36

3D Graphics

In graphics, we use so many techniques to represent 3D images on a computer screen, which is supposed to be a 2D plane. One of such techniques is called as "*depth cueing*" and we used this technique in "**VB Controls**". Another well-known technique is "*perspective projection*". This technique is widely used in 3D games and many other 3D applications. In this chapter, let's see perspective projection!

36.1 Perspective Projection

The idea of perspective projection is that we have to convert a point in 3D plane to 2D plane. That is, if we have a point A (x, y, z), we have to represent this point as A´ (x´, y´) omitting Z coordinate. To do this, we have to use the formula

$$X´ = \frac{X * distance}{Z + distance}$$

$$Y´ = \frac{Y * distance}{Z + distance}$$

These equations may look easy. But these equations are not even available in so called gem-books for graphics.

36.2 3D Rectangle

Here I present you a small program that plots a 3D Rectangle in 2D plane.

```
#include <graphics.h>

#define  distance (20)  /* your choice */

typedef struct
    {
       int x, y;
    } COORD_2D;



typedef struct
```

```
    {
        int x, y, z;
    } COORD_3D;

void Draw2DRectangle( COORD_2D *pts )
{
    int i;
    for( i=0 ; i<4-1 ; ++i )
        line( pts[i].x, pts[i].y, pts[i+1].x, pts[i+1].y );
    line( pts[0].x, pts[0].y, pts[3].x, pts[3].y );
} /*--Draw2DRectangle( )----------*/

/* converts given 3D coordinates to 2D coordinates */
void Perspective3Dto2D( COORD_2D *pts2d, COORD_3D *pts3d, int n )
{
    int i;
    for ( i=0; i<n ; ++i )
     {
      pts2d[i].x = (pts3d[i].x*distance)  / (pts3d[i].z + distance);
      pts2d[i].y = (pts3d[i].y*distance)  / (pts3d[i].z + distance);
     }
} /*--Perspective3Dto2D( )---------*/

int main( void )
{
    int gdriver = VGA, gmode = VGAHI;
    COORD_3D pts3d[4];
    COORD_2D pts2d[4];
    initgraph( &gdriver, &gmode, "d:\\tc\\bgi" );
    /* Our 3D rectangle's coordinates */
    pts3d[0].x = 200; pts3d[0].y = 220; pts3d[0].z = 15;
    pts3d[1].x = 500; pts3d[1].y = 220; pts3d[1].z = 5;
    pts3d[2].x = 500; pts3d[2].y = 450; pts3d[2].z = 5;
    pts3d[3].x = 200; pts3d[3].y = 450; pts3d[3].z = 15;
    Perspective3Dto2D( pts2d, pts3d, 4 );
    Draw2DRectangle( pts2d );
    getch( );
    closegraph( );
    return(0);
} /*--main( )-----------*/
```

Suggested Projects

1. Develop a CAD software.
2. Write a software that implements wire frame model.