الجامعة المستنصرية

كلية العلوم

قسم: علوم الحاسوب

عنوان التقرير

Two dimension Geometric transformation
(Translation)

أسم الطالب الرباعي: يوسف عبد الستار شيخان احمد

التـــــوقيع:

القسم: علوم الحاسوب **المـــرحلة: الثالثة**

الشعبـــــة : A1 **الفـــــــرع: CS**

البريد الالكتروني: yousufabdalstar1@gmail.com

التقرير مقدم من ضمن متطلبات درجة الامتحان النهائي

لمادة رسم بالحاسوب

يملئ من قبل أستاذ المادة					
اسم الأستاذ:					
الدرجة					
من					

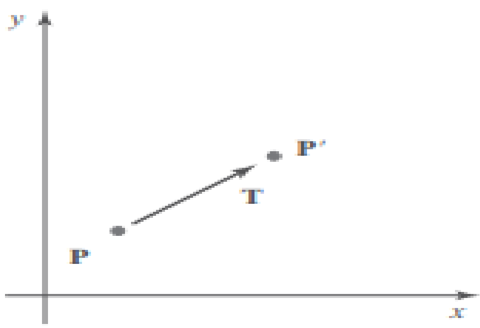
ختم اللجنة الامتحانية توقيع المدقق

Introduction

Two-Dimensional Translation

We perform a **translation** on a single coordinate point by adding offsets to its coordinates so as to generate a new coordinate position. In effect, we are moving the original point position along a straight-line path to its new location. Similarly, a translation is applied to an object that is defined with multiple coordinate positions, such as a quadrilateral, by relocating all the coordinate positions by the same displacement along parallel paths. Then the complete object is displayed at the new location.

To translate a two-dimensional position, we add translation distances tx and ty to the original coordinates (x, y) to obtain the new coordinate position (x ′ , y ′) as shown in Figure 1



[Figure 1] Translating a point from position P to position P ′ using a translation vector T.

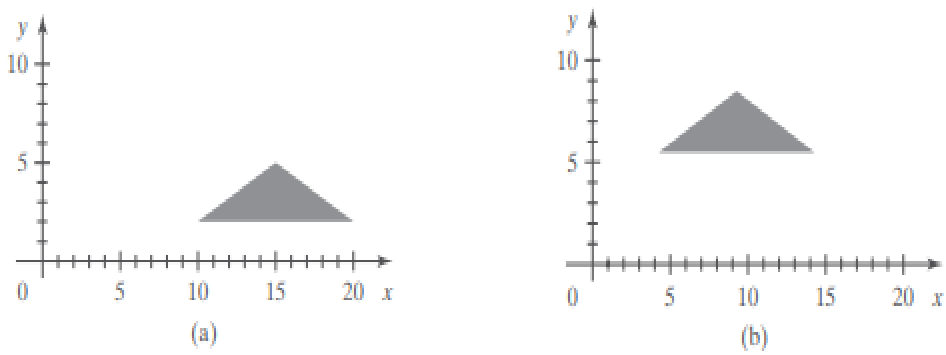
$$x' = x + tx\ ,\ y' = y + ty \tag{1}$$

The translation distance pair (tx, ty) is called a translation vector or shift vector. We can express Equations 1 as a single matrix equation by using the following column vectors to represent coordinate positions and the translation vector:

$$p = \binom{x}{y} \quad , \quad p' = \binom{x\prime}{y\prime} \quad , \quad t = \binom{tx}{ty} \tag{2}$$

This allows us to write the two-dimensional translation equations in the matrix form

$$P' = P + T \qquad (3)$$



[Figure 2]

Moving a polygon from position (a) to position (b) with the translation vector (−5.50, 3.75).

Translation is a rigid-body transformation that moves objects without deformation. That is, every point on the object is translated by the same amount. A straight-line segment is translated by applying Equation 3 to each of the two line endpoints and redrawing the line between the new endpoint positions. A polygon is translated similarly. We add a translation vector to the coordinate position of each vertex and then regenerate the polygon using the new set of vertex coordinates. **Figure 2** illustrates the application of a specified translation vector to move an object from one position to another. The following routine illustrates the translation operations. An input translation vector is used to move the n vertices of a polygon from one world-coordinate position to another, and OpenGL routines are used to regenerate the translated polygon.

Two-Dimensional Translation Matrix

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

$$\mathbf{P}' = \mathbf{T}(t_x, t_y) \cdot \mathbf{P}$$

Inverse translation matrix ν

Translate in the opposite direction

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

- If two successive translation vectors (t 1 x , t 1 y) and (t 2 x , t 2 y ( ) are 1 x , 1 y) (2 x , 2 y) applied to a 2-D coordinate position P, the final transformed location P’ is

$$\begin{aligned} \mathbf{P}' &= \mathbf{T}(t_{2x}, t_{2y}) \cdot \{\mathbf{T}(t_{1x}, t_{1y}) \cdot \mathbf{P}\} \\ &= \{\mathbf{T}(t_{2x}, t_{2y}) \cdot \mathbf{T}(t_{1x}, t_{1y})\} \cdot \mathbf{P} \end{aligned}$$

- The composite transformation matrix for this sequence of translations is

$$\begin{bmatrix} 1 & 0 & t_{2x} \\ 0 & 1 & t_{2y} \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & t_{1x} \\ 0 & 1 & t_{1y} \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_{1x} + t_{2x} \\ 0 & 1 & t_{1y} + t_{2y} \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{T}(t_{2x}, t_{2y}) \cdot \mathbf{T}(t_{1x}, t_{1y}) = \mathbf{T}(t_{1x} + t_{2x}, t_{1y} + t_{2y})$$

Example, to translate a triangle with vertices at original coordinates (10,20), (10,10), (20,10) by tx=5, ty=10, we compute as followings:

Translation of vertex (10,20):

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 5 \\ 0 & 1 & 10 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 10 \\ 20 \\ 1 \end{bmatrix} = \begin{bmatrix} 1*10+0*20+5*1 \\ 0*10+1*20+10*1 \\ 0*10+0*20+1*1 \end{bmatrix} = \begin{bmatrix} 15 \\ 30 \\ 1 \end{bmatrix}$$

Translation of vertex (10,10):

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 5 \\ 0 & 1 & 10 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 10 \\ 10 \\ 1 \end{bmatrix} = \begin{bmatrix} 1*10+0*10+5*1 \\ 0*10+1*10+10*1 \\ 0*10+0*10+1*1 \end{bmatrix} = \begin{bmatrix} 15 \\ 20 \\ 1 \end{bmatrix}$$

Translation of vertex (20,10):

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 5 \\ 0 & 1 & 10 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 20 \\ 10 \\ 1 \end{bmatrix} = \begin{bmatrix} 1*20+0*10+5*1 \\ 0*20+1*10+10*1 \\ 0*20+0*10+1*1 \end{bmatrix} = \begin{bmatrix} 25 \\ 20 \\ 1 \end{bmatrix}$$

The resultant coordinates of the triangle vertices are (15,30), (15,20), and (25,20) respectively.

اسم الطالب : يوسف عبد الستار شيخان احمد
المرحلة : الثالثة الشعبة: A1
الفرع : CS
المادة : رسم بالحاسوب

Reference

1. https://www.gatevidyalay.com/2d-transformation-in-computer-graphics-translation-examples/
2. https://www.cs.rit.edu/~icss571/clipTrans/2DTransBack.html
3. https://www.tutorialspoint.com/computer_graphics/2d_transformation.htm
4. https://www.javatpoint.com/computer-graphics-introduction-of-transformations
5. http://www.it.hiof.no/~borres/j3d/math/twod/p-twod.html