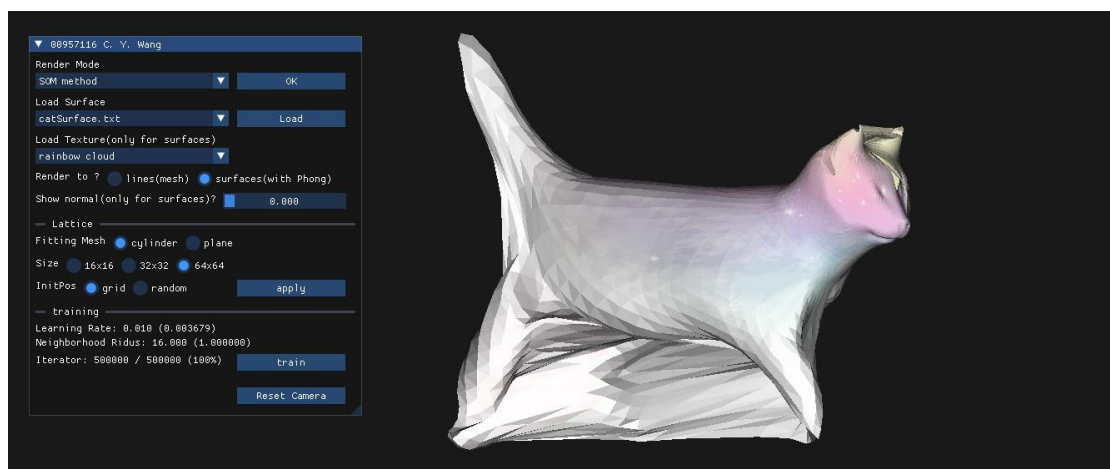


# Computer Project5: Fitting surface, base on SOM

00957116 王嘉羽

## 1 介紹

這次作業是要使用 Self-Organizing Map 演算法，將網格點貼合模型。首先會生成  $k \times k$  網格點的，並開始迭代(train)。每次迭代會隨機選一筆資料(模型位置)，並找出目前最接近資料的網格點，找出後將該點和其鄰居更新，使網格點更貼合模型。在 SOM 中需要特別調整的參數有：學習率(learning rate)、更新鄰居範圍(neighbourhood radius)、迭代次數(iterations)；需要調整的 function 有：學習率和鄰居範圍下降的方程式。完成以上後，將結果使用格子線或是三角形呈現。另外，也需算出法向量畫出明暗的效果。



## 2 實作方法 – Self-Organizing Map (SOM)

- 建立大小為  $k \times k$  的網格點，並隨機給網格點  $n$  個值( $n$  為訓練資料的維度)
- 每次迭代，隨機選一筆資料，並找出最像的網格點(BMU)
- 利用歐幾里得距離計算網格點間的位置，並更新自己和鄰居
- 隨著迭代次數更新學習率及鄰居範圍
- 利用外積計算三角形的法向量
- 利用幾何著色器將法向量視覺化
- 可以透過 GUI 調整 lattice 及渲染設定

## 3 新增的功能

- Load surface: 可以選擇不同擬合的表面
- Load texture(only for surfaces): 可以選擇不同的貼圖
- Render to?: 可以選擇畫出線(lines)或是表面(surfaces)
- Show normal(only for surfaces): 調整要畫出的法向量大小
- Fitting Mesh: 可以選擇網格形狀
- Size: 調整網格大小
- InitPos: 選擇網格的起始位置(分布)
- Train: 開始訓練模型

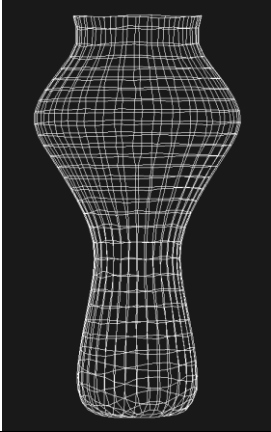
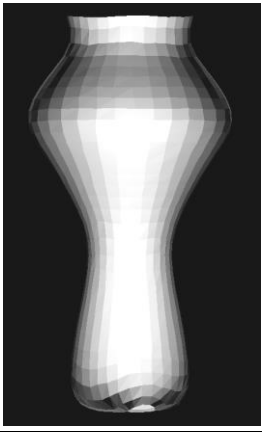
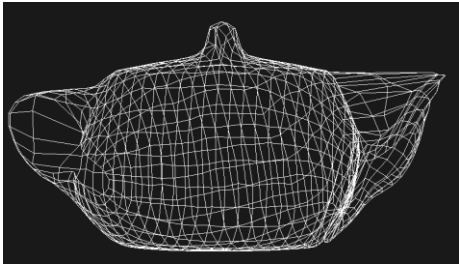
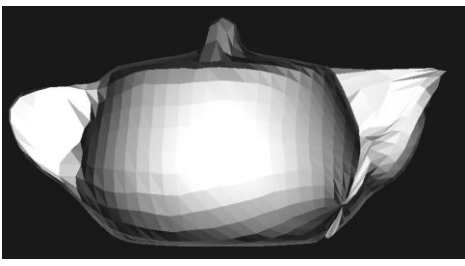
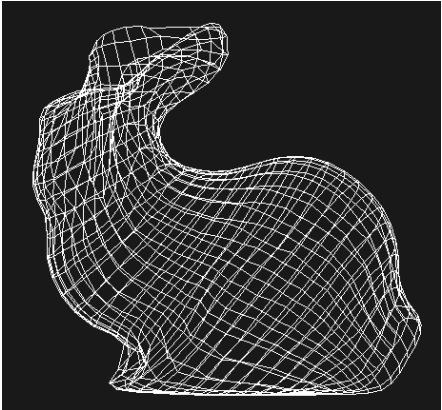

#### 4 參數選擇

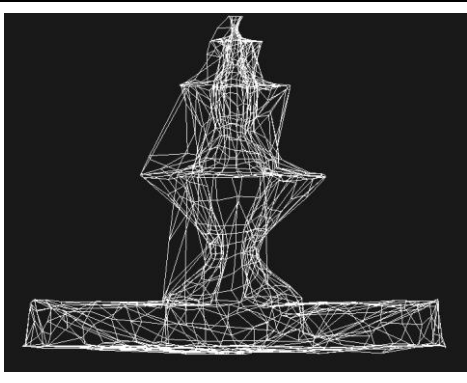
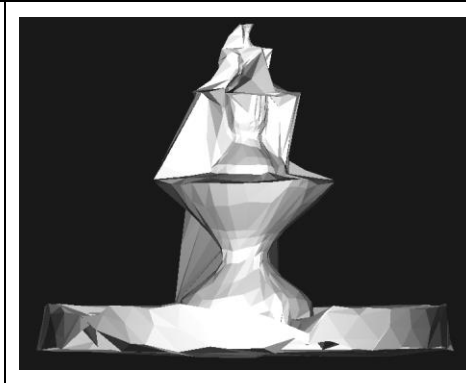
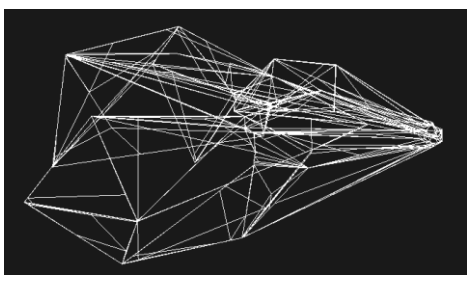
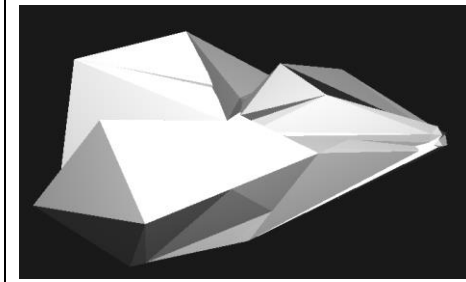
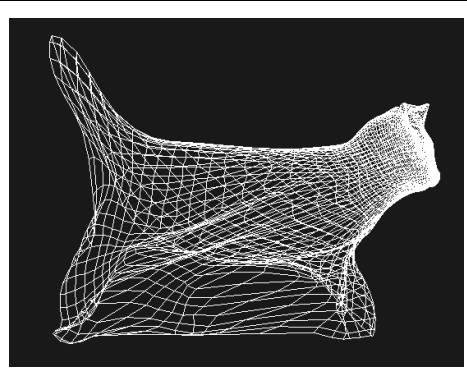
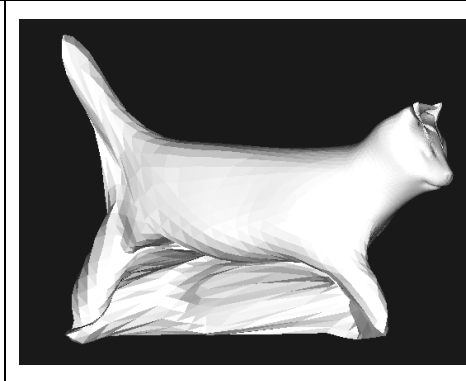
參數	值
Start Learning Rate	0.01
Start Neighbourhood Radius	k/4
Iterations	500000
Learning Rate	Start Learning Rate * $\exp(\text{當前迭代次數}/\text{iterations})$ ;
Neighbourhood Radius	Start Neighbourhood Radius * $((\text{Iterations} - \text{當前迭代次數}) / \text{Iterations})^2$ (min = 1)

#### 5 成果展示

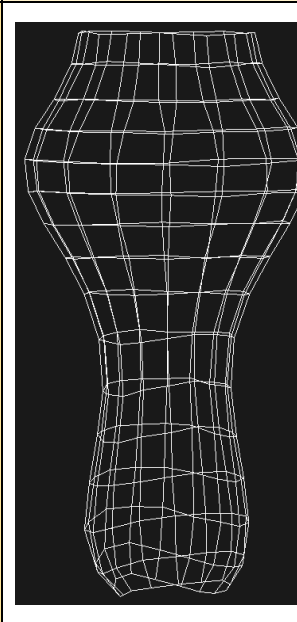
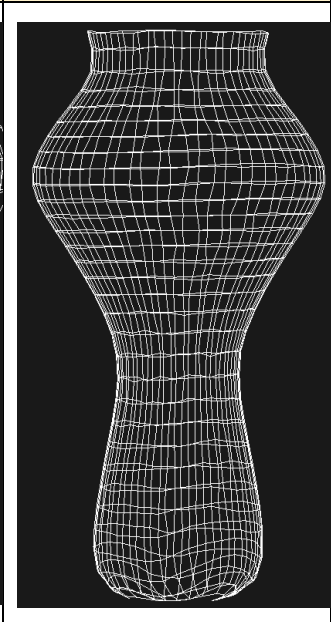
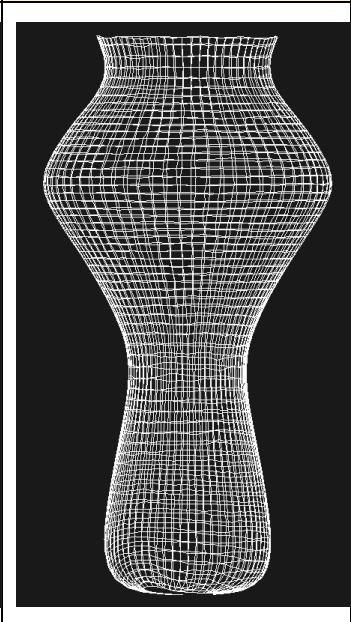
Demo Video: <https://youtu.be/8l7O3fyZHcQ>

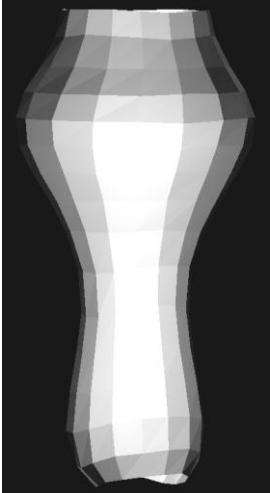

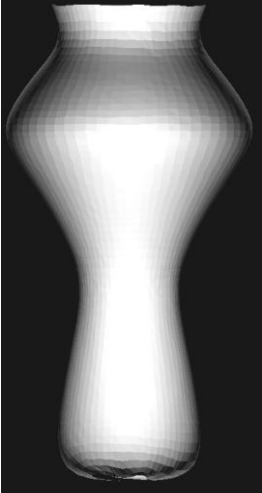
##### 5.1 不同的 model 渲染成不同的樣子

	line	surface
<b>vase</b> (32*32)		
<b>teapot</b> (32*32)		
<b>bunny</b> (32*32)		

<b>fountain</b> <b>(32*32)</b>		
<b>cloud</b> <b>(64*64)</b>		
<b>cat</b> <b>(64*64)</b>		

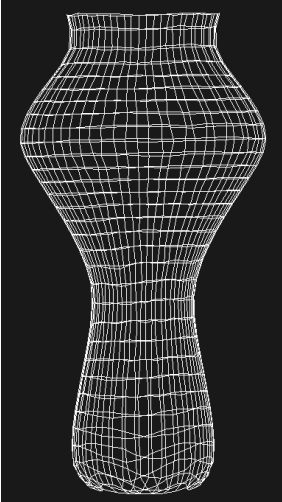
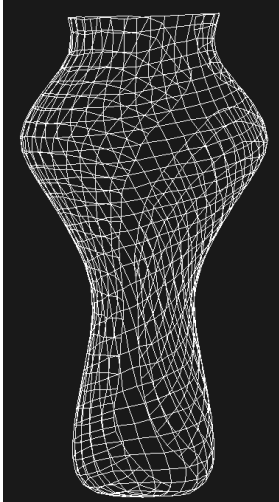
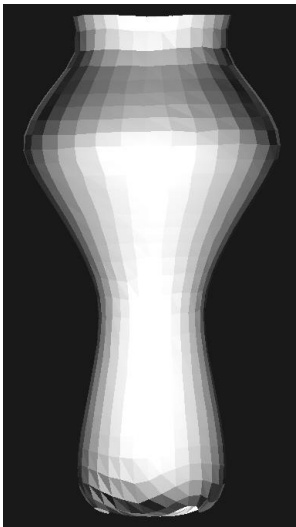
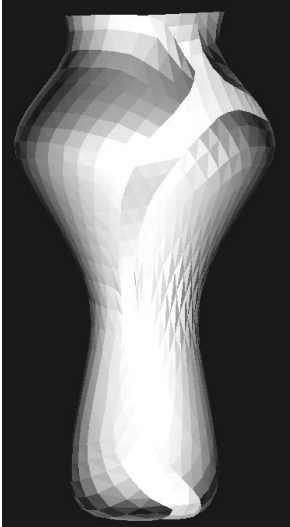
## 5.2 不同的 lattice size

	16 * 16	32 * 32	64*64
<b>line</b>			

surface			
---------	---	---	---

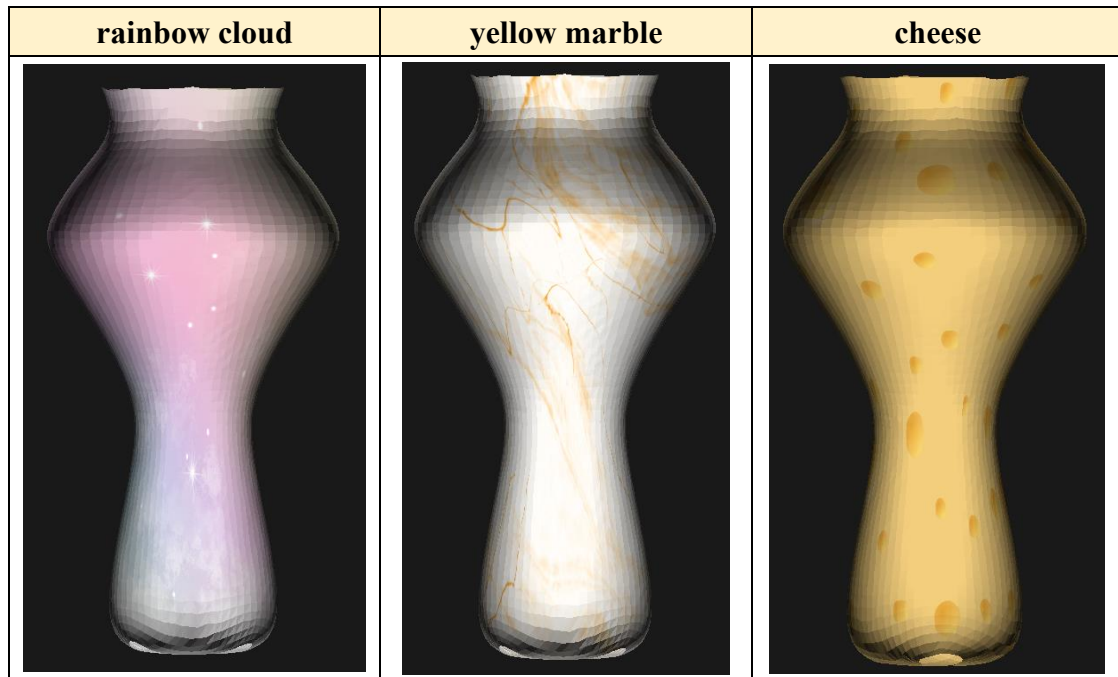
### 5.3 不同的 fitting mesh

皆是使用 32 \* 32 的網格

	cylinder	plane
line		
surface		

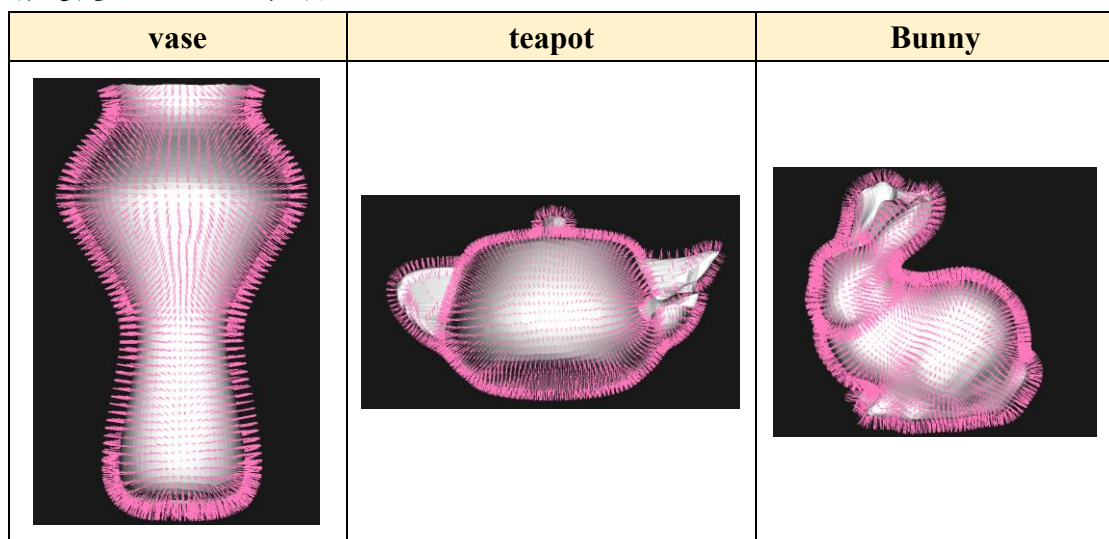
### 5.5 貼上紋理

皆是使用  $64 * 64$  的網格



### 5.5 法向量視覺化

皆是使用  $64 * 64$  的網格



## 6 心得

終於完成作業，這學期也隨之結束！這堂課內容很充實，學到很多視覺化的演算法，還有一些小知識，像是騎摩托車頭髮可能會往前飛。這次作業有一些參數可以調，調成不同的結果都會改變，讓我聯想到機器學習，調參數是非常痛苦的過程 q 我足足調了一個下午才有比較好看的結果~