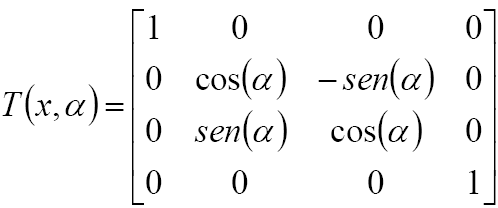
Hola Manolo.

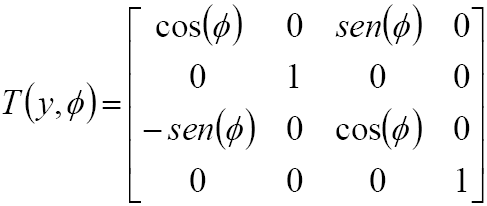
En el ejemplo del cubo se crea con líneas el mismo, y en cada extremo de la misma se le aplica las matrices de transformación para la rotación, y la traslación. Esto viene en la línea

RotaCubo(Cubo,trotxh(r\*180/pi)\*trotyh(r1\*180/pi)\*trotzh(r2\*180/pi));

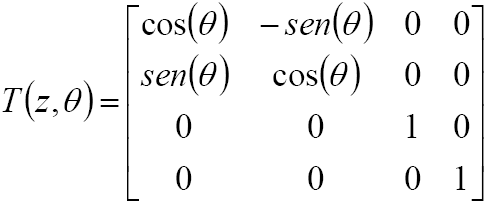
* Cubo es el objeto que define con líneas al cubo
* trotxh(r\*180/pi) es la matriz de rotación en X que genera una rotación del ángulo especificado



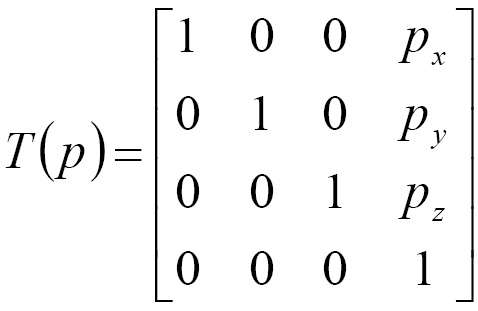
* trotyh(r\*180/pi) es la matriz de rotación en Y que genera una rotación del ángulo especificado



* trotzh(r2\*180/pi)) es la matriz de rotación en Z que genera una rotación del ángulo especificado



* translh(Pxyz) es la matriz de traslación en el vector Pxyz=[px,py,pz,1]’, donde px es la coordenada x, py la coordenada en y, pz la coordenada en z y 1 lo lleva para mantener el formato de las matrices de transformación homogénea.



Entonces cuando se aplica esto a un sistema en todos sus puntos se puede girar o mover. Este principio lo usan en los videos juegos para crear los ambientes desde el punto de vista del jugador.

A las funciones les puse la terminación h, porque Matlab tiene sus propias funciones de transformación homogénea y así el programa use el que yo necesito y que van con los demás archivos.