

1. Introduction

Vector 3D model representation in computer graphics requires some simple maths transformations to adapt model dimensions to the *graphics* object where the model will be drawn. In the following a vb.net environment is considered, and therefore a *graphics* object is an object capable to contain graphic objects such as a form or a panel control<sup>1</sup>.

In figure 1-a) the typical coordinate system for a computer device is shown, while the coordinate system for a generic model is shown in figure 1-b).

The dimensions of the graphical window are usually measured in pixels and are indicated as **Graphic.Width** and **Graphic.Height**.

The dimensions of the model is any, and it naturally can be in any location with respect of the model coordinate system. The third dimension will be treated further on, after dealing with the main mathematical transformations.

It is useful to highlight that if a printer object is considered instead of a *graphics* object, nothing changes.

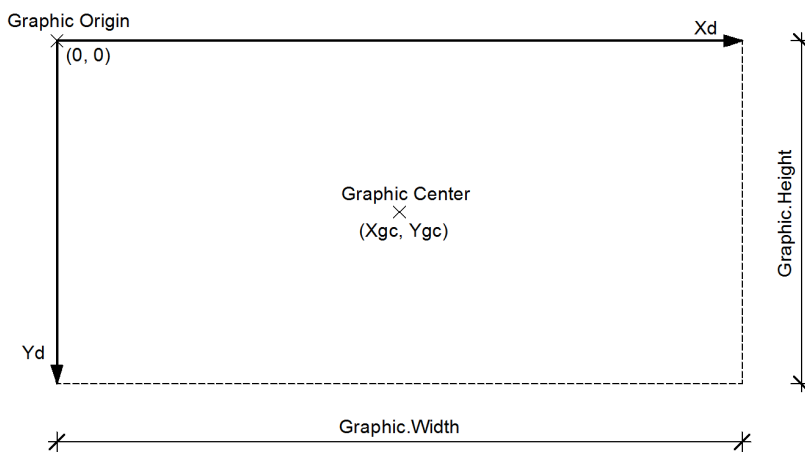


Figure 1-a)

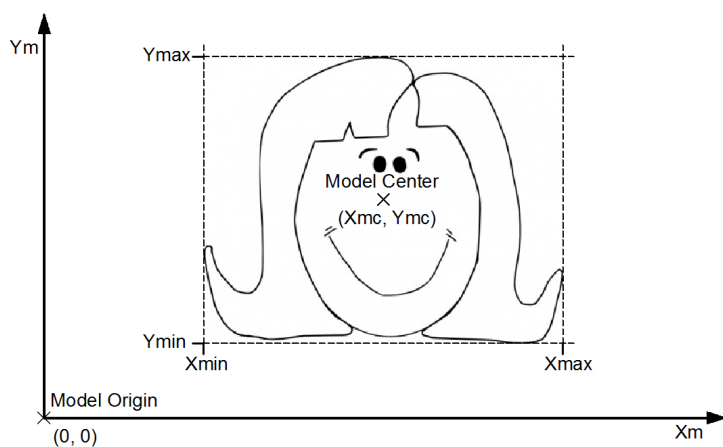


Figure 1-b)

<sup>1</sup> To obtain the best graphical management this object needs to recognize the mouse wheel event.

## 2. Main mathematical transformations

### 2.1 Zoom extents

This is the main task of the model representation. The model is drawn with the maximum possible size in the graphic window, as shown in figure 2. The axes are named Xd, Yd meaning x, y coordinates of the drawing.

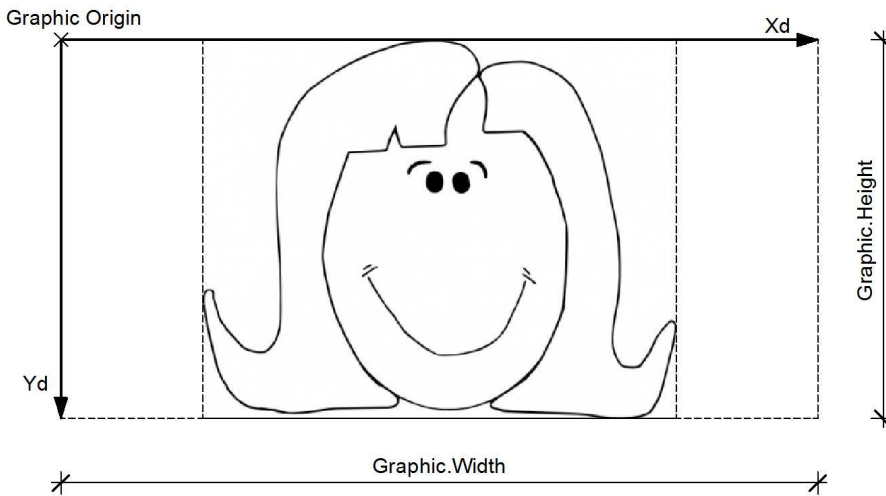


Figure 2

This requires a scale transformation to fill the graphic area, a shift transformation to move the model center ( $X_{mc}$ ,  $Y_{mc}$ ) to the graphic center ( $X_{gc}$ ,  $Y_{gc}$ ), and finally a mirror transformation to accomplish with the opposite orientation of the y axis.

All the three above operations can be written as:

$$X_{draw} = X_{gc} - \frac{X_{mc} - x_i}{GrRatio}; \quad Y_{draw} = Y_{gc} + \frac{Y_{mc} - y_i}{GrRatio} \quad (1)$$

where:

- $x_i, y_i$  are the coordinates of a point to be drawn
- $X_{gc}, Y_{gc}$  are the coordinates of the graphic center, as shown in figures 1-a)
- $X_{mc}, Y_{mc}$  are the coordinates of the model center, as shown in figures 1-b)
- $X_{draw}, Y_{draw}$  are the drawing coordinates of the model, transformed from their original reference system to the drawing window reference system

- $GrRatio = \max\left(\frac{X_{max} - X_{min}}{Graphic.Width}, \frac{Y_{max} - Y_{min}}{Graphic.Height}\right)$  is the ratio of the model size to the graphic window

## 2.2 Zoom window

The second main task of the model representation is to draw a zoom of the model contained in a selection rectangle, as shown in figure 3-a). The result of this is represented in figure 3-b).

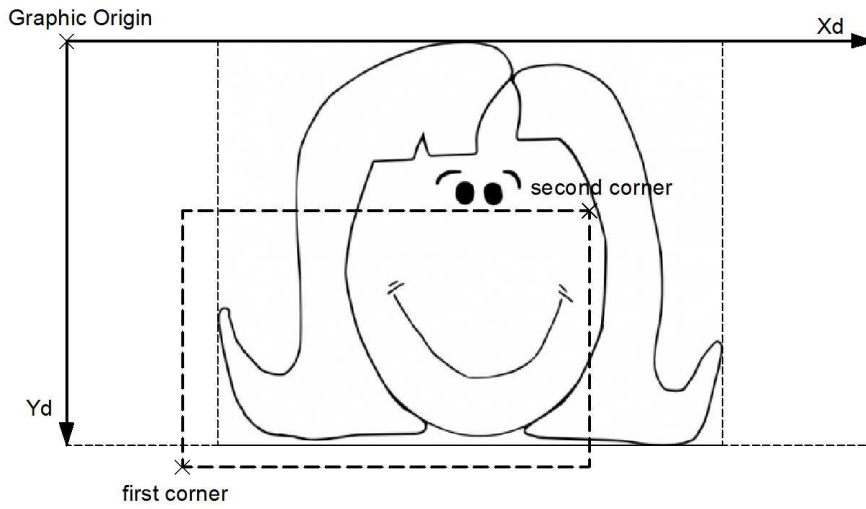


Figure 3-a)

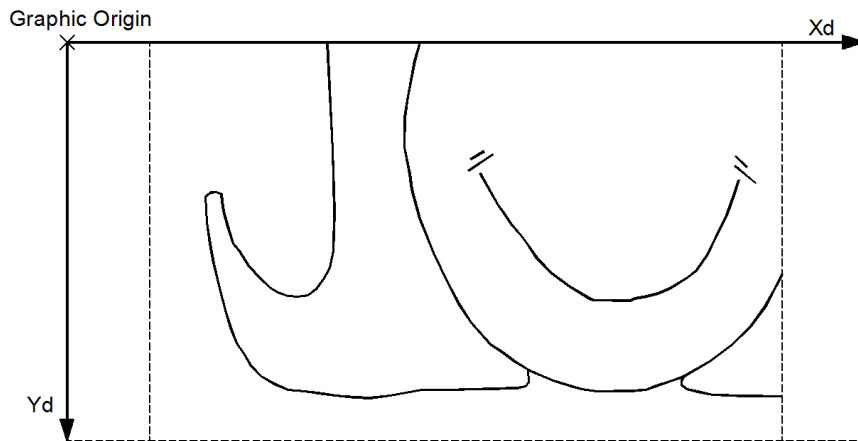


Figure 3-b)

To obtain this, two simple operations are needed: a check for the lines contained in the zoom rectangle, and a zoom extents operation as described in the previous paragraph.

The check of the lines contained in the zoom rectangle will calculate new  $X_{min}$ ,  $X_{max}$ ,  $Y_{min}$  and  $Y_{max}$  coordinates to introduce in equation (1).

### 2.3 Panning

In addition to the main operations described above, it could be useful to mention the pan transformation that means a repositioning of the view in the drawing area. It can be accomplished simply by changing the  $X_{gc}$ ,  $Y_{gc}$  coordinates of the graphic window.

For example the pan represented in figure 4, from the lower-left position to the higher-right position, is obtained with the following transformation:

$$X_{gc} = X_{gc} + \Delta x; \quad Y_{gc} = Y_{gc} - \Delta y$$

where  $\Delta x$  and  $\Delta y$  are positive values in this case.

The drawing coordinates are always obtained using equations (1).

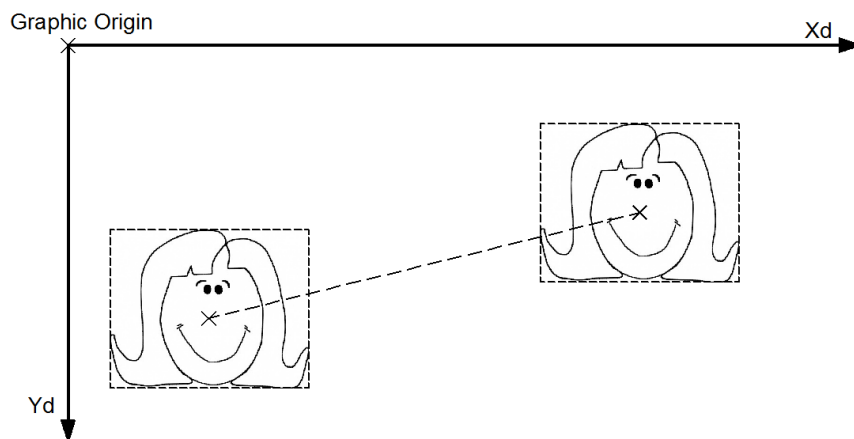


Figure 4

## 2.4 Zoom with the mouse wheel

The use of the mouse wheel in computer graphics is usually associated to the operations of zoom and pan at the same time. In the following figure 5 a positive zoom (magnification) is represented, together with a shift of the model from the original position to another.

For a positive click of the mouse wheel, with reference to figure 5, the equations that describe the transformation are:

$$X_{cg} = X_{cg} - wp \Delta x; \quad Y_{cg} = Y_{cg} - wp \Delta y \quad \text{turn away the model}$$

$$GrRatio = GrRatio / wp \quad \text{magnifies the model}$$

where:

- $wp$  is the “wheel power”, a positive constant that mediates the effect of a click of the wheel. The value could be 0.02 or 0.03 for a “power” of 20% or 30%.
- $\Delta x, \Delta y$  are the distances from the mouse position and the graphic center (both negative in the case of figure 5)

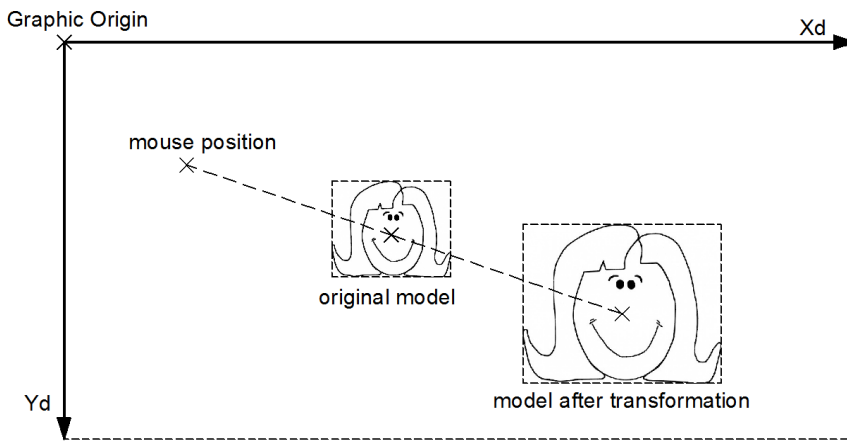


Figure 5

For a negative click of the mouse wheel, that could reverse the transformation shown in figure 5, the equations are:

$$X_{cg} = X_{cg} + wp \Delta x; \quad Y_{cg} = Y_{cg} + wp \Delta y \quad \text{bring near the model}$$

$$GrRatio = GrRatio wp \quad \text{reduces the model}$$

### 3. 3D representation

There are many articles available on the subject, nonetheless I believe that a simple description, along with some practical examples, could be useful.

The only representation considered in this article is the axonometric orthographic projection of a 3D model on a 2D plane. In this kind of transformation the real model is projected on a plane by means of parallel rays sent out from a point of view. Perspective effects are neglected and dimensions are not altered; parallelism of lines are preserved but angles are not preserved. It's the way engineering calculation programs act.

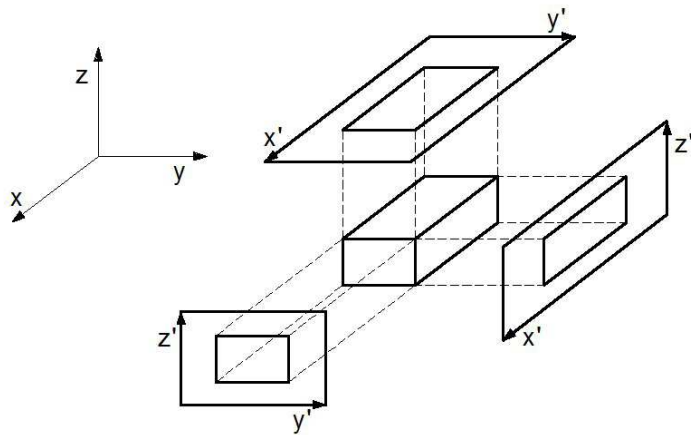


Figure 6

Figure 6 shows a parallelepiped projected from on three point of view:

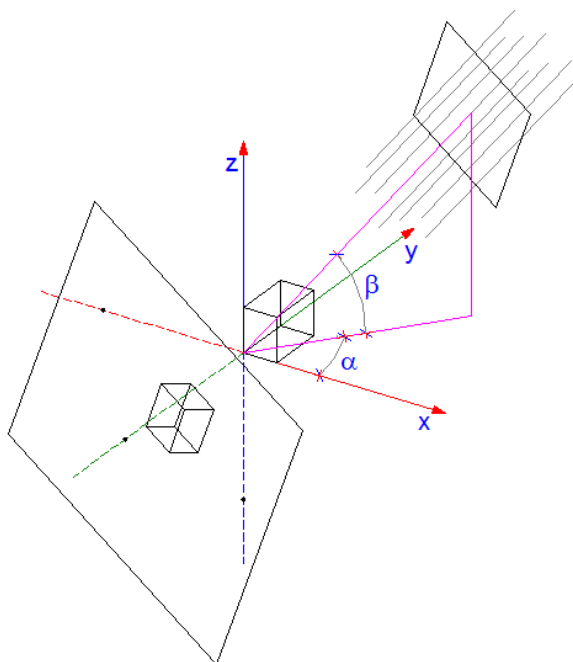
- with the point of view parallel to x axis the result is a projection on a plane parallel to (y, z) plane;
- with the point of view parallel to y axis the result is a projection on a plane parallel to (z, x) plane;
- with the point of view parallel to z axis the result is a projection on a plane parallel to (x, y) plane.

Figure 7 shows a projection obtained from a generic point of view. The point of view is defined by means of two angles: the first angle, named  $\alpha$ , defines the position of the projection of the point of view on the (x, y) plane; the second angle, named  $\beta$ , defines the elevation of the point of view with respect to the (x, y) plane.

With these assumptions, the coordinates  $(x_i, y_i, z_i)$  of a generic point in a 3D space are projected on a plane in the point  $(xp_i, yp_i, zp_i)$  with the following equations:

$$xp_i = -x_i \cdot \text{sen } \alpha + y_i \cdot \text{cos } \alpha$$

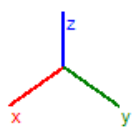
$$yp_i = (-x_i \cdot \text{cos } \alpha - y_i \cdot \text{sen } \alpha) \cdot \text{sen } \beta + z_i \cdot \text{cos } \beta$$



Figures 8, 9 and 10 show some examples of the projection of a cube from three different points of view.

It's interesting to notice how the global coordinate system appears from the various points of view.

Figure 7



$\alpha$   
45

$\beta$   
45

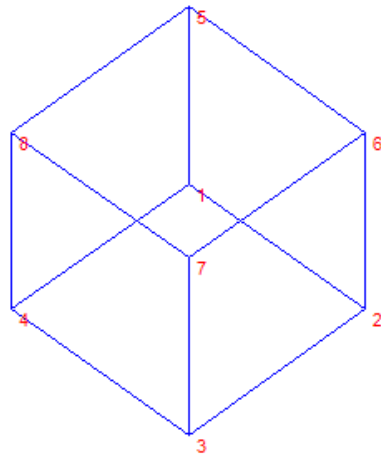


Figure 8



$\alpha$   
20

$\beta$   
20

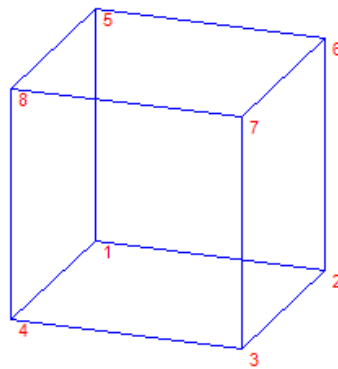


Figure 9



$\alpha$   
150

$\beta$   
20

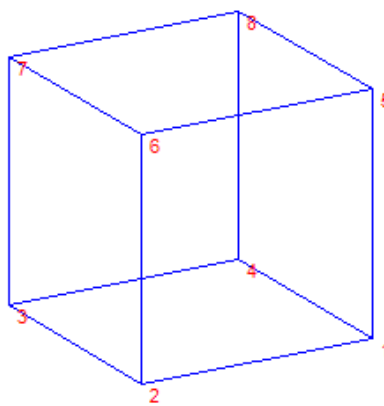


Figure 10