2D Transformations

Translation

Repositioning an object along a straight line path from one coordinate location to another

Add translational distance t_x ,t_y to original coordinate position (x,y) to move the point to a new position (x',y')

$$x' = x + t_x \qquad y' = y + t_y$$

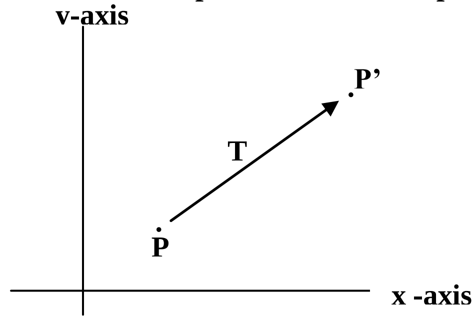
where the pair (t_x , t_y) is called the *translation vector.*

We can write equation as a single matrix equation by using column vectors to represent coordinate points and translation vectors. i.e.

$$P = \begin{bmatrix} x \\ y \end{bmatrix} \qquad T = \begin{bmatrix} t_x \\ t_y \end{bmatrix} \qquad P' = \begin{bmatrix} x' \\ y' \end{bmatrix}$$

so we can write

$$P' = P + T$$



Scaling

This transformation changes the *size* of an object

Such that we can magnify or reduce its size

In case of polygons scaling is done by multiplying coordinate values (x,y) of each vertex by scaling factors s_x s_y to produce the final transformed coordinates (x',y').

s_x scales object in 'x' direction

s_y scales object in 'y' direction

or

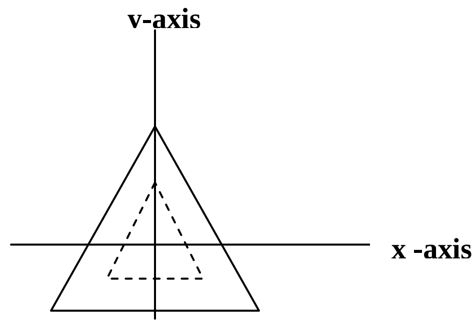
$$P = \begin{bmatrix} x \\ y \end{bmatrix} \qquad T = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \qquad P' = \begin{bmatrix} x' \\ y' \end{bmatrix}$$

or

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} . \begin{bmatrix} x \\ y \end{bmatrix}$$

or

$$P' = S . P$$



Values greater than 1 for s_x s_y produce *enlargement*

Values greater than 1 for s_x s_y *reduce* size of object

s_x = s_x = 1 leaves the size of the object *unchanged*

When s_x, s_y are assigned the same value s_x = s_x = 3 or 4 etc then a *Uniform Scaling* is produced.

Rotation

Repositioning an object along a circular path in x,y plane

Specify rotation angle 'θ' and position (x_r,y_r) of rotation point about which the object is to be rotated

+ value for 'θ' define *counter-clockwise* rotation about a point

- value for 'θ' define *clockwise* rotation about a point

If (x,y) is the original point 'r' the constant distance from origin, 'Φ' the original angular displacement from x-axis.

Now the point (x,y) is rotated through angle 'θ' in a counter clock wise direction

Express the transformed coordinates in terms of 'Φ' and 'θ' as

$$x' = r\cos(\Phi + \theta) = r\cos\Phi.\cos\theta - r\sin\Phi.\sin\theta \quad ....(i)$$

$$y' = r\sin(\Phi + \theta) = r\cos\Phi.\sin\theta + r\sin\Phi.\cos\theta \quad ....(ii)$$

We know that original coordinates of point in polar coordinates are

$$x = r\cos\Phi$$

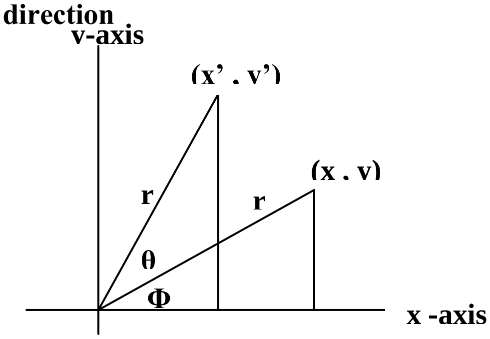
$$y = r\sin\Phi$$

substituting these values in (i) and (ii)

we get,

$$x' = x\cos\theta - y\sin\theta$$

$$y' = x\sin\theta + y\cos\theta$$



so using column vector representation for coordinate points the matrix form would be

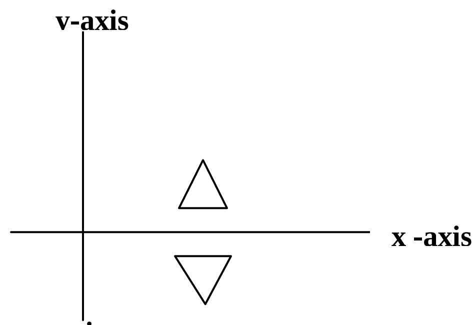
$$P' = R . P$$

where the rotation matrix is

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

Reflection

Generates a mirror image of the original object



(i).Reflection about x axis or about line y = 0

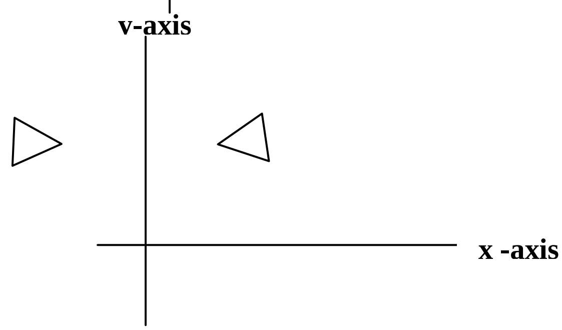
Keeps 'x' value same but flips y value of coordinate points

So $x' = x$

$y' = -y$

i.e.

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$



(ii).Reflection about y axis or about line x = 0

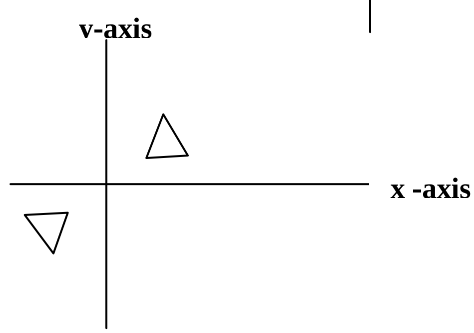
Keeps 'y' value same but flips x value of coordinate points

So $x' = -x$

$y' = y$

i.e.

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$



(iii).Reflection about origin

Flip both 'x' and 'y' coordinates of a point

So $x' = -x$

$y' = -y$

i.e.

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

(iv).Reflection about line y = x

Steps required:

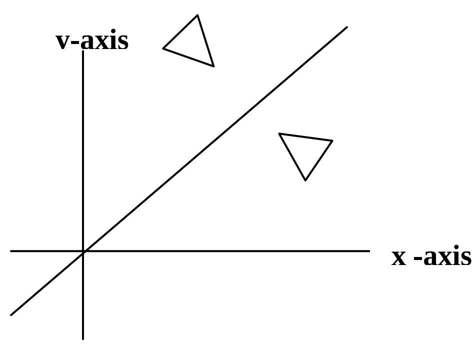
i. Rotate about origin in clockwise direction by 45 degree which rotates line y = x to x-axis

ii. Take reflection against x-axis

iii. Rotate in anti-clockwise direction by same angle

here,

$$R = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \qquad R' = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$



For reflection against x-axis,

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

so for $\theta = 45$ we get

$$R_{(y=x)} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

(v).Reflection about line y = -x

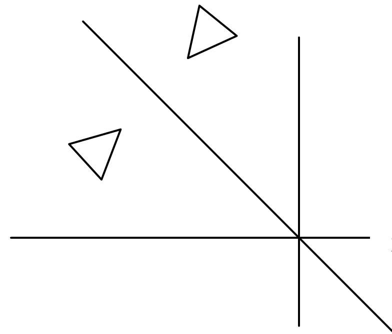
Steps required:

i. Rotate about origin in clockwise direction by 45

v-axis

ii. Take reflection against y-axis
iii. Rotate in anti-clockwise direction by same angle
here,

$$R = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \qquad R' = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$



For reflection against x-axis,

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

so for $\theta = 45$ we get

$$R_{(y=-x)} = \begin{bmatrix} 0 & -1 \\ -1 & 0 \end{bmatrix}$$

Shearing

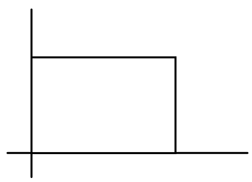
Distorts **the shape of object in either ‘x’ or ‘y’ or both direction**
In case of single directional shearing (e.g. in ‘x’ direction can be viewed as an object made up of very thin layer and slid over each other with the *base* remaining where it is).

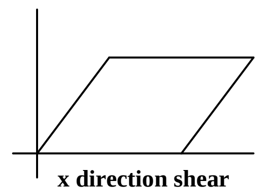
in ‘x’ direction,

$x' = x + s_{hx} \cdot y$

$y' = y$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & s_{hx} \\ 0 & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \end{bmatrix}$$





x direction shear

in ‘y’ direction,

$x' = x$

$y' = y + s_{hy} \cdot x$

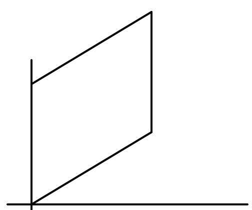
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ s_{hy} & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \end{bmatrix}$$

in both directions,

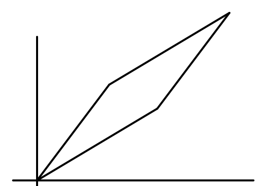
$x' = x + s_{hx} \cdot y$

$y' = y + s_{hy} \cdot x$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & s_{hx} \\ s_{hy} & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \end{bmatrix}$$



y direction shear



bi direction shear

Homogenous Coordinates

The matrix representations for translation, scaling and rotation are respectively:

$P' = T + P$

$P' = S \cdot P$

$P' = R \cdot P$

Translation is treated differently (as addition) from scaling and rotation (as multiplication)
We need to treat all three transformations in a consistent way so they can be combined easily.
Graphics applications involve sequence of geometric transformations. e.g. animation may require an object to be translated , rotated etc in a sequence.
For an animation that requires scaling, rotation, translation, instead of applying each transformation one at a time separately we can combine them so that final coordinates are obtained directly from initial coordinates thus eliminating intermediate steps.
Here, incase of homogenous coordinates we add a third coordinate ‘*h*’ to a point (x,y) so that each point is represented by (*h*x, *h*y, *h*).
‘*h*’ is normally set to 1.
If points are expressed in homogenous coordinates, all geometrical transformation equations can be represented as matrix multiplications.

Coordinates of a point are represented as three element column vectors, transformation operations are written as 3 x 3 matrices.
So, for translation we have

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

or, P' = T(t_x , t_y) . P
With T(t_x , t_y) as translation matrix, inverse of this translation matrix is obtained by representing t_x, t_y with $-t_x$, $-t_y$.
Similarly, the rotation transformation equation about the coordinate origin are

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

or, P' = R(θ) . P
Here, rotation operator R(θ) is a 3 x 3 matrix and inverse rotation is obtained with –θ.
Similarly, scaling relative to coordinate origin is

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} s_{hx} & 0 & 0 \\ 0 & s_{hy} & 0 \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

or, P' = S(s_{hx} , s_{hy}) . P
Inverse scaling matrix is obtained with $1 / s_{hx}$ and $1 / s_{hx}$.

Composite Transformation

With the matrix representation of transformation equations it is possible to setup a matrix for any sequence of transformations as a composite transformation matrix by calculating the matrix product of individual transformation.
For column matrix representation of coordinate positions we form composite transformation by multiplying matrices in order from right to left.

i. Two Successive Translation are Additive

Let two successive translation vectors (t_{x1} , t_{y1}) and (t_{x2} , t_{y2}) are applied to a coordinate position P then
or, P' = T(t_{x2} , t_{y2}) . {T(t_{x1} , t_{y1}) . P} and P' = {T(t_{x2} , t_{y2}) . T(t_{x1} , t_{y1}) } . P
Here the composite transformation matrix for this sequence of translation is

$$\begin{bmatrix} 1 & 0 & t_{x2} \\ 0 & 1 & t_{y2} \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} 1 & 0 & t_{x1} \\ 0 & 1 & t_{y1} \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_{x1} + t_{x2} \\ 0 & 1 & t_{y1} + t_{y2} \\ 0 & 0 & 1 \end{bmatrix}$$

or, $T(t_{x2} , t_{y2}) \; . \; T(t_{x1} , t_{y1}) = T(t_{x1} + t_{x2} , t_{y1} + t_{y2})$

ii. Two successive Scaling operations are Multiplicative

Let (s_{x1} , s_{y1}) and (s_{x2} , s_{y2}) be two successive vectors applied to a coordinate position P then the composite scaling matrix thus produced is
or, P' = S(s_{x2} , s_{y2}) . {S(s_{x1} , s_{y1}) . P} and P' = {S(s_{x2} , s_{y2}) . S(s_{x1} , s_{y1}) } . P
Here the composite transformation matrix for this sequence of translation is

$$\begin{bmatrix} s_{x2} & 0 & 0 \\ 0 & s_{y2} & 0 \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} s_{x1} & 0 & 0 \\ 0 & s_{y1} & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} s_{x2} . s_{x1} & 0 & 0 \\ 0 & s_{y2} . s_{y1} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

or, $S(s_{x2} , s_{y2}) \; . \; S(s_{x1} , s_{y1}) = S(s_{x1} . s_{x2} , s_{y1} . s_{y2})$

iii. Two successive Rotation operations are Additive

Let R(θ_1) and R(θ_2) be two successive rotations applied to a coordinate position P then the composite scaling matrix thus produced is

or, $P' = R(\theta_2) \cdot \{R(\theta_1) \cdot P\}$ and $P' = \{R(\theta_2) \cdot R(\theta_1)\} \cdot P$

Here the composite transformation matrix for this sequence of translation is

or,

$$\begin{bmatrix} \cos\theta_2 & -\sin\theta_2 \\ \sin\theta_2 & \cos\theta_2 \end{bmatrix} \cdot \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 \\ \sin\theta_1 & \cos\theta_1 \end{bmatrix}$$

or,

$$\begin{bmatrix} \cos\theta_2 \cdot \cos\theta_1 - \sin\theta_2 \cdot \sin\theta_1 \\ \sin\theta_2 \cdot \cos\theta_1 + \sin\theta_1 \cdot \cos\theta_2 \end{bmatrix}$$

or,

$$\begin{bmatrix} \cos(\theta_1 + \theta_2) & -\sin(\theta_1 + \theta_2) \\ \sin(\theta_1 + \theta_2) & \cos(\theta_1 + \theta_2) \end{bmatrix}$$

or, $R(\theta_2) \cdot R(\theta_1) = R(\theta_1 + \theta_2)$

Affine Transformation

Linear 2D geometric transformation which maps variables (e.g. pixel intensity values located at point (x_1,y_1) in an input image) into new variables (e.g. (x_2y_2) in an output image) by applying a linear combination of translation, rotation , scaling and/or shearing operations.

****NOTE:**

- **See class notes for pivot and fixed point transformations**
- **See class notes for reflection about arbitrary line (y = mx + c)**