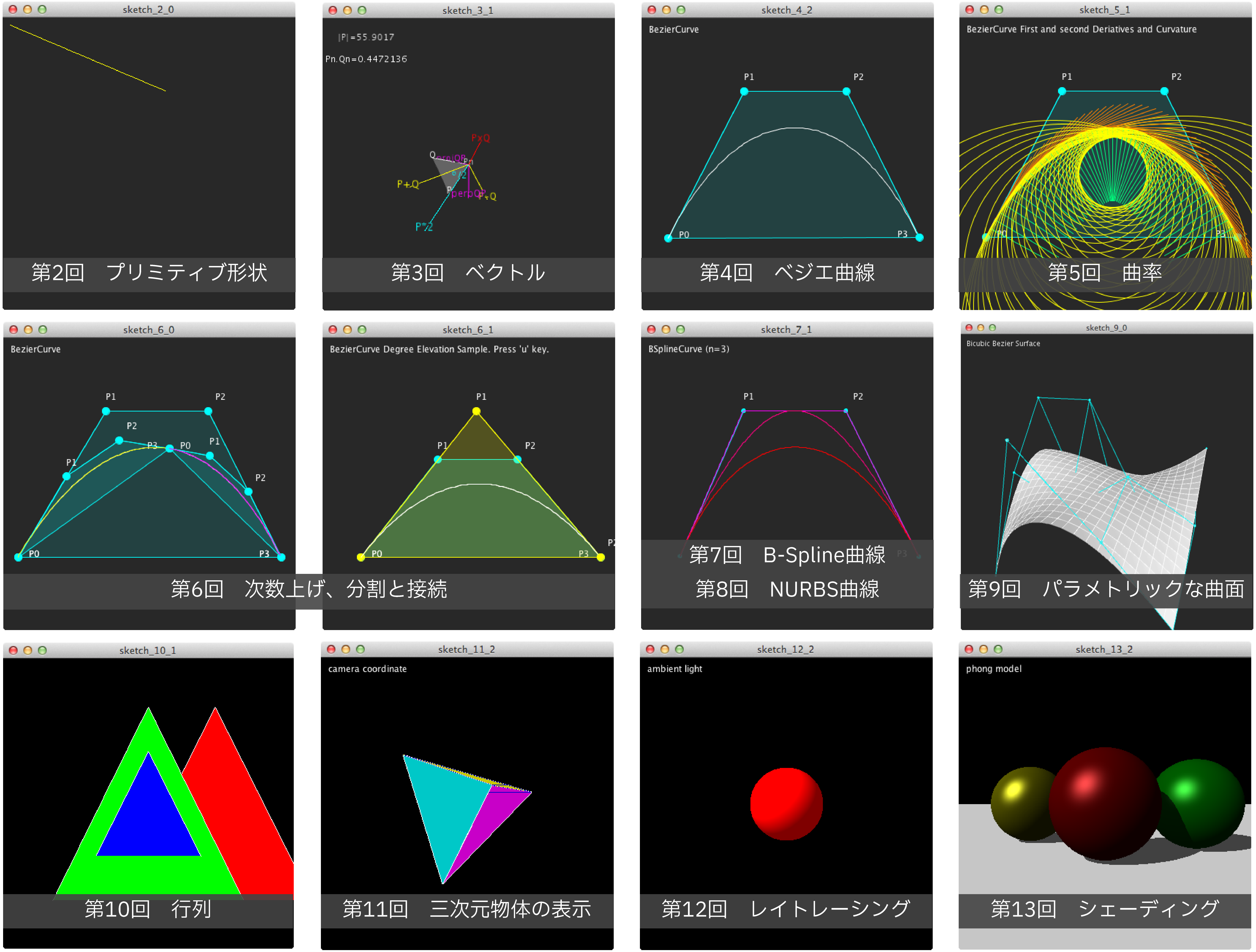
CGとCADの数理

GEOMETRIC MODELING AND COMPUTER GRAPHICS

第10回 行列



sketch_2_0
第2回　プリミティブ形状
sketch_3_1
|P|=55.9017
Pn.Qn=0.4472136
PxQ
P+Q
P-Q
P*2
第3回　ベクトル
sketch_4_2
BezierCurve
P1
P2
P0
P3
第4回　ベジエ曲線
sketch_5_1
BezierCurve First and second Deriatives and Curvature
P1
P2
第5回　曲率
sketch_6_0
BezierCurve
sketch_6_1
BezierCurve Degree Elevation Sample. Press 'u' key.
第6回　次数上げ、分割と接続
sketch_7_1
BSplineCurve (n=3)
第7回　B-Spline曲線
第8回　NURBS曲線
sketch_9_0
Bicubic Bezier Surface
第9回　パラメトリックな曲面
sketch_10_1
第10回　行列
sketch_11_2
camera coordinate
第11回　三次元物体の表示
sketch_12_2
ambient light
第12回　レイトレーシング
sketch_13_2
phong model
第13回　シェーディング

sketch_10_0.zip をダウンロードして下さい

3次元ベクトルと 3x3行列の積（移動、回転、拡大・縮小）

$$v_o = v_i \; M$$

$$= \left(v_i[0], v_i[1], v_i[2]\right)$$

$$\begin{bmatrix} M_{00} & M_{01} & M_{02} \\ M_{10} & M_{11} & M_{12} \\ M_{20} & M_{21} & M_{22} \end{bmatrix}$$

3次元ベクトルと 3x3行列の積（移動、回転、拡大・縮小）

$$v_o = v_i \, M$$

$$= (v_i[0], v_i[1], v_i[2])$$

$$\begin{bmatrix} M_{00} & M_{01} & M_{02} \\ M_{10} & M_{11} & M_{12} \\ M_{20} & M_{21} & M_{22} \end{bmatrix}$$

$$v_o[0] = v_i[0] \, M_{00} + v_i[1] \, M_{10} + v_i[2] \, M_{20}$$

3次元ベクトルと 3x3行列の積（移動、回転、拡大・縮小）

$$v_o = v_i \; M$$

$$= (v_i[0], v_i[1], v_i[2])$$

$$\begin{bmatrix} M_{00} & M_{01} & M_{02} \\ M_{10} & M_{11} & M_{12} \\ M_{20} & M_{21} & M_{22} \end{bmatrix}$$

$$v_o[0] = v_i[0] \; M_{00} + v_i[1] \; M_{10} + v_i[2] \; M_{20}$$

$$v_o[1] = v_i[0] \; M_{01} + v_i[1] \; M_{11} + v_i[2] \; M_{21}$$

3次元ベクトルと 3x3行列の積（移動、回転、拡大・縮小）

$$v_o = v_i M$$

$$= (v_i[0], v_i[1], v_i[2]) \begin{bmatrix} M_{00} & M_{01} & M_{02} \\ M_{10} & M_{11} & M_{12} \\ M_{20} & M_{21} & M_{22} \end{bmatrix}$$

$$v_o[0] = v_i[0]\, M_{00} + v_i[1]\, M_{10} + v_i[2]\, M_{20}$$

$$v_o[1] = v_i[0]\, M_{01} + v_i[1]\, M_{11} + v_i[2]\, M_{21}$$

$$v_o[2] = v_i[0]\, M_{02} + v_i[1]\, M_{12} + v_i[2]\, M_{22}$$

このままでは、原点（0, 0, 0）は必ず原点に変換されてしまうという問題が起こります。

そこでCGでは、先ほどの 3x3行列ではなく、4x4行列を良く用います。

$$v_{\mathrm{o}} = v_{\mathrm{i}} \; M$$

$$= \left(v_{\mathrm{i}}[0], v_{\mathrm{i}}[1], v_{\mathrm{i}}[2], v_{\mathrm{i}}[3]\right) \begin{bmatrix} M_{00} & M_{01} & M_{02} & M_{03} \\ M_{10} & M_{11} & M_{12} & M_{13} \\ M_{20} & M_{21} & M_{22} & M_{23} \\ M_{30} & M_{31} & M_{32} & M_{33} \end{bmatrix}$$

そこでCGでは、先ほどの 3x3行列ではなく、4x4行列を良く用います。

$$v_o = v_i M$$

$$= (v_i[0], v_i[1], v_i[2], v_i[3]) \begin{bmatrix} M_{00} & M_{01} & M_{02} & M_{03} \\ M_{10} & M_{11} & M_{12} & M_{13} \\ M_{20} & M_{21} & M_{22} & M_{23} \\ M_{30} & M_{31} & M_{32} & M_{33} \end{bmatrix}$$

$$v_o[0] = v_i[0]\ \boxed{M_{00}} + v_i[1]\ \boxed{M_{10}} + v_i[2]\ \boxed{M_{20}} + v_i[3]\ \boxed{M_{30}}$$

そこでCGでは、先ほどの 3x3行列ではなく、4x4行列を良く用います。

$$v_o = v_i \; M$$

$$= \left(v_i[0], v_i[1], v_i[2], v_i[3]\right) \begin{bmatrix} M_{00} & M_{01} & M_{02} & M_{03} \\ M_{10} & M_{11} & M_{12} & M_{13} \\ M_{20} & M_{21} & M_{22} & M_{23} \\ M_{30} & M_{31} & M_{32} & M_{33} \end{bmatrix}$$

$$v_o[0] = v_i[0] \; \boxed{M_{00}} + v_i[1] \; \boxed{M_{10}} + v_i[2] \; \boxed{M_{20}} + v_i[3] \; \boxed{M_{30}}$$

$$v_o[1] = v_i[0] \; \boxed{M_{01}} + v_i[1] \; \boxed{M_{11}} + v_i[2] \; \boxed{M_{21}} + v_i[3] \; \boxed{M_{31}}$$

そこでCGでは、先ほどの 3x3行列ではなく、4x4行列を良く用います。

$$v_o = v_i \; M$$

$$= \left(v_i[0], v_i[1], v_i[2], v_i[3]\right) \begin{bmatrix} M_{00} & M_{01} & M_{02} & M_{03} \\ M_{10} & M_{11} & M_{12} & M_{13} \\ M_{20} & M_{21} & M_{22} & M_{23} \\ M_{30} & M_{31} & M_{32} & M_{33} \end{bmatrix}$$

$$v_o[0] = v_i[0] \; M_{00} + v_i[1] \; M_{10} + v_i[2] \; M_{20} + v_i[3] \; M_{30}$$

$$v_o[1] = v_i[0] \; M_{01} + v_i[1] \; M_{11} + v_i[2] \; M_{21} + v_i[3] \; M_{31}$$

$$v_o[2] = v_i[0] \; M_{02} + v_i[1] \; M_{12} + v_i[2] \; M_{22} + v_i[3] \; M_{32}$$

そこでCGでは、先ほどの3x3行列ではなく、4x4行列を良く用います。

$$v_o = v_i\ M$$

$$= (v_i[0], v_i[1], v_i[2], v_i[3]) \begin{bmatrix} M_{00} & M_{01} & M_{02} & M_{03} \\ M_{10} & M_{11} & M_{12} & M_{13} \\ M_{20} & M_{21} & M_{22} & M_{23} \\ M_{30} & M_{31} & M_{32} & M_{33} \end{bmatrix}$$

$$v_o[0] = v_i[0]\ M_{00} + v_i[1]\ M_{10} + v_i[2]\ M_{20} + v_i[3]\ M_{30}$$

$$v_o[1] = v_i[0]\ M_{01} + v_i[1]\ M_{11} + v_i[2]\ M_{21} + v_i[3]\ M_{31}$$

$$v_o[2] = v_i[0]\ M_{02} + v_i[1]\ M_{12} + v_i[2]\ M_{22} + v_i[3]\ M_{32}$$

$$v_o[3] = v_i[0]\ M_{03} + v_i[1]\ M_{13} + v_i[2]\ M_{23} + v_i[3]\ M_{33}$$

そこでCGでは、先ほどの3x3行列ではなく、4x4行列を良く用います。

$$v_o = v_i \; M$$

$$= \left(v_i[0], v_i[1], v_i[2], v_i[3]\right) \begin{bmatrix} M_{00} & M_{01} & M_{02} & M_{03} \\ M_{10} & M_{11} & M_{12} & M_{13} \\ M_{20} & M_{21} & M_{22} & M_{23} \\ M_{30} & M_{31} & M_{32} & M_{33} \end{bmatrix}$$

$$v_o[0] = v_i[0] \; M_{00} + v_i[1] \; M_{10} + v_i[2] \; M_{20} + v_i[3] \; M_{30}$$

```
Vo[0] = Vi[0]*M[0][0] + Vi[1]*M[1][0] + Vi[2]*M[2][0] + Vi[3]*M[3][0];
```

$$v_o[1] = v_i[0] \; M_{01} + v_i[1] \; M_{11} + v_i[2] \; M_{21} + v_i[3] \; M_{31}$$

```
Vo[1] = Vi[0]*M[0][1] + Vi[1]*M[1][1] + Vi[2]*M[2][1] + Vi[3]*M[1][1];
```

$$v_o[2] = v_i[0] \; M_{02} + v_i[1] \; M_{12} + v_i[2] \; M_{22} + v_i[3] \; M_{32}$$

```
Vo[2] = Vi[0]*M[0][2] + Vi[1]*M[1][2] + Vi[2]*M[2][2] + Vi[3]*M[3][2];
```

$$v_o[3] = v_i[0] \; M_{03} + v_i[1] \; M_{13} + v_i[2] \; M_{23} + v_i[3] \; M_{33}$$

```
Vo[3] = Vi[0]*M[0][3] + Vi[1]*M[1][3] + Vi[2]*M[2][3] + Vi[3]*M[3][3];
```

この行列演算による移動、回転、拡大・縮小をアフィン変換（*Affine Transformation*）と言います

ちなみに講義では、以下のような4x4行列を用います。

$$v_o = v_i \, M$$

$$= \left(v_i[0], v_i[1], v_i[2], 1 \right) \begin{bmatrix} M_{00} & M_{01} & M_{02} & 0 \\ M_{10} & M_{11} & M_{12} & 0 \\ M_{20} & M_{21} & M_{22} & 0 \\ M_{30} & M_{31} & M_{32} & 1 \end{bmatrix}$$

$$v_o[0] = v_i[0] \, M_{00} + v_i[1] \, M_{10} + v_i[2] \, M_{20} + 1 \, M_{30}$$

```
Vo[0] = Vi[0]*M[0][0] + Vi[1]*M[1][0] + Vi[2]*M[2][0] + Vi[3]*M[3][0];
```

$$v_o[1] = v_i[0] \, M_{01} + v_i[1] \, M_{11} + v_i[2] \, M_{21} + 1 \, M_{31}$$

```
Vo[1] = Vi[0]*M[0][1] + Vi[1]*M[1][1] + Vi[2]*M[2][1] + Vi[3]*M[1][1];
```

$$v_o[2] = v_i[0] \, M_{02} + v_i[1] \, M_{12} + v_i[2] \, M_{22} + 1 \, M_{32}$$

```
Vo[2] = Vi[0]*M[0][2] + Vi[1]*M[1][2] + Vi[2]*M[2][2] + Vi[3]*M[3][2];
```

$$v_o[3] = v_i[0] \; 0 + v_i[1] \; 0 + v_i[2] \; 0 + 1 \; 1$$

```
Vo[3] = Vi[0]*M[0][3] + Vi[1]*M[1][3] + Vi[2]*M[2][3] + Vi[3]*M[3][3];
```

もっと深く勉強してみたい方は、四元数（***Quaternion***：クォータニオン）を調べてみましょう。

単位行列（*Unit Matrix*）：対角成分は１、それ以外は全て０の正方行列

$$M = \begin{bmatrix} M_{00} & M_{01} & M_{02} & M_{03} \\ M_{10} & M_{11} & M_{12} & M_{13} \\ M_{20} & M_{21} & M_{22} & M_{23} \\ M_{30} & M_{31} & M_{32} & M_{33} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

M[0][0]	M[0][1]	M[0][2]	M[0][3]
M[1][0]	M[1][1]	M[1][2]	M[1][3]
M[2][0]	M[2][1]	M[2][2]	M[2][3]
M[3][0]	M[3][1]	M[3][2]	M[3][3]

```
M[0][0]=1; M[0][1]=0; M[0][2]=0; M[0][3]=0;
M[1][0]=0; M[1][1]=1; M[1][2]=0; M[1][3]=0;
M[2][0]=0; M[2][1]=0; M[2][2]=1; M[2][3]=0;
M[3][0]=0; M[3][1]=0; M[3][2]=0; M[3][3]=1;
```

これから、移動、回転、拡大・縮小の行列をそれぞれ設定していきます。

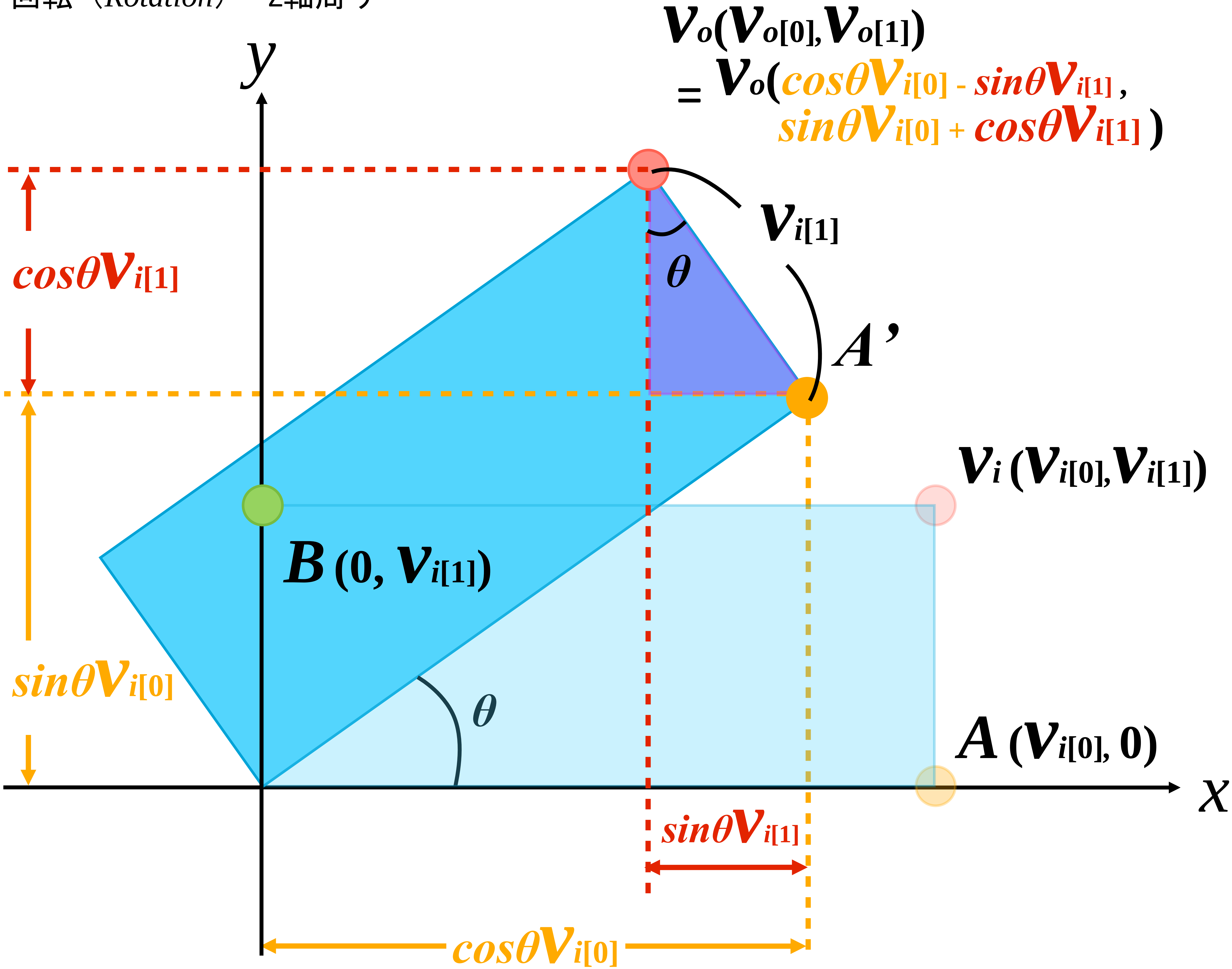
移動（*Translation*）

$$v_o[0] = v_i[0] + x$$
$$v_o[1] = v_i[1] + y$$
$$v_o[2] = v_i[2] + z$$
$$v_o[3] = 1$$

$$(v_o[0], v_o[1], v_o[2], 1) = (v_i[0], v_i[1], v_i[2], 1)\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ x & y & z & 1 \end{bmatrix}$$

```
void matrix_translate(float[][] M, float x, float y, float z) {
  init_matrix(M);
  M[3][0]=x; M[3][1]=y; M[3][2]=z;
}
```

回転（Rotation） - Z軸周り



回転（*Rotation*） - Z軸周り

$$v_o[0] = \cos\theta V_{i[0]} - \sin\theta V_{i[1]}$$

$$v_o[1] = \sin\theta V_{i[0]} + \cos\theta V_{i[1]}$$

$$v_o[2] = v_o[2]$$

$$v_o[3] = 1$$

$$(v_o[0], v_o[1], v_o[2], 1) = (v_i[0], v_i[1], v_i[2], 1)\begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

```
else if( axis=='z'||axis=='Z' ) {
    M[0][0]= cos(theta);
    M[0][1]= sin(theta);
    M[1][0]=-sin(theta);
    M[1][1]= cos(theta);
 }
```

回転（*Rotation*） - Y軸周り

$$v_o[0] = \sin\theta V_{i[2]} + \cos\theta V_{i[0]}$$

$$v_o[1] = v_o[1]$$

$$v_o[2] = \cos\theta V_{i[2]} - \sin\theta V_{i[0]}$$

$$v_o[3] = 1$$

$$(v_o[0], v_o[1], v_o[2], 1) = (v_i[0], v_i[1], v_i[2], 1) \begin{bmatrix} \cos\theta & 0 & -\sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ \sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

```
} else if( axis=='y'||axis=='Y' ) {
  M[2][2]= cos(theta);
  M[2][0]= sin(theta);
  M[0][2]=-sin(theta);
  M[0][0]= cos(theta);
}
```

回転（*Rotation*） - X軸周り

$$v_o[0] = v_o[0]$$

$$v_o[1] = \cos\theta \, v_{i[1]} - \sin\theta \, v_{i[2]}$$

$$v_o[2] = \sin\theta \, v_{i[1]} + \cos\theta \, v_{i[2]}$$

$$v_o[3] = 1$$

$$(v_o[0], v_o[1], v_o[2], 1) = (v_i[0], v_i[1], v_i[2], 1)\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & \sin\theta & 0 \\ 0 & -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

```
if( axis=='x'||axis=='X' ) {
  M[1][1]= cos(theta);
  M[1][2]= sin(theta);
  M[2][1]=-sin(theta);
  M[2][2]= cos(theta);
}
```

拡大・縮小（*Scale*）

$$v_o[0] = a\ v_i[0]$$
$$v_o[1] = b\ v_i[1]$$
$$v_o[2] = c\ v_i[2]$$
$$v_o[3] = 1$$

$$(v_o[0], v_o[1], v_o[2], 1) = (v_i[0], v_i[1], v_i[2], 1)\begin{bmatrix} a & 0 & 0 & 0 \\ 0 & b & 0 & 0 \\ 0 & 0 & c & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

```
// scale
void matrix_scale(float[][] M,float x,float y,float z){
  init_matrix(M);
  M[0][0]=x; M[1][1]=y; M[2][2]=z;
}
```

x方向に2倍、y方向に3倍拡大 $\longrightarrow$ z軸周りに-90度回転 $\longrightarrow$ x方向に5、y方向に5移動

$$(v_o[0], v_o[1], v_o[2], 1) = (v_i[0], v_i[1], v_i[2], 1)\begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 5 & 5 & 0 & 1 \end{bmatrix}$$

$$= (v_i[0], v_i[1], v_i[2], 1) \times \begin{bmatrix} 0 & -2 & 0 & 0 \\ 3 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 5 & 5 & 0 & 1 \end{bmatrix}$$

拡大・縮小、回転、移動行列を全て掛け合わせた行列

$$\begin{bmatrix} a & 0 & 0 & 0 \\ 0 & b & 0 & 0 \\ 0 & 0 & c & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

拡大・縮小行列

$$\begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

回転行列

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ x & y & z & 1 \end{bmatrix}$$

移動行列

移動（*Translation*）

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

単位行列

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ t & 0 & 0 & 1 \end{bmatrix}$$

移動行列

```
// Translate
t+=ts;
if(t>screen_x || t<-screen_x) ts=-ts;

init_matrix(M);
matrix_translate(M, t, 0, 0);

T0.draw(color(255,0,0));
```

回転（*Rotation*）

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

単位行列

$$\begin{bmatrix} \cos(r) & 0 & -\sin(r) & 0 \\ 0 & 1 & 0 & 0 \\ \sin(r) & 0 & \cos(r) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

回転行列（Y軸周り）

```
// Rotate
r+=rs;
if(r>TWO_PI) r=0.0;

init_matrix(M);

matrix_rotate(M, 'Y', r);

T0.draw(color(0,255,0));
```

拡大・縮小（*Scale*）

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

単位行列

$$\begin{bmatrix} s & 0 & 0 & 0 \\ 0 & s & 0 & 0 \\ 0 & 0 & s & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

拡大行列

```
// Scale
s+=ss;
if(s>1 || s<0.5) ss=-ss;

init_matrix(M);

matrix_scale(M, s, s, s);

T0.draw(color(0,0,255));
```