Computer Graphics Homework #2 Report

103062528, CS, Yu-Shan Lin

**一、架構設計**

上圖為我的系統之架構設計，大略可以分為「Classes」與「其他」兩個部分：

1. Classes
   1. Matrix

基本上就是一個 4 x 4 的二維矩陣，並提供一些矩陣運算的methods

* 1. Model

紀錄單一model的各項資料，包括vertex、color，以及用於這個model的transform matrixes

* 1. World

負責管理所有的model，並在一開始時將model移到正確的位置

* 1. Camera  
     代表使用者的視角，掌控eye、center、up vector等資訊
  2. Projection

掌控projection的方式，並產生對應的transform matrix

1. 其他
   1. Render

包含與繪圖相關的functions，並且負責管理world、camera與projection的物件

* 1. Event Handlers

主要負責處理滑鼠與鍵盤的事件，並呼叫對應的動作

* 1. Main

程式的進入點，負責註冊上面兩個檔案的functions

**二、程式執行特點**

本程式當onRender這個function被呼叫後，就會先產生一個存有identity的matrix物件。並從camera與projection取出預先計算好的transform matrix，再乘上前面的matrix。做完之後，將得到的結果轉交給world，也就是呼叫world的draw，讓world畫出想畫的內容。World則會再轉交給其管理的model，讓model自行根據需要繪圖。

這邊特別撰寫matrix class，主要是為了要實現類似早期OpenGL之中，matrix stack的效果。因為function的呼叫就如同stack一般，一層一層往內，同時如果使用call by value的方式傳遞matrix物件的話，那麼上層function中的matrix就會保有原本的資料，不受到下層function的影響。這樣就可以實現類似的效果。

**三、程式使用方式**

使用滑鼠拖曳或者滾輪來調整x, y, z數值

想要切換模式可以按以下按鍵：

**Model Transform**

T: Translate Mode

R: Rotate Mode

S: Scale Mode

M: Switch model

**Camera Control**

E: Eye Control Mode

C: Center Control Mode

U: Up Vector Control Mode

**其他指令**

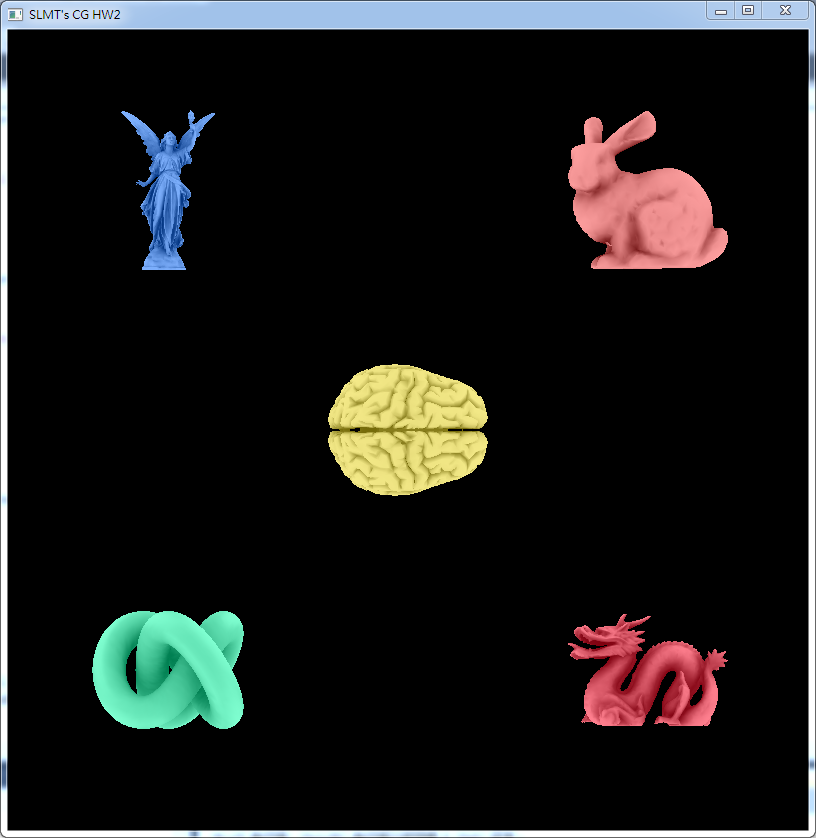
P: 切換Projection Mode

Enter: 重設Camara或者Model (依照現在模式判斷)

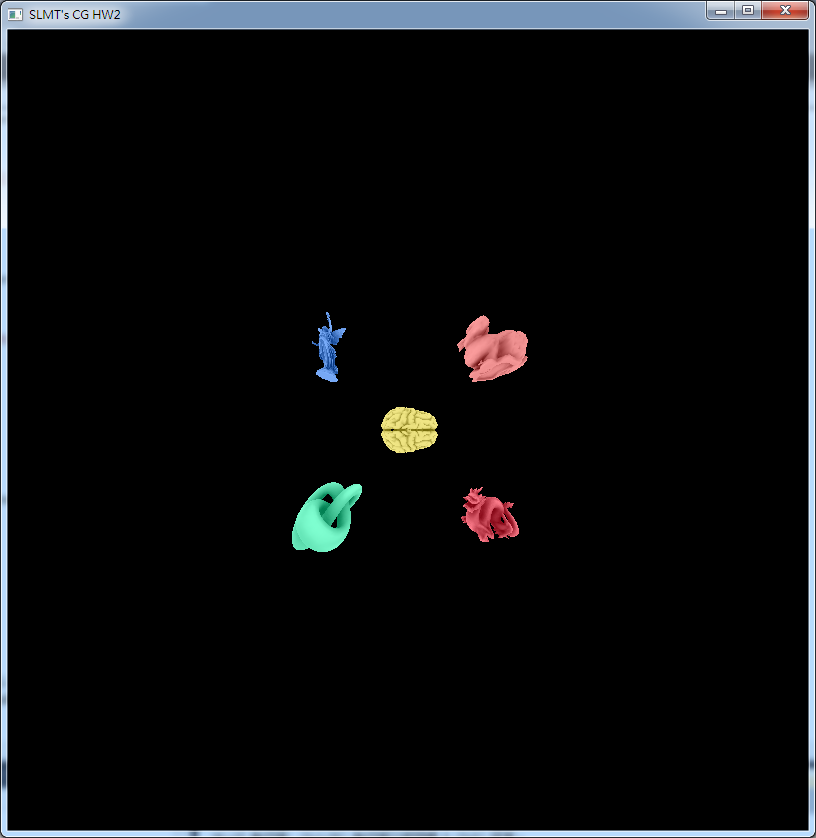
H: 顯示幫助清單

**四、程式截圖**

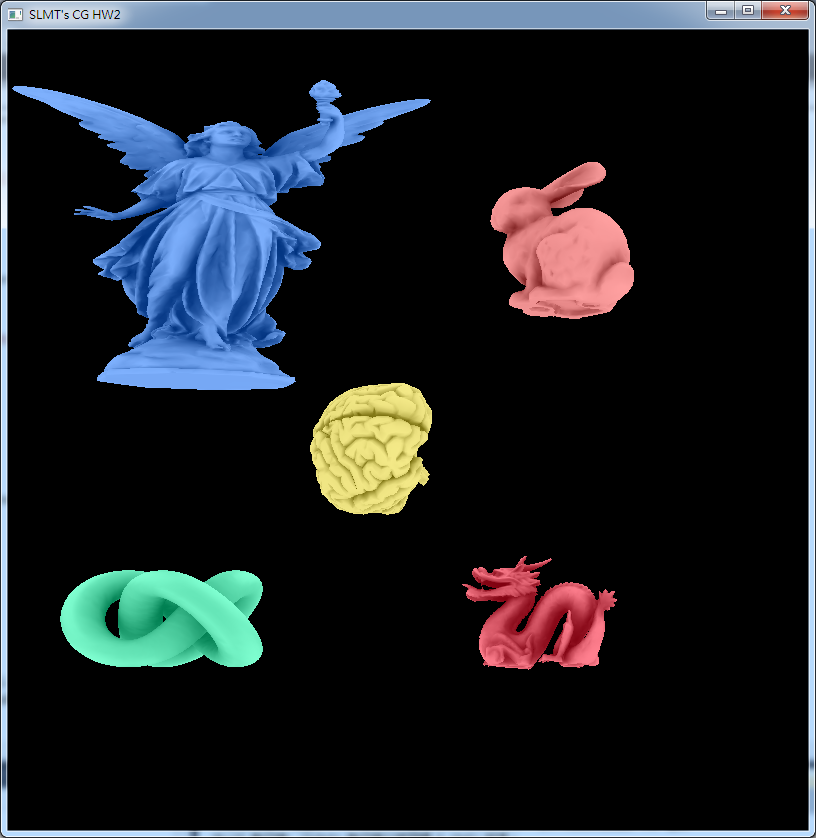
**1. Orthogonal Projection**

****

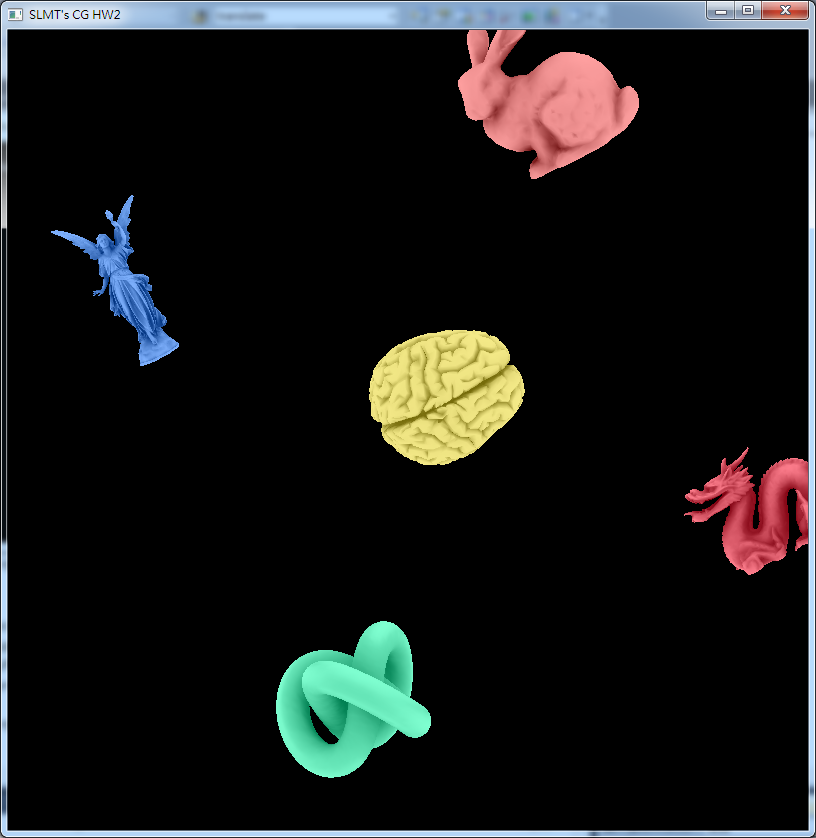
**2. Perspective Projection**

****

**3. Model Transform**



**4. View Transform**

****

**五、心得**

這次作業花最多時間的是在設計架構吧。因為想要設計出方便讓多個model共存的架構，又可以在以後快速加入新功能。因此就設計出現在這樣簡潔漂亮的架構。不過代價就是遲交一周，而且還為了打這份報告又多遲交十幾分鐘，所以總共遲交兩周 (扣6分Orz)