

Maze Visibility Document

Target

在這個 Project 中為了將原本以 Vertex3f 表達的座標點轉為以 Vertex2f 的平面座標，其中必須實作，gluPerspective、gluLookAt、Clipping 和 Depth Test

Perspective

這一個矩陣是將視野內的物體產生如同人類視野一般的效果，近大遠小的效果，並且將算出來的 4x4 矩陣存入 FuncMatrix 的 projection 中

$$P = \begin{bmatrix} \frac{2 * near}{right - left} & 0 & \frac{right + left}{right - left} & 0 \\ 0 & \frac{2 * near}{top - bottom} & \frac{top + bottom}{top - bottom} & 0 \\ 0 & 0 & -\frac{far + near}{far - near} & -\frac{2 * far * near}{far - near} \\ 0 & 0 & -1 & 0 \end{bmatrix}$$

Model View

這一個矩陣是將世界座標的物體轉進視野座標內，分別由一個旋轉矩陣和一的轉移矩陣相乘而成，並且將算出來的 4x4 矩陣存入 FuncMatrix 的 modelView 中

Clipping & Depth Test

簡而言之就是將視野外的物體忽略不畫，以節省運算效率。

實作步驟如下：

Step1:

將視野分為左視野和右視野，分別由Viewer的
所在位置向外射出的向量來表示

Step2:

將視野資訊傳入

`Is_Clip(Direction direction[2], Direction view_direction)`

這個 function 內的 direction 分別存入了一面牆的
start和end，這兩點相對於視野視線的位置(左、右、
交於上)，view_direction則代表了這是一個左視線還是
右視線，若此牆在視野內將回傳True，反之則否，並且
分為兩種情況：

1. start和end兩點都在左視線(右視線)的右側(左側)，
則回傳True
2. 兩點分別在一條視線的不同側，則回傳True

Step3

opaque

T

若此牆壁是需要顯示的，並且兩射線的 Is_Clip 值皆為 True，則畫此牆，並使用 Cross_Param 來計算視野射線和此牆的交點在此牆上的百分比數，將此牆切至視野交點大小

F

若此牆壁是需要不需要顯示的，並且兩射線的 Is_Clip 值皆為 True，則畫此牆，並使用 Cross_Param 來計算視野射線和此牆的交點在此牆上的百分比數，將視野切至被牆擋住的範圍外，並且遞迴至下一個合法的 Cell 中

Vertex2f:

將依照上述步驟而產生的新 start 和 end 依照

(start[1], 1, start[0])

(end[1], 1, end[0])

(end[1], -1, end[0])

(start[1], -1, start[0])

傳入 MyGlu::Vertex3，在裡面先乘上 FuncMatrix::modeView，再乘上 FuncMatrix::projection，將座標點轉換到平面座標