{1m}, a_{2m}, \cdots, a_{nm}) + (y_1, y_2, \cdots, y_n)$$

$$T(P) = (b_1, b_2, \cdots, b_n) = Q$$

위의 식에서 다음이 성립한다.

$$b_1 = a_{11}x_1 + a_{12}x_2 + \cdots + a_{1m}x_m + y_1$$

$$b_2 = a_{21}x_1 + a_{22}x_2 + \cdots + a_{2m}x_m + y_2$$

$$b_i = a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{im}x_m + y_i$$

$$b_n = a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nm}x_m + y_n$$

위의 식을 행렬을 사용하여 표현하면 다음과 같다.

$$(b_1, b_2, \cdots, b_{n-1}, b_n, 1) = (x_1, x_2, \cdots, x_{m-1}, x_m, 1)\begin{bmatrix} a_{11} & a_{21} & \cdots & a_{n1} & 0 \\ a_{12} & a_{22} & \cdots & a_{n2} & 0 \\ \cdots & & \cdots & \cdots & \\ a_{1m} & a_{2m} & \cdots & a_{nm} & 0 \\ y_1 & y_2 & \cdots & y_n & 1 \end{bmatrix}$$

$$A = \begin{bmatrix} a_{11} & a_{21} & \cdots & a_{n1} & 0 \\ a_{12} & a_{22} & \cdots & a_{n2} & 0 \\ \cdots & \cdots & \cdots & & \\ a_{1m} & a_{2m} & \cdots & a_{nm} & 0 \\ y_1 & y_2 & \cdots & y_n & 1 \end{bmatrix} \qquad b = (b_1, b_2, \cdots, b_n, 1)$$

$$b = xA$$

위의 식은 아핀 공간 V의 m-차원 벡터 x를 아핀 공간 W로의 n-차원 벡터 b로 변환하는 것은 $(m+1)\times(n+1)$ 행렬 M으로 표현할 수 있음을 보여준다. 그리고 행렬 M의 j-번째 행 벡터는 아핀 공간 V의 기저 벡터 v_j가 변환 T에 의하여 변환된 결과임을 보여준다. 그리고 행렬 M의 마지막 행은 아핀 공간 V의 원점이 변환 T에 의하여 변환된 결과임을 보여준다. 즉, 어떤 아핀 변환을 행렬로 표현하기 위하여 아핀 공간 V의 기저 벡터 v_j를 변환 T로 변환한 결과를 행렬의 j-번째 행벡터로 만들고 아핀 공간 V의 원점을 변환 T에 의하여 변환된 결과를 $(m+1)$ 행벡터로 만들면 된다는 것이다. 행렬 M을 아핀 변환 행렬이라고 한다. 위의 식을 블록 행렬로 표현하면 다음과 같다. 행렬 A는 선형 변환 행렬이고 $y = T(O_V)$이다.

$$[x \quad 1]M = [x \quad 1]\begin{bmatrix} A & 0 \\ y & 1 \end{bmatrix} = [xA + y \quad 1]$$

$$A = \begin{bmatrix} a_{11} & a_{21} & \cdots & a_{n1} \\ a_{12} & a_{22} & \cdots & a_{n2} \\ \cdots & \cdots & \cdots & \cdots \\ a_{1m} & a_{2m} & \cdots & a_{nm} \end{bmatrix}$$

3-차원 아핀 변환은 3-차원 게임의 객체들을 처리(객체의 이동과 회전 등)하기 위한 중요한 수단이다. 모든 선형 변환 행렬 A는 다음과 같이 아핀 변환 행렬 M으로 표현할 수 있다. 그리고 점 벡터 x를 아핀 변환 행렬 M으로 변환하는 것은 점 벡터 x를 선형 변환 행렬 A로 변환하는 것과 같다.

$$M = \begin{bmatrix} A & 0 \\ 0 & 1 \end{bmatrix}$$

$$[x \quad 1]M = [x \quad 1]\begin{bmatrix} A & 0 \\ 0 & 1 \end{bmatrix} = [xA \quad 1]$$

그리고 벡터 v를 선형 변환 행렬 A로 변환하는 것은 벡터 $(v, 0)$를 아핀 변환 행렬 M으로 변환하는 것과 같다.

$$[v \quad 0]M = [v \quad 0]\begin{bmatrix} A & 0 \\ y & 1 \end{bmatrix} = [vA \quad 0]$$

$$v = xA$$

3-차원 선형 변환을 아핀 변환으로 표현할 수 있으므로, 선형 변환 행렬을 아핀 변환 행렬로 표현할 수 있다. 3-차원 아핀 변환 행렬이 (4×4) 행렬이므로 3-차원 선형 변환 행렬도 (4×4) 행렬로 표현할 수 있다.

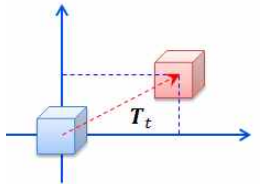
대표적인 아핀 변환의 예는 평행 이동(Translation)과 좌표계의 변환(Coordinate transformation)이다.

- 평행 이동(Translation) 변환

  3-차원 아핀 공간(R^3) V에서 3-차원 아핀 공간(R^3) W로의 평행 이동은 아핀 공간 V의 기저 벡터가 u, v, w이고 원점이 O_V일 때 아핀 공간 V의 점(Point) P를 아핀 공간 W의 점 $T(P)$로 변환하는 아핀 변환이다.

  $$T(P) = T(xu + yv + zw + O_V) = xT(u) + yT(v) + zT(w) + T(O_V)$$

  벡터 a를 평행 이동 변환을 한 결과는 벡터 a와 같다(평행 이동에 의하여 겹쳐지는 벡터는 서로 같다: 벡터의 상등의 정의). 즉, 벡터(크기와 방향)는 평행 이동에 의하여 변하지 않는다. 점 P를 평행 이동 변환을 하는 것은 점 P에 벡터 $t = (t_x, t_y, t_z)$를 더하는 것과 같다. 즉, 평행 이동 변환을 하면 점의 위치가 변하게 된다.

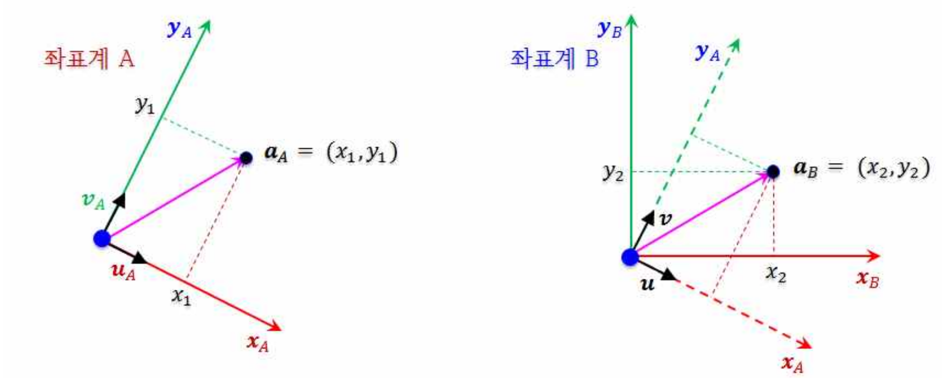


아핀 공간 V와 W가 같다면 평행 이동 변환은 다음과 같은 행렬 T로 표현할 수 있다.

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ t_x & t_y & t_z & 1 \end{bmatrix} = \begin{bmatrix} I & 0 \\ t & 1 \end{bmatrix}$$

- 좌표계의 변환(Coordinate Transformation)

  좌표계의 변환은 한 좌표계의 점 또는 벡터를 다른 좌표계로 표현하는 것을 의미한다. 다음 그림은 벡터에 대한 2-차원 좌표계의 변환을 나타내고 있다.



2-차원 좌표계 A의 기저 벡터가 $\boldsymbol{u}_A$와 $\boldsymbol{v}_A$일 때 벡터 $\boldsymbol{a}_A$는 다음과 같이 표현할 수 있다.

$$\boldsymbol{a}_A = (x_1, y_1) = x_1\boldsymbol{u}_A + y_1\boldsymbol{v}_A$$

2-차원 좌표계 B의 기저 벡터가 $\boldsymbol{u}_B$와 $\boldsymbol{v}_B$일 때 벡터 $\boldsymbol{a}_B$는 다음과 같이 표현할 수 있다.

$$\boldsymbol{a}_B = (x_2, y_2) = x_2\boldsymbol{u}_B + y_2\boldsymbol{v}_B = \boldsymbol{T}(x_1\boldsymbol{u}_A) + \boldsymbol{T}(y_1\boldsymbol{v}_A) = x_1\boldsymbol{u} + y_1\boldsymbol{v}$$

$$\boldsymbol{a}_B = (x_2, y_2) = x_2\boldsymbol{u}_B + y_2\boldsymbol{v}_B = x_1\boldsymbol{T}(\boldsymbol{u}_A) + y_1\boldsymbol{T}(\boldsymbol{v}_A) = x_1\boldsymbol{u} + y_1\boldsymbol{v}$$

벡터 $\boldsymbol{u}$와 $\boldsymbol{v}$는 좌표계 A의 기저 벡터 $\boldsymbol{u}_A$와 $\boldsymbol{v}_A$를 좌표계 B로 표현한 벡터이다. 좌표계 변환을 해도 벡터의 길이 x_1과 y_1는 변하지 않는다. 위의 수식은 좌표계 A의 벡터 $\boldsymbol{a}_A$를 좌표계 B로 표현하려면 좌표계 A의 (x_1, y_1)과 좌표계 B의 벡터 $\boldsymbol{u}$와 $\boldsymbol{v}$를 알아야 함을 나타낸다. 즉, 벡터 $\boldsymbol{a}_A = (x_1, y_1)$를 좌표계 B로 표현하려면 좌표계 A의 기저 벡터 $\boldsymbol{u}_A$와 $\boldsymbol{v}_A$가 좌표계 B의 어떤 벡터로 변환되는 가를 알아야 한다.

3-차원좌표계 A의 기저 벡터가 $\boldsymbol{u}_A$, $\boldsymbol{v}_A$, $\boldsymbol{w}_A$일 때 벡터 $\boldsymbol{a}_A$는 다음과 같이 표현할 수 있다.

$$\boldsymbol{a}_A = (x_1, y_1, z_1) = x_1\boldsymbol{u}_A + y_1\boldsymbol{v}_A + z_1\boldsymbol{w}_A$$

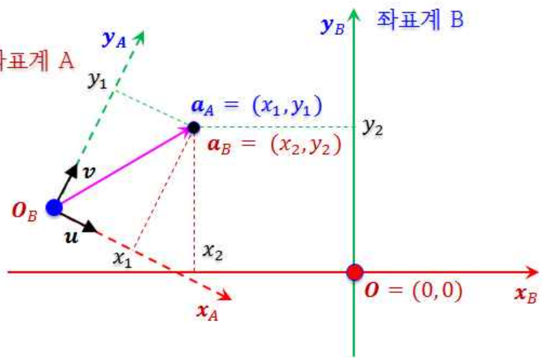
3-차원좌표계 B의 기저 벡터가 $\boldsymbol{u}_B$, $\boldsymbol{v}_B$, $\boldsymbol{w}_B$일 때 벡터 $\boldsymbol{a}_B$는 다음과 같이 표현할 수 있다.

$$a_B = (x_2, y_2, z_2) = x_2 u_B + y_2 v_B + z_2 w_B = T(x_1 u_A) + T(y_1 v_A) + T(z_1 w_A)$$

$$a_B = x_2 u_B + y_2 v_B + z_2 w_B = x_1 T(u_A) + y_1 T(v_A) + z_1 T(w_A) = x_1 u + y_1 v + z_1 w$$

벡터 u, v, w는 좌표계 A의 기저 벡터 u_A, v_A, w_A를 좌표계 B로 표현한 벡터이다.

다음 그림은 점에 대한 2-차원 좌표계의 변환을 나타내고 있다.



2-차원 좌표계 A의 기저 벡터가 u_A와 v_A일 때 점 a_A는 다음과 같이 표현할 수 있다. O_A는 좌표계 A의 원점이고 $O_A = (0,0)$으로 가정한다(모든 좌표계의 원점을 0으로 가정하자).

$$a_A = (x_1, y_1) = x_1 u_A + y_1 v_A + O_A = x_1 u_A + y_1 v_A$$

2-차원 좌표계 B의 기저 벡터가 u_B와 v_B일 때 점 a_B는 다음과 같이 표현할 수 있다.

$$a_B = (x_2, y_2) = x_2 u_B + y_2 v_B = T(x_1 u_A) + T(y_1 v_A) + T(O_A) = x_1 u + y_1 v + O_{AB}$$

$$a_B = (x_2, y_2) = x_2 u_B + y_2 v_B = x_1 T(u_A) + y_1 T(v_A) + O_{AB} = x_1 u + y_1 v + O_{AB}$$

벡터 u와 v는 좌표계 A의 기저 벡터 u_A와 v_A를 좌표계 B로 표현한 벡터이고 점 O_{AB}는 좌표계 A의 원점 O_A를 좌표계 B로 표현한 것이다. 좌표계 변환을 해도 벡터의 길이는 변하지 않는다. 위의 수식은 좌표계 A의 점 a_A를 좌표계 B로 표현하려면 좌표계 A의 (x_1, y_1)과 좌표계 B의 벡터 u와 v 그리고 점 O_{AB}를 알아야 함을 나타낸다. 즉, 벡터 $a_A = (x_1, y_1)$를 좌표계 B로 표현하려면 좌표계 A의 기저 벡터 u_A와 v_A가 좌표계 B의 어떤 벡터로 변환되는 가를 알아야 하고, 좌표계 A의 원점 O_A를 좌표계 B로 표현한 점 O_{AB}를 알아야 한다.

3-차원좌표계 A의 기저 벡터가 u_A, v_A, w_A일 때 점 a_A는 다음과 같이 표현할 수 있다. O_A는 좌표계 A의 원점이고 $O_A = (0,0,0)$으로 가정한다.

$$a_A = (x_1, y_1, z_1) = x_1 u_A + y_1 v_A + z_1 w_A + O_A = x_1 u_A + y_1 v_A + z_1 w_A$$

3-차원좌표계 B의 기저 벡터가 u_B, v_B, w_B일 때 점 a_B는 다음과 같이 표현할 수 있다.

$$\boldsymbol{a}_B = (x_2, y_2, z_2) = x_2\boldsymbol{u}_B + y_2\boldsymbol{v}_B + z_2\boldsymbol{w}_B = \boldsymbol{T}(x_1\boldsymbol{u}_A) + \boldsymbol{T}(y_1\boldsymbol{v}_A) + \boldsymbol{T}(z_1\boldsymbol{w}_A) + \boldsymbol{T}(\boldsymbol{O}_A)$$

$$\boldsymbol{a}_B = x_1\boldsymbol{T}(\boldsymbol{u}_A) + y_1\boldsymbol{T}(\boldsymbol{v}_A) + z_1\boldsymbol{T}(\boldsymbol{w}_A) + \boldsymbol{T}(\boldsymbol{O}_A) = x_1\boldsymbol{u} + y_1\boldsymbol{v} + z_1\boldsymbol{w} + \boldsymbol{O}_{AB}$$

벡터 $\boldsymbol{u}$, $\boldsymbol{v}$, $\boldsymbol{w}$는 좌표계 A의 기저 벡터 $\boldsymbol{u}_A$, $\boldsymbol{v}_A$, $\boldsymbol{w}_A$를 좌표계 B로 표현한 벡터이고 점 $\boldsymbol{O}_{AB}$는 좌표계 A의 원점 $\boldsymbol{O}_A$를 좌표계 B로 표현한 것이다.

3-차원 좌표계 변환을 행렬을 사용하여 표현하도록 하자. 3-차원 좌표계 A의 벡터 $\boldsymbol{a}_A = (x_1, y_1, z_1)$를 좌표계 B로 표현한 벡터 $\boldsymbol{a}_B$는 다음과 같다.

$$\boldsymbol{a}_B = (x_2, y_2, z_2) = x_1\boldsymbol{u} + y_1\boldsymbol{v} + z_1\boldsymbol{w}$$

3-차원 좌표계 A의 점 $\boldsymbol{a}_A = (x_1, y_1, z_1)$를 좌표계 B로 표현한 점 $\boldsymbol{a}_B$는 다음과 같다.

$$\boldsymbol{a}_B = (x_2, y_2, z_2) = x_2\boldsymbol{u}_B + y_2\boldsymbol{v}_B + z_2\boldsymbol{w}_B = x_1\boldsymbol{u} + y_1\boldsymbol{v} + z_1\boldsymbol{w} + \boldsymbol{O}_{AB}$$

벡터와 점의 변환을 한꺼번에 표현하기 위하여 3-차원 좌표계 A의 점(벡터)와 3-차원 좌표계 B의 점(벡터)를 동차 좌표계(Homogenious coordinates)를 사용하여 4-차원 점(벡터)로 표현하도록 하자. 3-차원 벡터 (x, y, z)는 4-차원 벡터 $(x, y, z, 0)$로 표현할 수 있다. 3-차원 점 (x, y, z)는 4-차원 점 $(x, y, z, 1)$로 표현할 수 있다. 다음 식에서 실수 k는 벡터를 표현할 때 0이고, 점을 표현할 때는 1이다.

$$\boldsymbol{a}_B = (x_2, y_2, z_2, k) = x_1\boldsymbol{u} + y_1\boldsymbol{v} + z_1\boldsymbol{w} + k\boldsymbol{O}_{AB}$$

벡터 $\boldsymbol{u}$, $\boldsymbol{v}$, $\boldsymbol{w}$는 좌표계 B의 벡터이므로 다음과 같이 표현할 수 있다. 벡터 $\boldsymbol{u}$는 좌표계 A의 기저 벡터 $\boldsymbol{u}_A$(x-축의 단위 벡터)를 좌표계 B로 표현한 것이다. 벡터 $\boldsymbol{v}$는 좌표계 A의 기저 벡터 $\boldsymbol{v}_A$(y-축의 단위 벡터)를 좌표계 B로 표현한 것이다 벡터 $\boldsymbol{w}$는 좌표계 A의 기저 벡터 $\boldsymbol{w}_A$(z-축의 단위 벡터)를 좌표계 B로 표현한 것이다.

$$\boldsymbol{u} = (u_x, u_y, u_z, 0),\ \boldsymbol{v} = (v_x, v_y, v_z, 0),\ \boldsymbol{w} = (w_x, w_y, w_z, 0)$$

점 $\boldsymbol{O}_B$는 좌표계 B의 점이므로 다음과 같이 표현할 수 있다.

$$\boldsymbol{O}_B = (O_x, O_y, O_z, 1)$$

식 $(x_2, y_2, z_2, k) = x_1\boldsymbol{u} + y_1\boldsymbol{v} + z_1\boldsymbol{w} + k\boldsymbol{O}_{AB}$을 요소별로 정리하면 다음과 같다.

$$(x_2, y_2, z_2, k) = x_1(u_x, u_y, u_z, 0) + y_1(v_x, v_y, v_z, 0) + z_1(w_x, w_y, w_z, 0) + k(O_x, O_y, O_z, 1)$$

$$= (x_1u_x + y_1v_x + z_1w_x + kO_x, x_1u_y + y_1v_y + z_1w_y + kO_y, x_1u_z + y_1v_z + z_1w_z + kO_z, k)$$

$$x_2 = x_1u_x + y_1v_x + z_1w_x + kO_x$$

$$y_2 = x_1u_y + y_1v_y + z_1w_y + kO_y$$

$$z_2 = x_1u_z + y_1v_z + z_1w_z + kO_z$$

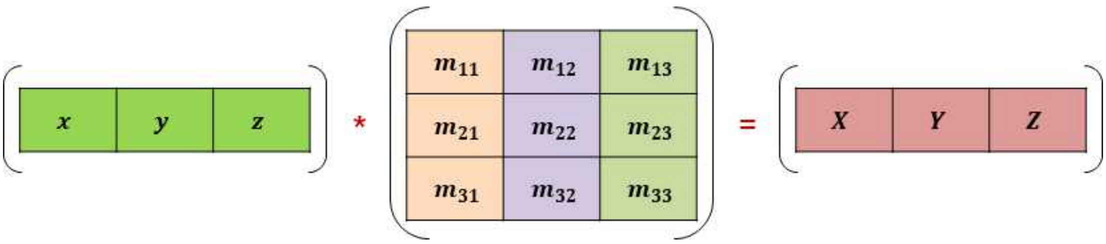
위의 식을 행렬을 사용하여 표현하면 다음과 같다.

$$(x_2, y_2, z_2, k) = (x_1, y_1, z_1, k)\begin{bmatrix} u_x & u_y & u_z & 0 \\ v_x & v_y & v_z & 0 \\ w_x & w_y & w_z & 0 \\ O_x & O_y & O_z & 1 \end{bmatrix}$$

이 결과는 선형 변환을 행렬로 표현하기 위하여 벡터 공간 V의 기저 벡터 v_j를 변환 T로 변환한 결과를 행렬의 j-번째 행벡터로 만들면 된다는 것과 일치하는 것이다. 위의 결과는 3-차원 좌표계 A의 점(벡터)과 3-차원 좌표계 B의 점(벡터)로 선형 변환하기 위해서는 (4×4) 정사각형 행렬을 사용할 수 있음을 보여준다. 이 행렬의 1~3 행벡터는 좌표계 A의 기저 벡터 $\boldsymbol{u}_A$, $\boldsymbol{v}_A$, $\boldsymbol{w}_A$를 좌표계 B로 표현한 벡터 $\boldsymbol{u}$, $\boldsymbol{v}$, $\boldsymbol{w}$가 되고 4번째 행벡터는 좌표계 A의 원점 $\boldsymbol{O}_A$를 좌표계 B로 표현한 점 $\boldsymbol{O}_B$가 된다.

⑥ 벡터와 행렬의 곱에 대한 이해

3-차원 벡터는 (1×3) 행렬로 취급할 수 있으므로 (3×3) 행렬과 곱셈을 할 수 있다. 3-차원 벡터 (x, y, z)와 (3×3) 행렬 $\boldsymbol{M}$의 곱셈의 결과는 (1×3) 행렬 즉, 3-차원 벡터이다. 이것은 벡터 $\boldsymbol{v} = (x, y, z)$를 행렬 $\boldsymbol{M}$과 행렬의 곱셈을 하여 벡터 (X, Y, Z)로 변환할 수 있음을 보여준다.



$$(X, Y, Z) = \boldsymbol{vM} = (x, y, z)\boldsymbol{M}$$

$$X = (x, y, z) \bullet (m_{11}, m_{21}, m_{31}) = xm_{11} + ym_{21} + zm_{31}$$

$$Y = (x, y, z) \bullet (m_{12}, m_{22}, m_{32}) = xm_{12} + ym_{22} + zm_{32}$$

$$Z = (x, y, z) \bullet (m_{13}, m_{23}, m_{33}) = xm_{13} + ym_{23} + zm_{33}$$

행렬 $\boldsymbol{M}$이 단위행렬이면 $(x, y, z) = (x, y, z)\boldsymbol{M}$이다. 이때 벡터 (x, y, z)는 행렬 $\boldsymbol{M}$에 의하여 변환이 되지 않은 것이다.

$$\boldsymbol{M} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$(x, y, z)\boldsymbol{M} = (x, y, z)\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = (x, y, z)$$

행렬 $\boldsymbol{M}$이 단위행렬이 아니면 $(x, y, z) \neq (X, Y, Z)$이다. 즉, 벡터 (x, y, z)를 행렬 $\boldsymbol{M}$에 곱하여 새로운(변환된) 벡터를 구한 것이다. 벡터 (x, y, z)를 행렬 $\boldsymbol{M}$에 곱하여 새로운 벡

터를 구하는 것을 "벡터 (x, y, z)를 행렬 M으로 변환한다"라고 하며, 행렬 M을 변환 행렬(Transformation matrix)이라고 한다. 즉, 벡터(점)에 행렬을 곱하는 것은 벡터(점)을 변환하는 것이고, 행렬은 변환을 수행하는 수학적 수단이다.

행렬 M이 변환 T를 나타내는 행렬일 때 행렬 M의 역 행렬 M^{-1}은 변환 T의 역 변환을 나타내는 행렬이다. 벡터 v_1을 행렬 M으로 변환한 결과가 벡터 v_2라고 하자. 벡터 v_2에 역 행렬 M^{-1}를 곱하면 벡터 v_1이 된다. 이것은 역 행렬 M^{-1}가 변환 T의 역 변환을 나타냄을 보여준다.

$$v_2 = v_1 M$$

$$v_2 M^{-1} = (v_1 M) M^{-1} = v_1 (M M^{-1}) = v_1 I = v_1$$

다음은 회전 변환을 나타내는 행렬의 역 행렬이 역 변환임을 나타낸다. z-축을 중심으로 θ만큼의 회전을 하는 변환의 역 변환은 z-축을 중심으로 $-\theta$만큼의 회전을 하는 것이다. z-축을 중심으로 θ만큼의 회전을 하는 변환을 $R(\theta)$로 표기할 때, z-축을 중심으로 θ만큼의 회전을 하는 변환 $R(\theta)$의 변환 행렬은 다음과 같다.

$$R(\theta) = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$R^{-1}(\theta) = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} = R^T(\theta)$$

$$R(-\theta) = R^{-1}(\theta)$$

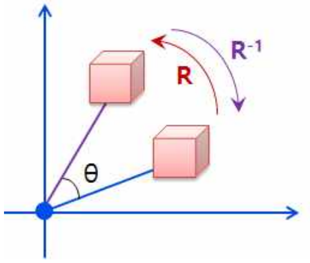
z-축을 중심으로 $-\theta$만큼의 회전을 하는 변환 $R(-\theta)$의 변환 행렬은 다음과 같다.

$$R(-\theta) = \begin{bmatrix} \cos(-\theta) & \sin(-\theta) & 0 \\ -\sin(-\theta) & \cos(-\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$R(\theta)R(-\theta) = \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$\sin(-\theta) = -\sin(\theta)$이고 $\cos(-\theta) = \cos(\theta)$이므로 $R(\theta)R(-\theta) = I$이다. 그러므로 변환 $R(\theta)$의 역 행렬 $R(-\theta)$은 $R(\theta)$의 역 변환 행렬이다.

$$R(-\theta) = R^{-1}(\theta)$$



직교행렬의 역행렬은 전치행렬이므로 $R^{-1}(\theta)=R^{T}(\theta)$이다.

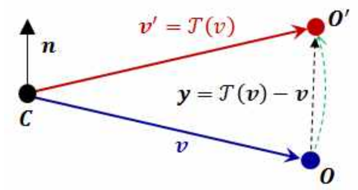
평행 이동 변환 행렬 T의 역행렬은 다음과 같다. 3-차원 좌표계의 점 벡터 $P=(x,y,z)$를 x-축으로 t_x, y-축으로 t_y, z-축으로 t_z 만큼 평행 이동하는 변환 T의 역변환은 x-축으로 $-t_x$, y-축으로 $-t_y$, z-축으로 $-t_z$ 만큼 평행 이동하는 변환이다.

$$T=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\t_x&t_y&t_z&1\end{bmatrix}=\begin{bmatrix}I&0\\t&1\end{bmatrix}$$

$$T^{-1}=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\t_x&t_y&t_z&1\end{bmatrix}^{-1}=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\-t_x&-t_y&-t_z&1\end{bmatrix}=\begin{bmatrix}I&0\\-t&1\end{bmatrix}$$

$$TT^{-1}=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\t_x&t_y&t_z&1\end{bmatrix}\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\-t_x&-t_y&-t_z&1\end{bmatrix}=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix}=I$$

지금까지 회전을 할 때 회전의 중심으로 좌표계의 원점을 사용하였다. 회전의 중심이 좌표계의 원점이 아닌 임의의 점일 때를 고려해보자. 다음 그림에서 점 O가 원점, 점 $C=(c_x,c_y,c_z)$가 회전의 중심, 그리고 벡터 v를 점 C를 중심으로 회전을 한다고 가정하자. 회전 행렬이 R일 때 회전의 결과 $v'=T(v)$는 다음과 같이 표현할 수 있다.



$$v=O-C=(-c_x,-c_y,-c_z)$$

$$T(v)=v'=vR$$

$$T(O)=O'=C+T(v)$$

$$y=T(v)-v=vR-v=v(R-I)=(-v)(I-R)$$

$$O'=O+y=y$$

이것은 점 C를 원점으로 v만큼 평행 이동하고 원점을 중심으로 회전을 한 결과를 다시 $-v$ 만큼 평행 이동한 결과와 같다.

$$M=\begin{bmatrix}I&0\\v&1\end{bmatrix}\begin{bmatrix}R&0\\0&1\end{bmatrix}\begin{bmatrix}I&0\\-v&1\end{bmatrix}=\begin{bmatrix}R&0\\v(R-I)&1\end{bmatrix}$$

위의 점 C를 중심으로 원점을 회전한 결과가 $(O'=y)$이므로 임의의 점 P를 점 C를

중심으로 회전한 결과는 $(P+y)$로 표현할 수 있다.

⑦ 변환의 합성(Composition of Transformation)

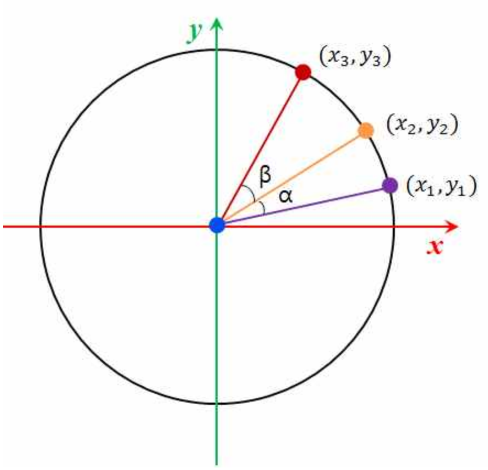
변환 S가 벡터 공간 U에서 벡터 공간 V로의 선형 변환이고 변환 T가 벡터 공간 V에서 벡터 공간 W로의 선형 변환일 때, $(S \circ T)$는 벡터 공간 U에서 벡터 공간 W로의 선형 변환이다. $(S \circ T)$는 벡터 공간 U의 벡터 x를 T로 변환하고, $T(x)$를 S로 변환하는 것을 의미한다. $(S \circ T)$를 변환 S와 변환 T의 합성(Composition) 변환이라고 한다.

$$T \circ S : U \rightarrow W$$

$$(S \circ T)(x) = S(T(x))$$

변환 T에 대한 변환 행렬이 A이고, 변환 S에 대한 변환 행렬이 B이면 변환 $(S \circ T)$에 대한 변환 행렬은 AB이다. 변환을 행렬로 표현할 수 있고 합성 변환을 행렬의 곱으로 표현할 수 있으므로, 여러 가지 순차적인 변환들을 하나의 행렬(행렬들의 곱셈)로 표현할 수 있다.

다음은 회전 변환의 합성은 회전 변환 행렬의 곱과 같음을 나타낸다.



z-축을 중심으로 α만큼 회전을 하고 다시 β만큼 회전을 한 것은 z-축을 중심으로 $(\alpha+\beta)$만큼 회전을 하는 것과 같다.

$$x_2 = x_1 \cos(\alpha) - y_1 \sin(\alpha)$$

$$y_2 = x_1 \sin(\alpha) + y_1 \cos(\alpha)$$

$$x_3 = x_2 \cos(\beta) - y_2 \sin(\beta)$$

$$y_3 = x_2 \sin(\beta) + y_2 \cos(\beta)$$

$$x_3 = x_1 \cos(\alpha+\beta) - y_1 \sin(\alpha+\beta)$$

$$y_3 = x_1 \sin(\alpha+\beta) + y_1 \cos(\alpha+\beta)$$

z-축을 중심으로 α만큼 회전을 하는 변환 R_α의 변환 행렬, z-축을 중심으로 β만큼 회전을 하는 변환 R_β의 변환 행렬, z-축을 중심으로 $(\alpha+\beta)$만큼 회전을 하는 변환 $R_{\alpha+\beta}$의

변환 행렬은 다음과 같다.

$$R_{\alpha}=\begin{bmatrix}\cos(\alpha) & \sin(\alpha) & 0\\ -\sin(\alpha) & \cos(\alpha) & 0\\ 0 & 0 & 1\end{bmatrix}$$

$$R_{\beta}=\begin{bmatrix}\cos(\beta) & \sin(\beta) & 0\\ -\sin(\beta) & \cos(\beta) & 0\\ 0 & 0 & 1\end{bmatrix}$$

$$R_{\alpha+\beta}=\begin{bmatrix}\cos(\alpha+\beta) & \sin(\alpha+\beta) & 0\\ -\sin(\alpha+\beta) & \cos(\alpha+\beta) & 0\\ 0 & 0 & 1\end{bmatrix}$$

$$R_{\alpha+\beta}=R_{\alpha}R_{\beta}=\begin{bmatrix}\cos(\alpha) & \sin(\alpha) & 0\\ -\sin(\alpha) & \cos(\alpha) & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}\cos(\beta) & \sin(\beta) & 0\\ -\sin(\beta) & \cos(\beta) & 0\\ 0 & 0 & 1\end{bmatrix}$$

$$=\begin{bmatrix}\cos(\alpha)\cos(\beta)-\sin(\alpha)\sin(\beta) & \cos(\alpha)\sin(\beta)+\sin(\alpha)\cos(\beta) & 0\\ -\sin(\alpha)\cos(\beta)-\cos(\alpha)\sin(\beta) & -\sin(\alpha)\sin(\beta)+\cos(\alpha)\cos(\beta) & 0\\ 0 & 0 & 1\end{bmatrix}$$

$$=\begin{bmatrix}\cos(\alpha+\beta) & \sin(\alpha+\beta) & 0\\ -\sin(\alpha+\beta) & \cos(\alpha+\beta) & 0\\ 0 & 0 & 1\end{bmatrix}$$

다음은 회전 변환 R과 평행 이동 변환 T를 합성하는 예이다.

$$T=\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ t_x & t_y & t_z & 1\end{bmatrix} \qquad R_z=\begin{bmatrix}\cos(\alpha) & \sin(\alpha) & 0 & 0\\ -\sin(\alpha) & \cos(\alpha) & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

$$R_zT=\begin{bmatrix}\cos(\alpha) & \sin(\alpha) & 0 & 0\\ -\sin(\alpha) & \cos(\alpha) & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ t_x & t_y & t_z & 1\end{bmatrix}=\begin{bmatrix}\cos(\alpha) & \sin(\alpha) & 0 & 0\\ -\sin(\alpha) & \cos(\alpha) & 0 & 0\\ 0 & 0 & 1 & 0\\ t_x & t_y & t_z & 1\end{bmatrix}$$

$$TR_z=\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ t_x & t_y & t_z & 1\end{bmatrix}\begin{bmatrix}\cos(\alpha) & \sin(\alpha) & 0 & 0\\ -\sin(\alpha) & \cos(\alpha) & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

$$=\begin{bmatrix}\cos(\alpha) & \sin(\alpha) & 0 & 0\\ -\sin(\alpha) & \cos(\alpha) & 0 & 0\\ 0 & 0 & 1 & 0\\ t_x\cos(\alpha)-t_y\sin(\alpha) & t_x\sin(\alpha)+t_y\cos(\alpha) & t_z & 1\end{bmatrix}$$

아핀 변환 S와 T의 합성 $(T\circ S)$은 다음과 같이 블록 행렬으로 표현할 수 있다.

$$S=\begin{bmatrix}A & 0\\ y & 1\end{bmatrix}, \qquad T=\begin{bmatrix}B & 0\\ x & 1\end{bmatrix}$$

$$ST=\begin{bmatrix}A & 0\\ y & 1\end{bmatrix}\begin{bmatrix}B & 0\\ x & 1\end{bmatrix}=\begin{bmatrix}AB & 0\\ yB+x & 1\end{bmatrix}$$

(3×3) 선형 변환 행렬을 (4×4) 아핀 변환 행렬로 표현할 수 있으므로 3-차원 좌표계

의 변환 행렬을 (4×4) 아핀 변환 행렬로 표현한다. 3-차원 벡터 $\boldsymbol{v}=(x,y,z)$를 (4×4) 아핀 변환 행렬로 변환하려면 (1×4) 벡터가 되어야 하므로 벡터 $\boldsymbol{v}$를 $(x,y,z,1)$로 표현한다(동차 좌표계: Homogeneous coordinates). 즉, 4-차원 벡터 $(x,y,z,1)$와 3-차원 벡터 (x,y,z)를 같은 것으로 정의한다. 4-차원 벡터 $(x,y,z,1)$를 3-차원 벡터 (x,y,z)와 동차 좌표라고 한다.

임의의 3-차원 벡터 (x,y,z)가 주어질 때 벡터 (x,y,z)를 4-차원 벡터 $(x,y,z,1)$로 대응(매핑: Mapping)할 수 있다. 임의의 4-차원 벡터 (x,y,z,w)가 주어질 때, 벡터 (x,y,z,w)를 3-차원 벡터로 대응(매핑: Mapping)시키는 방법은 (x,y,z,w)의 각 원소를 w로 나누는 것이다(w 나누기). 이것은 4-차원 벡터를 3-차원 벡터로 투영(Projection)하는 것이다.

$$(x,y,z,w)\rightarrow(\frac{x}{w},\frac{y}{w},\frac{z}{w},\frac{w}{w})=(\frac{x}{w},\frac{y}{w},\frac{z}{w},1)\rightarrow(\frac{x}{w},\frac{y}{w},\frac{z}{w})$$

⑧ 행렬 곱셈의 결합 법칙

행렬의 곱은 교환 법칙은 성립하지 않지만 결합 법칙이 성립한다.

$$A(BC)=(AB)C$$

여러 가지 변환을 순서대로 적용할 때 행렬의 곱의 결합 법칙을 사용하면 계산량을 줄일 수 있다. 다음은 행렬의 곱의 결합 법칙이 변환의 계산량을 줄일 수 있음을 보여준다.

- 3-차원 게임 객체에 대하여 n번의 선형 변환을 한다면, 모델 메쉬(Mesh)를 렌더링하기 위하여 모델 메쉬의 각 정점을 n번의 선형 변환을 해야 한다. 모델 메쉬의 정점의 개수가 k개이고 정점의 집합이 $\{\boldsymbol{v}_{10},\boldsymbol{v}_{20},\cdots,\boldsymbol{v}_{k0}\}$라고 가정하자.

$(n=4)$일 때와 $(n=10)$일 때의 연산의 양을 계산하자. $\boldsymbol{A}_j$는 j-번째 변환 행렬이다 $(1\le j\le n)$. $\boldsymbol{v}_{ij}$는 i-번째 정점에 대하여 변환 $\boldsymbol{A}_j$를 하여 얻은 변환의 결과이다. 다음과 같이 모든 정점에 대하여 순차적으로 변환을 하는 경우와 결합 법칙을 사용하는 경우의 연산의 양을 비교하자. 3-차원 벡터와 (3×3) 선형 변환 행렬의 곱을 할 때 15번의 연산(곱셈과 덧셈)이 필요하다.

▫ i-번째 정점에 대하여 순차적으로 변환을 하는 경우

$\boldsymbol{v}_{i0}\boldsymbol{A}_1 = \boldsymbol{v}_{i1}$

$\boldsymbol{v}_{i1}\boldsymbol{A}_2 = \boldsymbol{v}_{i2}$

$\boldsymbol{v}_{i2}\boldsymbol{A}_3 = \boldsymbol{v}_{i3}$

$\cdots$

$\boldsymbol{v}_{i(n-2)}\boldsymbol{A}_{n-1} = \boldsymbol{v}_{i(n-1)}$

$\boldsymbol{v}_{i(n-1)}\boldsymbol{A}_n = \boldsymbol{v}_{in}$

$(n=4)$, $(k=10000)$: 연산량 $=(15\times 4)\times 10000 = 600000$

$(n=10)$, $(k=10000)$: 연산량 $=(15\times 10)\times 10000 = 1500000$

▫ 결합 법칙을 사용하는 경우

$\boldsymbol{v}_{i0}\boldsymbol{A}_1 = \boldsymbol{v}_{i1}$

$\boldsymbol{v}_{i1}\boldsymbol{A}_2 = (\boldsymbol{v}_{i0}\boldsymbol{A}_1)\boldsymbol{A}_2 = \boldsymbol{v}_{i2}$

$\boldsymbol{v}_{i2}\boldsymbol{A}_3 = ((\boldsymbol{v}_{i0}\boldsymbol{A}_1)\boldsymbol{A}_2)\boldsymbol{A}_3 = \boldsymbol{v}_{i3}$

$\cdots$

$\boldsymbol{v}_{i(n-2)}\boldsymbol{A}_{n-1} = ((((\boldsymbol{v}_{i0}\boldsymbol{A}_1)\boldsymbol{A}_2)\boldsymbol{A}_3)\cdots)\boldsymbol{A}_{n-1} = \boldsymbol{v}_{i(n-1)}$

$\boldsymbol{v}_{i(n-1)}\boldsymbol{A}_n = (((((\boldsymbol{v}_{i0}\boldsymbol{A}_1)\boldsymbol{A}_2)\boldsymbol{A}_3)\cdots)\boldsymbol{A}_{n-1})\boldsymbol{A}_n) = \boldsymbol{v}_{in}$

$\boldsymbol{v}_{in} = \boldsymbol{v}_{i0}(\boldsymbol{A}_1\boldsymbol{A}_2\boldsymbol{A}_3\cdots\boldsymbol{A}_{n-1}\boldsymbol{A}_n) = \boldsymbol{v}_{i0}\boldsymbol{M}$

$(\boldsymbol{A}_1\boldsymbol{A}_2\boldsymbol{A}_3\cdots\boldsymbol{A}_{n-1}\boldsymbol{A}_n) = \boldsymbol{M}$

두 개의 (3×3) 선형 변환 행렬의 곱을 할 때 45번의 연산(곱셈과 덧셈)이 필요하다.

$(n=4)$, $(k=10000)$: 연산량 $=(45\times 3)+(15\times 10000) = 150135$

$(n=10)$, $(k=10000)$: 연산량 $=(45\times 9)+(15\times 10000) = 150405$

순차적으로 변환을 하는 경우와 결합 법칙을 사용하는 경우의 연산의 양을 비교하면 $(n=4)$일 때 600,000번과 150,135번이고 $(n=10)$일 때 1,500,000번과 150,405번이다. 이 결과는 결합 법칙을 사용하는 경우 변환의 횟수 n에 관계없이 연산량이 거의 일정함을 보이고 있다.