**Two Dimension Transformations**

One of the most common and important tasks in computer graphics is to transform the coordinates ( position, orientation, size and shape ) of either objects within the graphical scene or the camera that is viewing the scene. It is also frequently necessary to transform coordinates from one coordinate system to another, (e.g. world coordinates to viewpoint coordinates to screen coordinates). All of these transformations can be efficiently and sufficiently handled using some simple matrix representations, which we will see can be particularly useful for combining multiple transformations into a single composite transform matrix.

The basic types of transformations that can perform in two dimensions:

- Translations

- Scaling

- Rotation

These basic transformations can also be combined to obtain more complex transformations.

**Representation of Points/Objects**

A point p in 2D is represented as a pair of numbers: p= (x, y) where x is the x-coordinate of the point p and y is the y-coordinate of p . 2D objects are often represented as a set of points (vertices), {𝑝1, 𝑝2,..., 𝑝𝑛} . We can also write points in vector/matrix as : p = (xy)

 **Translation:** A translation moves an object to a different position on the screen. You can translate a point in 2D by adding translation coordinate (𝑡𝑥 , 𝑡𝑦 ) to the original coordinate (X, Y) to get the new coordinate (X’, Y’).



We can represent the translation transformation using column vectors as follows:

P’ = P+T, 𝑃 = , 𝑇 = , 𝑃′ = =

Example : Consider a triangle defined by its three vertices (20,0), ( 60,0), (40,100) been moved 100 units to the right and 10 units up. Find the new coordinates of this triangle after translation.

𝑇 = [100 10 ] , 𝑃 = [ 20 60 40 0 0 100], 𝑃′ = [ 20 + 𝑡𝑥 60 + 𝑡𝑥 40+𝑡𝑥 0 + 𝑡𝑦 0 + 𝑡𝑦 100 + 𝑡𝑦 ]

So, the new coordinated of the triangle are : 𝑃′ = [120 160 140 10 10 110]