

第十九屆旺宏科學獎

成果報告書

參賽編號：SA19-086

作品名稱：水膜戳戳樂

姓名：李惟平

關鍵字：紗窗、水膜、影像處理

一、摘要

本篇研究探討常見現象－紗窗上水膜在破裂時的行為與原理。研究中使用光學方法紀錄紗窗上水膜之存滅，一次實驗就能累積大量資料。利用手機錄影，配合適當的光線，以加大水膜破掉前後的變化。將影像資料以程式分析，得到水膜存活率與水膜間相互作用關係之數據。在探討相關因素（重力、表面張力、表面波）與分析水膜統計資料後，利用傅立葉分析找出是否具有隱藏之相關性。

二、研究動機

洗完紗窗後，紗窗的網格內常有水形成的薄膜，慢慢的一格一格的破掉。讓我聯想到古早的遊戲--戳戳樂。想了解某一格水膜的生存時間或那幾格水膜先破掉，這格的水膜破掉，鄰近格子的水膜生存時間是增長或減少，似乎是非常困難的事。因為一個紗窗內含好幾千個網格。

二維條碼讀圖程式給我很大的啟發，例如，有一個在 Windows 執行的程式 QuickMark，可以不經過照相機的鏡頭，直接讀取二維條碼。紗窗上的水膜好像二維條碼一樣，應該可以設計程式，直接去讀取。這樣不需使用大量的人工去判斷水膜有無破掉。

三、研究目的

A. 現象觀察：

1. 觀察不同傾角時紗窗上水膜的破裂情形。
2. 以高速攝影觀測局部水膜破裂與其物理意義。
3. 討論重力、表面張力與表面波在系統中之作用。

B. 統計數據：

1. 探討水膜破裂情形是否具有重複性。
2. 探討紗窗上水膜存留率隨時間變化。
3. 計算出水膜間存留時間之相關係數。
4. 在 3*3 格範圍內，探討水膜存留情形對中央水膜生存率之影響。

C. 導論：

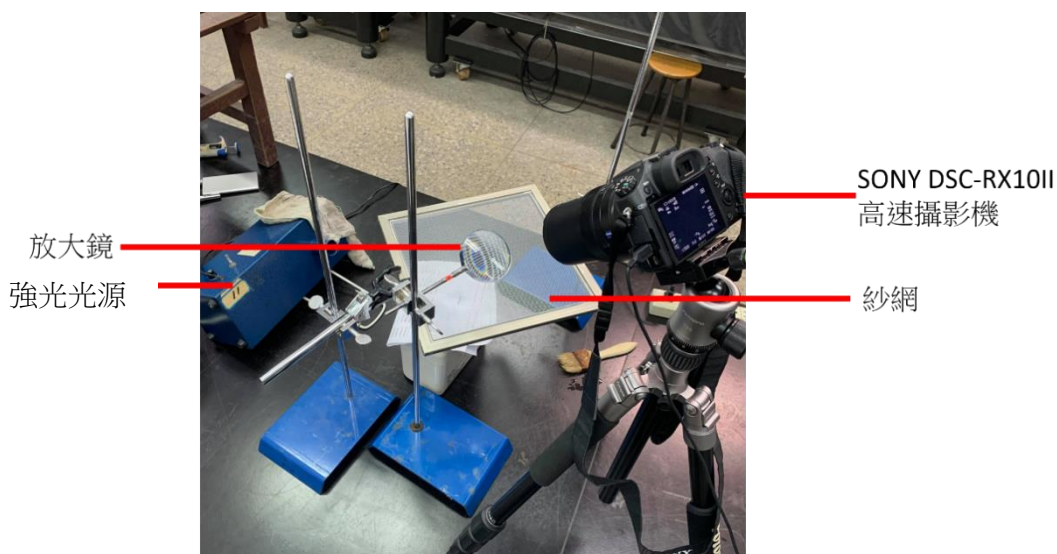
1. 對相鄰、對角關係之水膜進行分析，建立理論模型。
2. 對水膜存留曲線建立定性與定量理論。
3. 對現象提出解釋。
4. 對影格水膜存留/破裂矩陣進行傅立葉分析，尋找隱藏的關聯性。

四、研究設備及器材

以下為本次兩種實驗的實驗裝置：

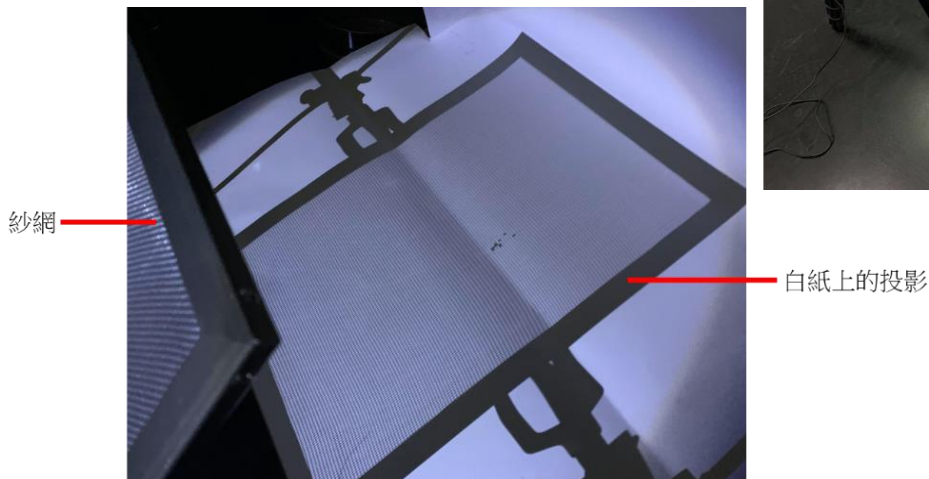
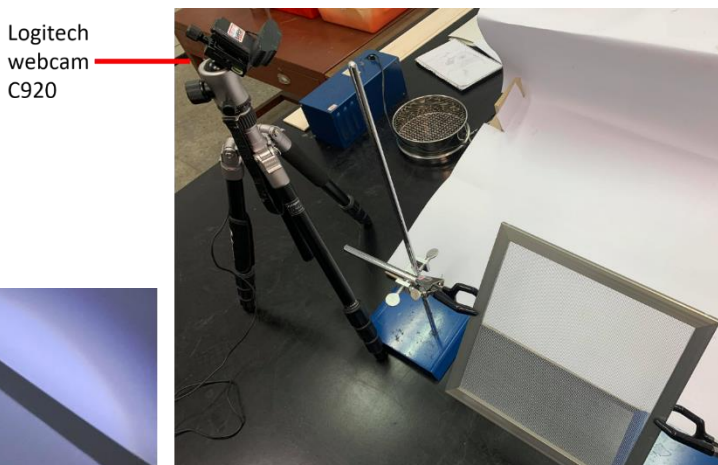
A. 高速攝影、局部放大：

使用 SONY DSC-RX10II 高速攝影機配合凸透鏡局部放大紗網上數格水膜，在水波槽光源照射下捕捉單一水膜破裂的畫面，以利探討破裂原因，以及過程中對周圍水膜的影響。



B. 不同紗窗傾角水膜生存實驗：

使用 Logitech webcam C920 紀錄紗網與水膜在白紙上的投影情形，以利後續大範圍分析紗網中水膜的存留/破裂。



使用器材清單：

設備與工具	數量	設備與工具	數量	設備與工具	數量
30 cm*30 cm 紗網	1 個	放大鏡	1 支	SONY DSC-RX10II 高速攝影機	1 台
白色紙屏	1 個	相機腳架	1 支		
Logitech webcam C920	1 台	水波槽光源	1 台		

使用程式：Python, C++, Matlab

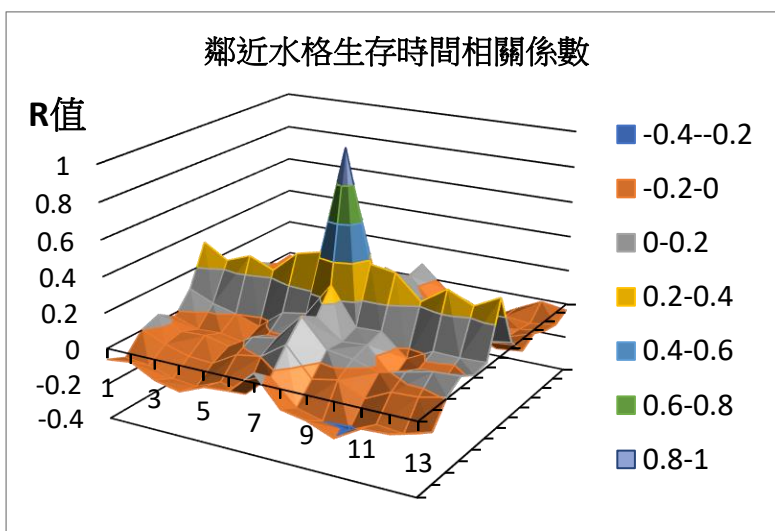
五、研究過程及方法

A. 紀錄現象並蒐集數據：

1. 肉眼觀察：使用 Webcam 錄影後直接觀察紗窗傾角對水膜破裂順序的影響。
2. 局部放大慢動作分析：使用相機 HFR 模式紀錄水膜破裂時表面波的形成與傳遞，並觀察殘餘水分之行為。
3. 製作存留時間矩陣 (T 矩陣)：首先將紗窗與水膜投影至白色紙屏上並錄製，相對實體提高破裂與存留之水膜間的對比度，並以程式將影片轉為二維矩陣呈現的數據。

B. 水膜生存時間與鄰近水膜相關性矩陣分析：

藉由選取水膜存留時間矩陣多組 13 格*13 格範圍，並將座標 (7,7) 水膜之存留秒數與其餘水膜進行線性回歸，可得該座標關係之相關係數值。



註：為避免實驗總體誤差（如加水過多，致所有水膜生存時間一同變長）造成分析之誤差，每組資料皆先扣除相對應 13*13 格水膜生存時間之平均，再進行相關性分析。

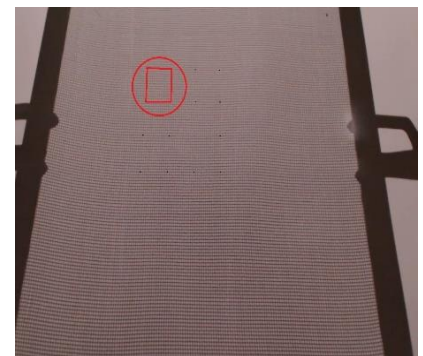
*附錄（三）為計算相關係數矩陣所使用的 matlab 程式碼。

C. T 矩陣分析：利用水膜生存時間矩陣分析水膜生存與互相影響的行為模式。

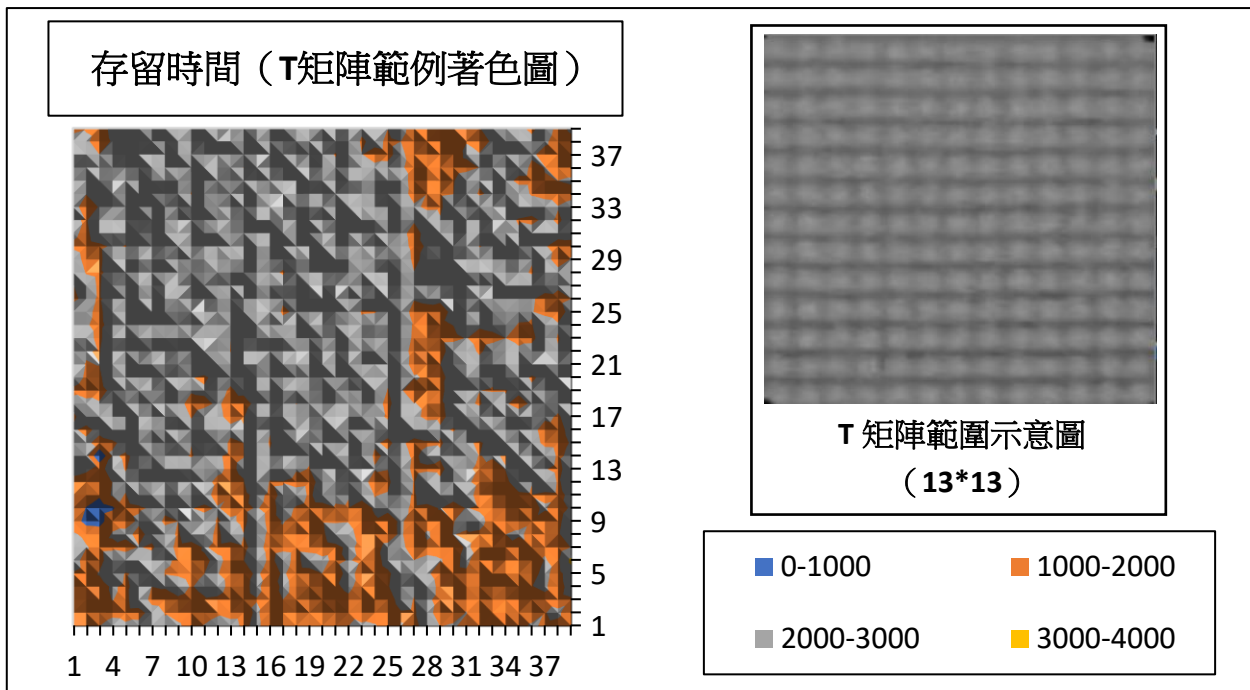
T 矩陣之定義：T 矩陣內含有 39 * 39 格資料，每格資料對應到一個紗窗上之水膜，而資料即為此水膜在觀測資料中破裂的時間點。

T 矩陣之資料取得方法：我們將錄製的影像資料進行**梯形校正**，取得每個紗窗網格在影像中之像素座標 (pixel)，並以 python 程式訂定**亮度門檻**以判斷水膜是否破裂。

* 附錄（一）為計算存留時間所使用的 python 程式碼。

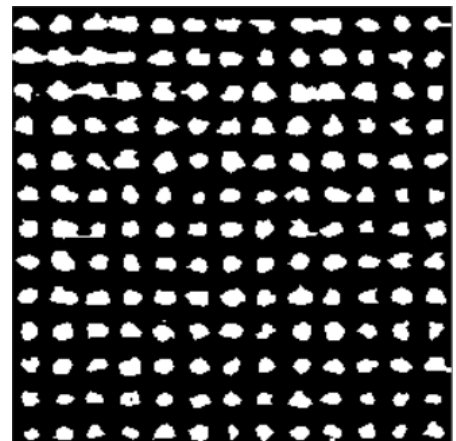


梯形校正示意圖



T 矩陣資料範例：格中數字為水膜之生存時間

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	1763.5	1749.5	1988	2197	2063	2137	2321	2043.5	2087.5	1783	2023	2169	2241.5
2	1696.5	1649	1728.5	1885	1856.5	1834.5	2171.5	1986.5	1803	1687	1812.5	1935.5	1783.5
3	1757.5	1772.5	1820	2010.5	1898.5	2285.5	1885.5	1844	1949.5	1872.5	1963	1920.5	1715
4	1823	1842.5	1812	2064.5	1908	2193.5	2001.5	2028.5	2136	1979	2088	2225	1856
5	1964	1800	2027	2355	1963.5	2088	2287.5	1969.5	1963.5	1900.5	1998.5	2021	1824
6	2273	1752.5	1873	2242.5	2404.5	2308.5	2210.5	2025.5	1978	1856.5	1936	2088	1887
7	2203.5	2026	1898.5	1920	1788	2026	1936	2247	1871.5	1781	2043	1900.5	1824.5
8	2053	1914	1715.5	2039.5	2127	1886.5	2131.5	2168.5	1950.5	2295.5	2188.5	1915	2247.5
9	1766	532.5	471.5	1844	1827.5	2067.5	2248	1925.5	1999	2056	2129.5	1979.5	2069.5
10	1783.5	933	465	827	1900	1949	2039	2087	2071	2010.5	1935.5	2035	1857.5
11	1976	1639.5	1137	2121	1983.5	2196	2342.5	2127	2287.5	2087	2160.5	1847	1971.5
12	1873	1817	1162	2014.5	2070	2001	2052	2024	2164.5	2116.5	2072.5	1949.5	1900.5
13	2155.5	2065.5	1898.5	2138	1984	2117.5	2180	2437.5	2377	2297.5	2125	2052	1949.5



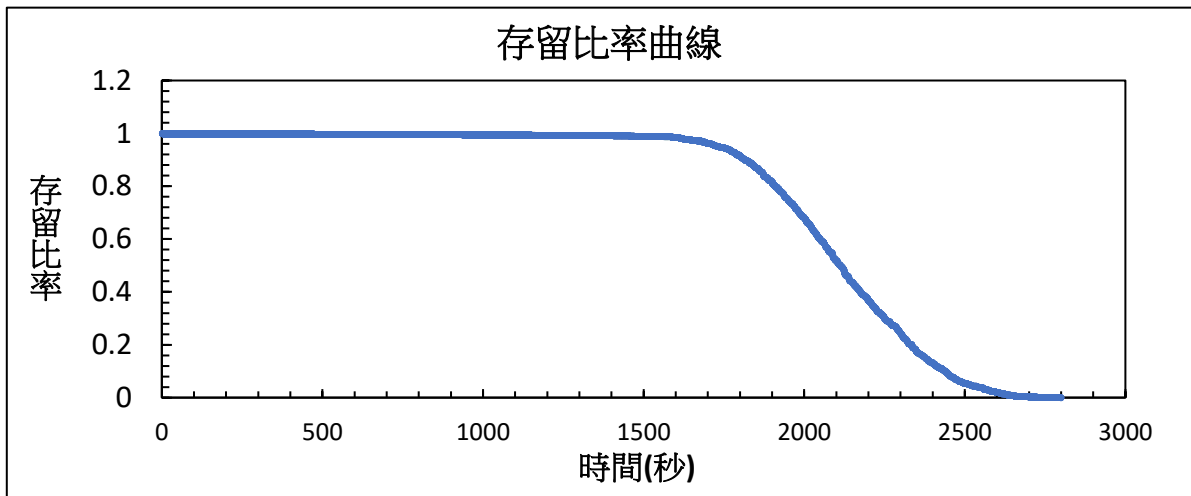
亮度門檻篩選示意圖

由 T 矩陣資料可以進行有關水膜破裂模式的更多分析：

四個 C++程式分析：範圍內水膜存留率九宮格圖形之出現時間分布、破裂機率與破裂時機。

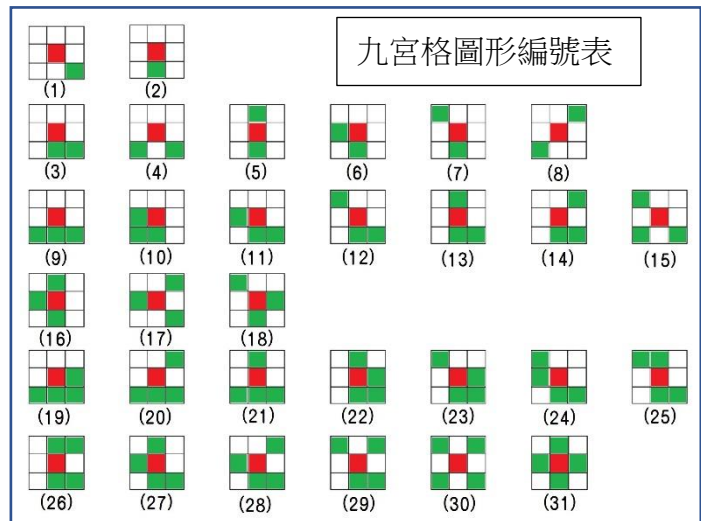
1. 圖形出現時間分布：各種圖形在範圍內隨時間之豐富度。*附錄（四）
2. 九宮格圖形之定義及編號原理：各圖形破裂時間：各種圖形中央水膜在某時間破裂之數量統計。*附錄（五）
3. 各圖形破裂機率：各種圖形中央水膜在單位時間內破裂之機率。*附錄（六）

4. 39*39 範圍內水膜存留率：得到 T 矩陣後，便可利用附錄（二）之迴圈計算出每個選用畫格存留的水膜數目，進而推得水膜存留比率隨時間的圖形。



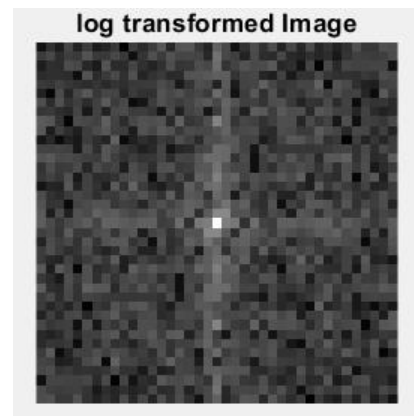
5. 藉由 C++ 程式將九宮格編號由 1 排至 50，討論 3*3 範圍內正中央以外的八個水膜的存亡情形，其圖形如圖所示。

註：n 代表外圍八個水膜存留的水膜數， x 代表圖形編號。n = 5、6、7、8 的圖形為 n = 3、2、1、0 之圖形反白，且編號 $50 - x$ 為編號 x 圖形反白。



D. 傅立葉分析：

藉由附錄（七）之 matlab 程式可由水膜存留時間矩陣逆向推導出任一畫格之水膜存留/破裂情形矩陣，再利用 matlab 內建之二為傅立葉分析功能得出對數轉換後的傅立葉分析情形。



- E. 討論鉛直與水平實驗組中重力與表面張力產生的影響。

六、研究結果與討論

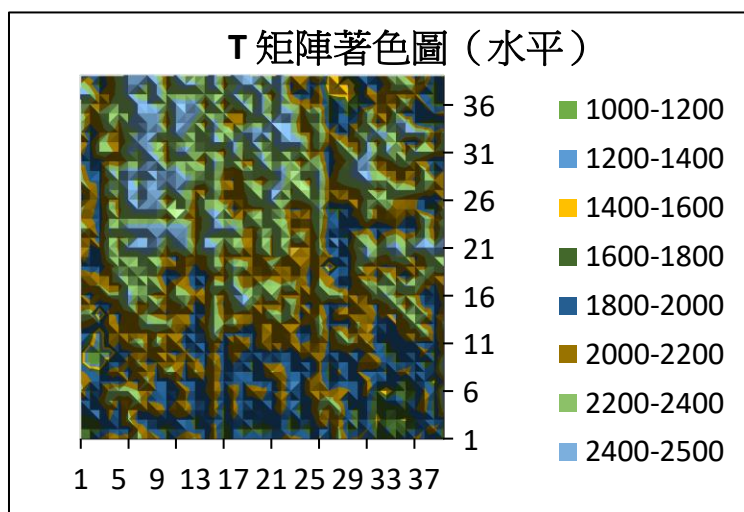
A. 紀錄現象並蒐集數據：

肉眼觀察：

不同傾角下皆能觀察到紗網上和水膜內水分分為兩個階段消失。紗網上的水分先蒸發消失，水格內的水膜接著破裂。

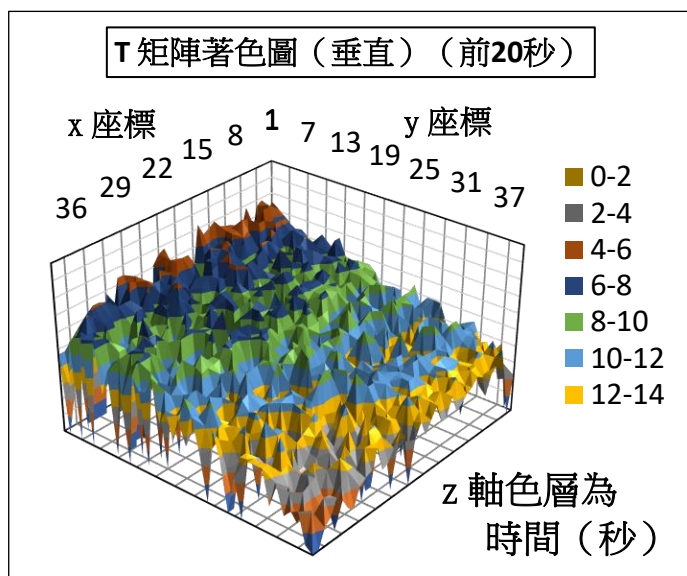
水平組：

將水膜破裂投影的影片轉換為 T 矩陣後，觀察水平組別的數據，發現接近紗窗中心之水膜平均存留時間較外部為長。

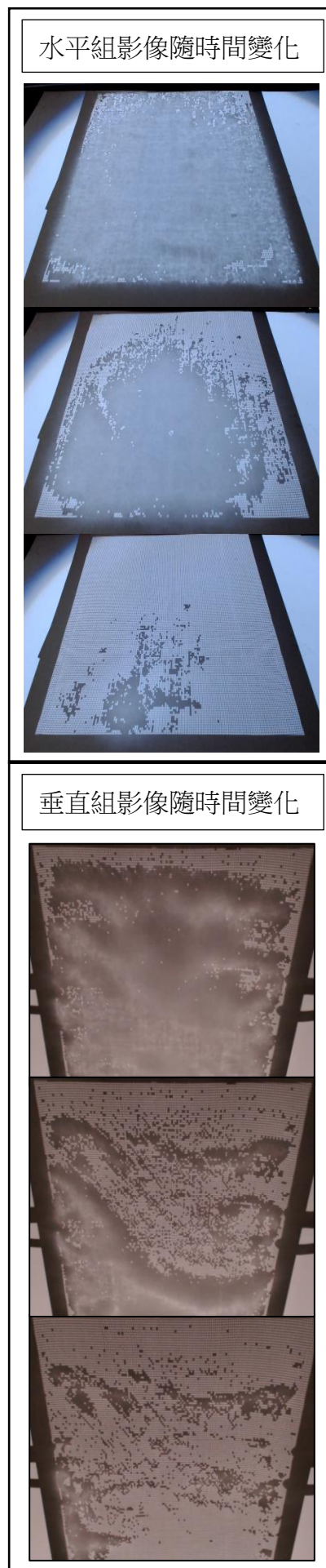


傾斜與垂直組：

傾斜與鉛直的實驗組中，當紗網上水分向下流離紗網後，不少水膜就已破裂，僅部分區域水膜群聚存留繼續破裂。

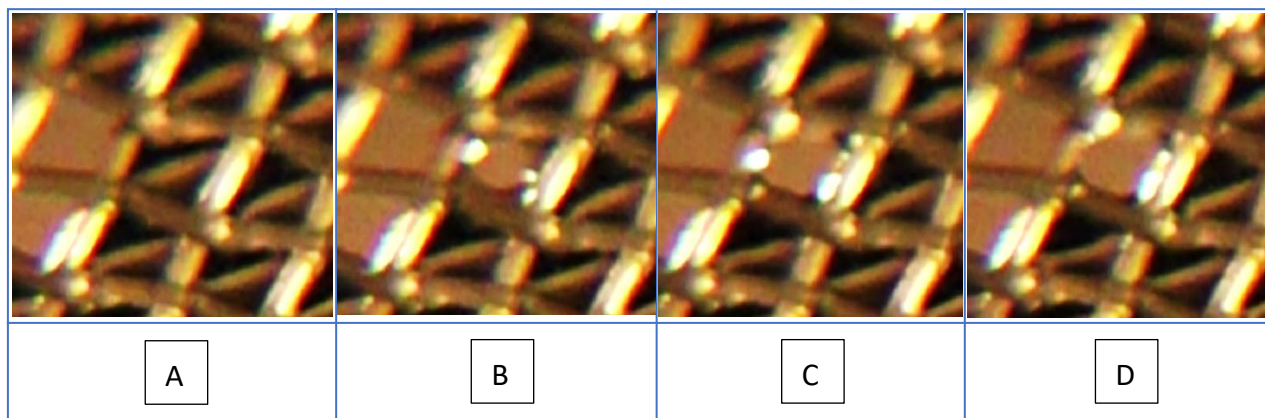


傾斜與鉛直的實驗組則明顯觀察到前 30~40 秒水膜破裂的順序由上而下，且餘下的水膜易產生群聚，而群聚內的水膜亦由上而下破裂。



局部放大慢動作：

觀看單一水膜破裂過程的高速攝影可觀察到向外傳遞的表面波，且破裂水膜與相鄰和對角水膜皆有接觸。另外，水膜破裂後，剩餘的水分將在四角形成水滴流向相鄰水膜。



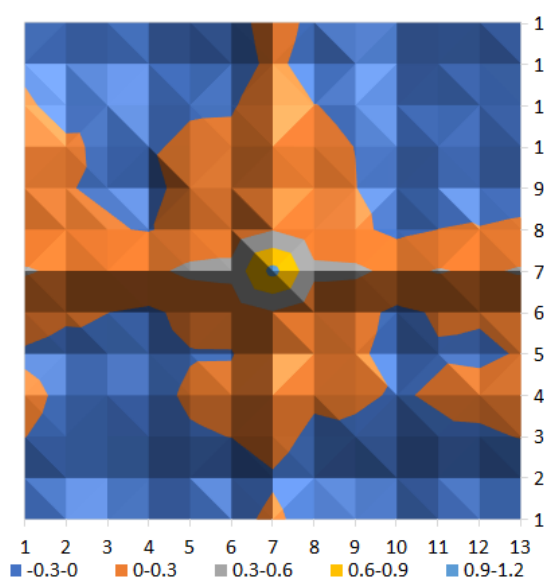
推論：

1. 相鄰水膜之水分均有直接接觸，且可以**互相補充**，並因表面張力施以拉力。
2. 紗網上的水因重力流動時易牽動水格內水膜導致破裂。
3. 在任何非完全水平的情形下，**重力**會造成水膜間的**不對稱性**，導致離地面較遠者較易破裂。不對稱性可能源自下方水膜無法將水補充至上方水膜，因此導致鉛直組別中殘餘的群聚水膜仍由上而下破裂。
4. 導致水膜破裂的因素包含：**水分蒸發**使水膜變薄、水膜形狀受鄰近水膜表面張力影響改變，進而使**面積變大**、**表面波**傳遞時，水膜形狀改變。

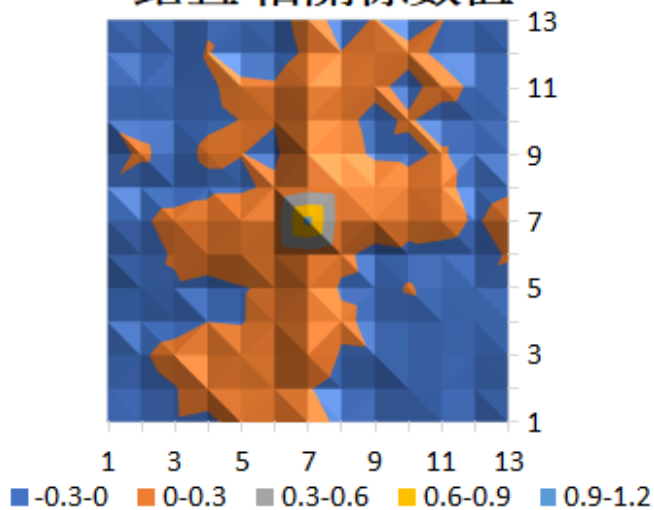
B. 相關性分析：

對比水平與鉛直組別，則注意到水平組與基準水膜相鄰之水膜相關係數值較高，鉛直組則反之。

水平相關係數值



鉛直相關係數值



觀察水平與鉛直組別相關性分析矩陣，皆可發現相同距離下，與觀測基準水膜為橫向或直向相接者，相關係數值較高。

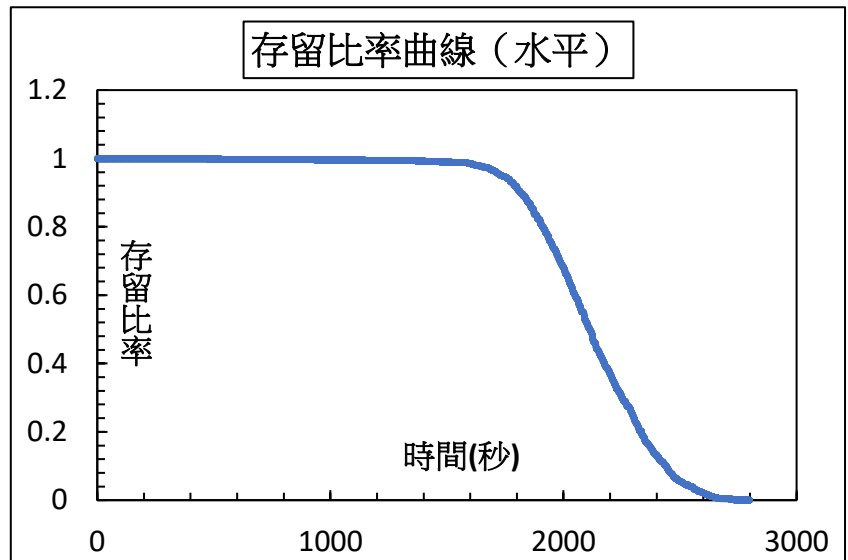
推論：

1. **力平衡假說**：水膜破裂並非單純因水分蒸發導致，亦受到相鄰、對角水膜不均勻拉力影響。若僅考量蒸發一破裂的機制，破裂時間應近似隨機，且無相關性。考慮中間水膜較早破裂時，造成周遭水膜受力不均勻，因而較易破裂。假說詳細驗證於「力學模型與驗證」中呈現。
2. 鉛直實驗組中紗網上水分受重力影響流動過程連帶使部分水格內水膜破裂，且此過程近乎隨機；水平實驗組中紗網上的水未因重力影響而流動，而是蒸發消失，去除隨機破裂的過程，導致水平實驗組中基準水膜相鄰之水膜相關係數較鉛直組為高。
3. 相同距離之相鄰水膜較對角水膜相關係數高：因為相接水膜除了因表面張力互相拉扯外，亦提供補充水分的作用。此敘述詳細驗證於「力學模型與驗證」中呈現。

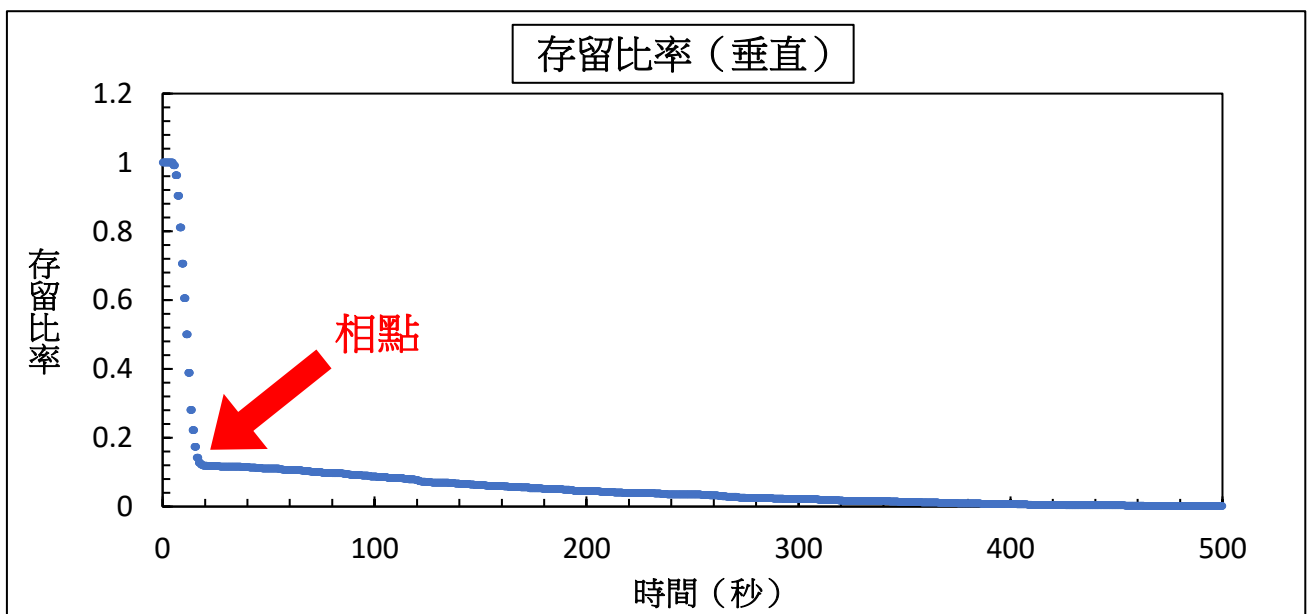
C. 力學模型與驗證：

存留比率曲線 - 數據觀察：

水平存留比率曲線在範圍中有初始破裂時變化平緩，當累積到一定數量時迅速破裂，在大量水膜破裂後又趨於平緩。



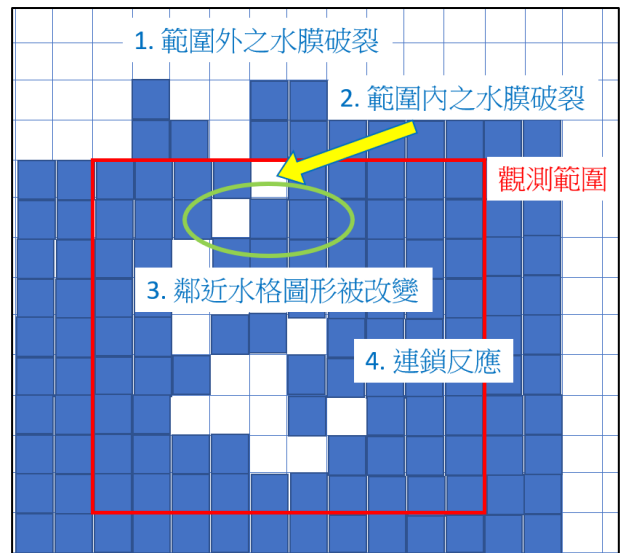
(下圖) 垂直組之存留比率曲線中出現了一個「**相點**」，且此項點出現之時間點均在範圍內第一個水膜破裂後之十五秒內。相點後之曲線十分近似於水平組數據，我們推測在**相點前**後水膜破裂之模式是有所不同的，且可能與重力有密切關聯。



紗窗上水膜破裂現象解釋（定性）：

水平組：

1. 在實驗之初，由**觀測範圍外**之水膜開始破裂，直到破裂趨勢逐漸向觀測範圍邊界靠近，令範圍中有**第一格**水膜破裂。
2. 自第一格水膜破裂後，周圍之相鄰水膜九宮格圖形被改變（shape 50 變成 49 或是 48）**提升了中央水膜破裂機率**。
3. 周圍水膜隨時間破裂，更外圍之周圍的水膜也被改變了圖形。因更多水膜之**破裂機率在圖形轉換下改變**，而轉換成破裂機率較高之圖形的水膜越來越多，存留比例**迅速下降**。
4. 轉換過程中許多圖形並不是所有圖形皆被轉換成機率較高之九宮格圖形，而是機率較低的，又因水膜不可復生，此類圖形隨時間大量累積、緩慢破裂。
5. 直到最終水膜皆破裂，實驗結束。



垂直組：

1. 紗窗上網格留住水分的能力有限，造成紗窗上多餘水分會在實驗過程中向下流動。由於向下流動的水分對水膜中的水膜亦有吸引力，有機率在流動時拉動網格內水分使其破裂，因此實驗的第一階段便觀察到大量水膜由上而下近乎隨機迅速破裂，使 $n = 7、8$ 的圖形數量銳減。
2. 多餘的水分流離紗窗底部，紗網上僅剩部分水膜群聚存留，進入類似與水平組定性解釋中 3. 的圖形轉換階段，因而產生存留比率中的**相點**。
3. 因為**重力**因素，額外之拉力使水膜整體破裂機率提升，**水膜平均存留時間下降**。
4. 實驗組累積破裂機率較低的圖形，但由於水分補充的限制，最終水膜破裂的時間點提前，實驗結束。

存留比率曲線定量解釋（理論一）：

在存留比例上，越多水膜破裂代表越多圖形轉換，越多圖形轉換將導致更多水膜形成容易破裂的九宮格圖形。（此處設存留比率為 r ）

於是我們假設破裂之速度與破裂之加速度有著**正比**關係：

起始破裂 (shape 50) 方程： $\frac{dx_{50}}{dt} = k \cdot \frac{d^2x_{50}}{dt^2}$ 。

再者，因為各圖形皆有其在一定時間內破裂之機率（固定值），我們可以將每個圖形 i 測得之單位時間內破裂機率 p_i 、含量 x_i 與單位時間內新增含量 y_i （來自其他圖形之轉換）賦予一個破裂公式：

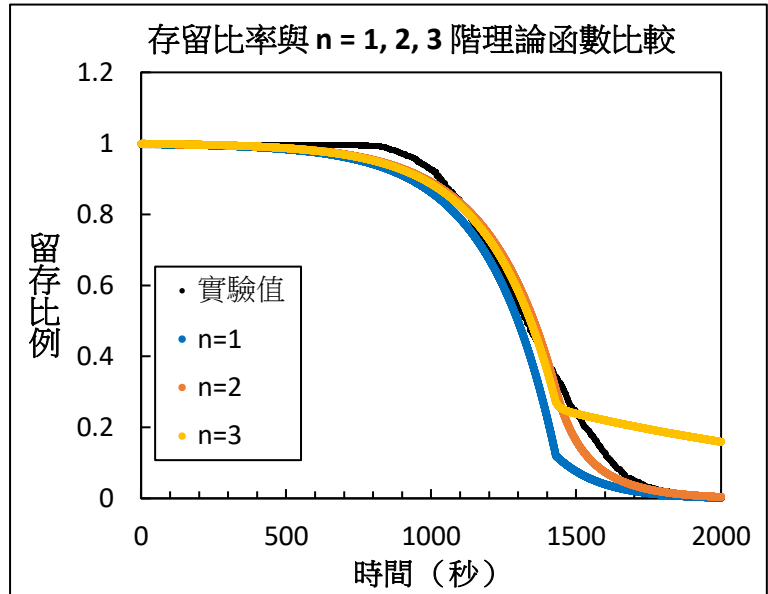
$$\frac{dx_i}{dt} = y_i - p_i x_i, i \in Z, 0 \leq i \leq 49$$

註：全滿之 shape 50 為圖形之初始狀態，故不適用破裂公式。

在存留比率之曲線上，此破裂公式將有 50 個 x_i 隨時間而浮動，計算製圖複雜度過於龐大，下以一階（由所有圖形之平均 \bar{p} 所做之一個破裂函數）、二階（ p_i 最高、最低之 25 個圖形之平均所做之兩個破裂函數）、三階函數與起始破裂方程做配合：

$$r = \sum_{i=0}^{50} x_i$$

高階函數明顯較低階更近於實驗值。



垂直與水平實驗組在存留比率曲線上之差異：

垂直組相點前之曲線經多次實驗測試確認為隨機破裂，僅與大規模表面水流速度有關，而相點後之曲線與水平實驗組僅有絕對值（起始時間早、剩餘格數低）而無相對值（曲線圖形相似性）之差異。

存留比率曲線定量解釋（理論二）：

仔細觀察存留比例曲線與實驗影像，我們不禁思考人類社會中是否有類似的行為就像水膜的破裂模式一樣。過程中我們注意到水膜破裂的模式其實就像傳染病一樣，一格接著一格破裂，於是我們套用了傳染病常用的 SIR 模型來與水膜存留比率進行比較。

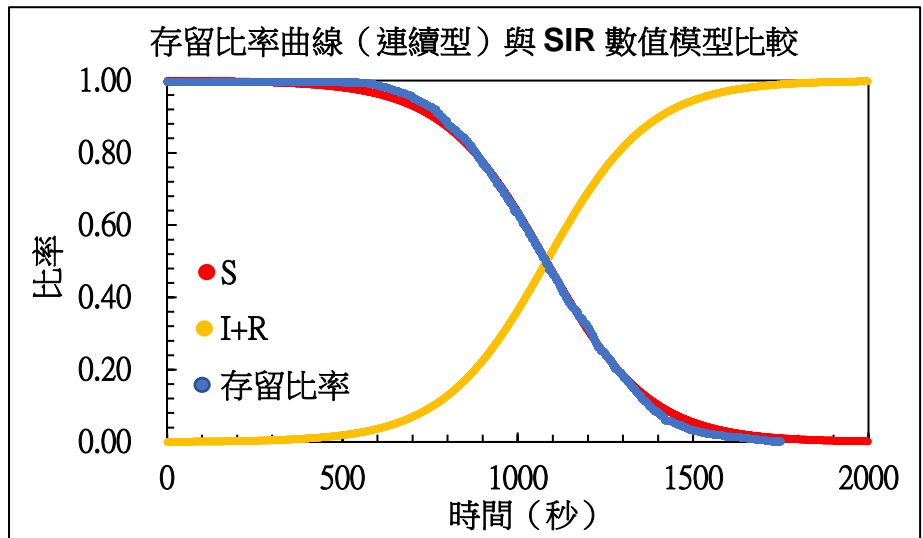
傳染病中的 SIR 模型將所有人口分為三類：可被感染的 S（Susceptible 敏感性的）、具有感染力的 I（Infectious 傳染性的）及已痊癒的 R（Recovered 痊癒的）。在我們的類比中，S、I、R 分別對應到此研究實驗中的倖存水膜、可影響倖存水膜之破裂水膜及其餘破裂水膜。

最簡 SIR 模型假設：痊癒者不復發、傳染發生完全隨機、治癒率固定與人口總數不變。
以方程式來說明如下：

5. 人口總數不變： $I + S + R = 1$ (一般化之總人口數)
2. 傳染過程發生完全隨機： $\frac{dS}{dt} = -\beta IS$ (β' 為一般化修正之傳染係數)
3. 治癒率固定： $\frac{dI}{dt} = \beta' IS - \gamma I$ (γ 為治癒率)
4. 痊癒者不復發： $\frac{dR}{dt} = \gamma I$

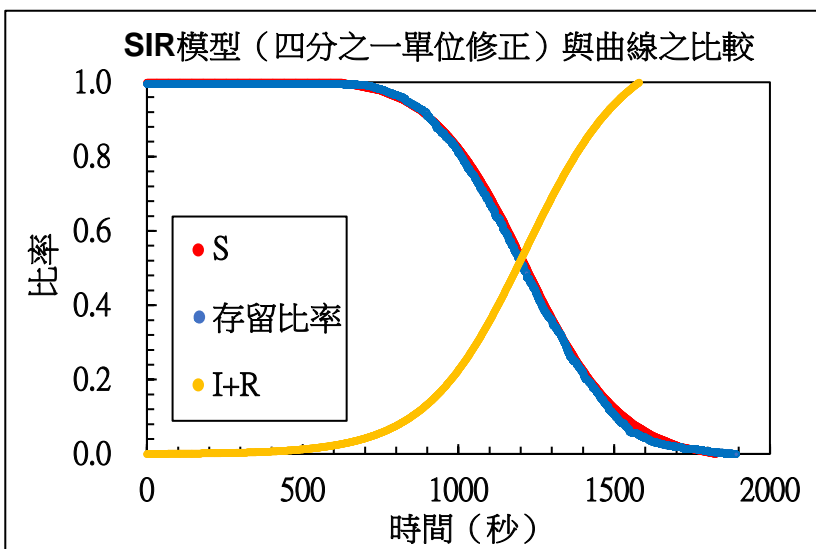
此處我們額將水膜類比為模型中之 S (可破裂之水膜) 利用 Excel 進行 SIR 模型之歐拉法模擬，用實驗資料對傳染率進行擬合 (β 與 γ 值)，發現擬合後 (後面將會討論) 模型與實驗數據吻合程度非常高。

從此模型比較我們甚至可以得到更多有趣的分析：單位、傳染率與整體感染狀況對時間變化等價的有趣意義。



SIR 模型：單位

可是同時我們也注意到曲線與 SIR 模型在開始與結束時有較大的誤差，重新翻查數據後，我們認為與 SIR 模型最大的缺點有關：**連續性**。

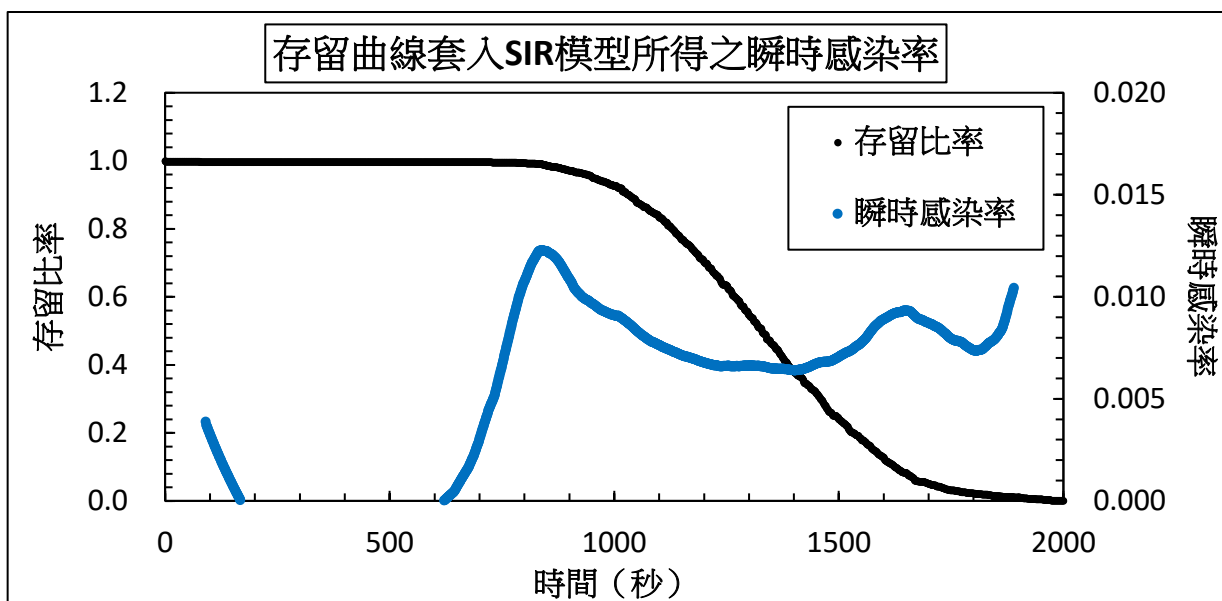


事實上，傳染病是在人體身上一個個傳播，水膜也是一樣，不過當我們利用電腦運算高斯符號以一格水膜來改進 SIR 模型時，吻合程度的提升並不多，直到將單位調整成四分之一個水膜時，模型與曲線有了極 (最) 高的吻合。僅在

重要的是，這個四分之一個結論不僅修正了 SIR 模型，更能解釋後面我們對於每個水膜實為由四個角落的小水滴所組成。

SIR 模型：傳染率

雖然我們對傳染率擬合出的結果大略符合期待，但是畢竟此擬合值為一整個實驗的平均。為了詳細了解感染率在水膜破裂實驗中的意義，我們將水膜存留曲線以 SIR 模型做計算，平滑數據後得到每時刻對應的感染率（破裂影響率）。



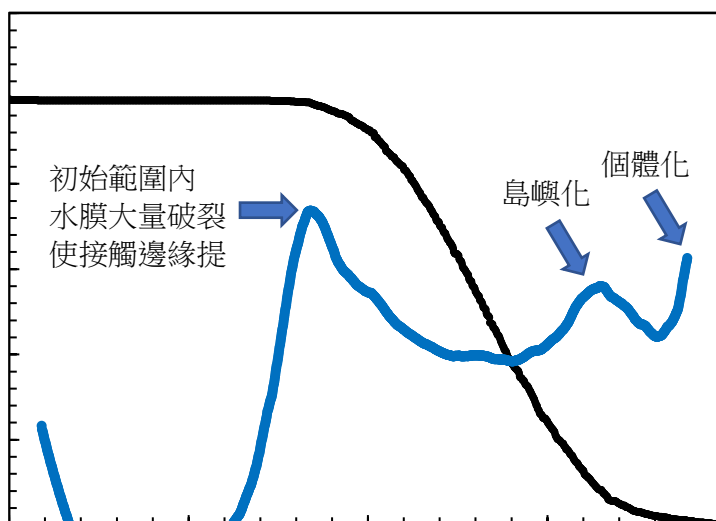
從傳染率的圖形可以看到傳染率在水膜大量破裂的時段前後與實驗結尾都有峰值。

第一個波峰是由於範圍內首有水膜開始大量破裂，邊緣的突破使倖存水膜與破裂水膜之接觸邊緣長度急速增加而導致感染率飆升。

對於後兩個波峰，我們比對影片資料推論這些峰值是因水膜生存「**群落化**」造成。群落化又在實驗中因尺度分為兩階段，**島嶼化**與**個體化**。

島嶼化：因為當範圍中水膜開始破裂後，範圍邊緣的水膜並不會均勻的破裂，造成倖存水膜多互相相連，有如一團團小島，但是此狀況令倖存水膜與破裂水膜之接觸邊界大量延長，所以感染率便上升。

個體化：在島嶼化後，各「小島」又會重復在島嶼化所發生的狀況。但是由於島嶼的尺寸遠小於觀測範圍，第二次形成更小規模的小島，小至一個單一水膜，故稱個體化。



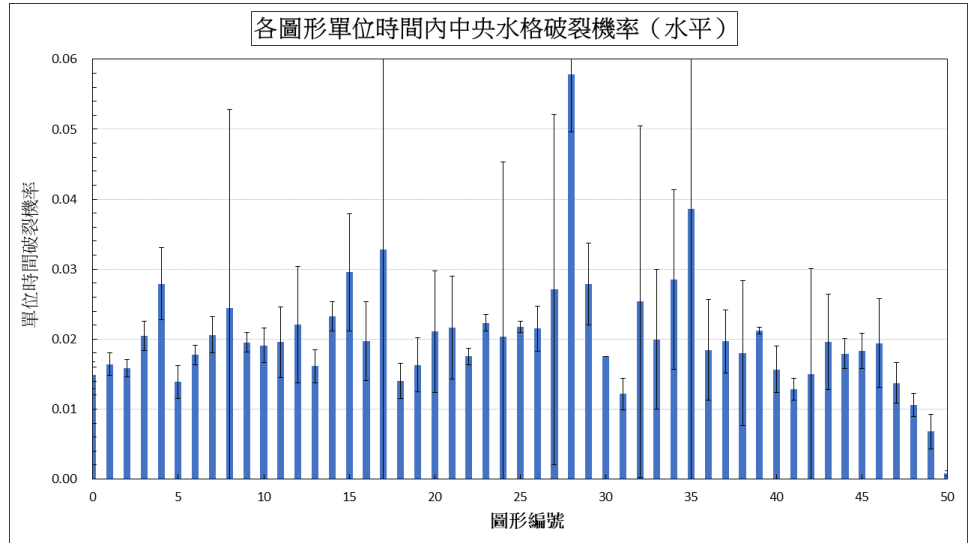
而 SIR 模型與我們實驗的關聯不只這樣，水膜之間有很多奇妙的互動關係，就如破裂時所產生的**表面波**就有如打噴嚏一樣，快速的**傳送能量**至周邊水膜，以致面積過大破裂。

各圖形破裂機率：

由各組別水平實驗組中的 50 種九宮格圖形單位時間破裂機率統計圖，能看出各圖形破裂機率的**不同**。

破裂率較**高**的圖形：
28、35、17、15、34、
4、29、27

破裂率較**低**的圖形：
5、18、31、41、47、
48、49、50。



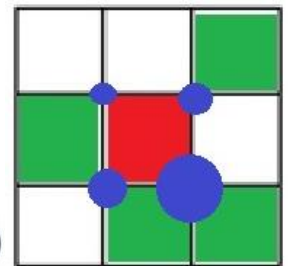
推論：

導致水膜破裂的可能原因有主要三個：**表面波、水分蒸發與水膜形變**影響表面積，此分析主要針對後兩者的影響討論並驗證「**相關係數分析**」中提出的力平衡假說。

首先，觀察破裂機率顯著較低且屬於 $n = 7、8$ 的圖形 47、48、49、50。推測其存留率高的原因包含兩者：

1. 由「各圖形破裂時間」可以得知 $n = 7、8$ 之圖形皆多存在於**實驗初期**，水膜內含水量豐富，尚未蒸散。
2. 若中央水膜產生任何不穩定，直接相鄰的水膜數皆大於 3，能予以**補充水分**。

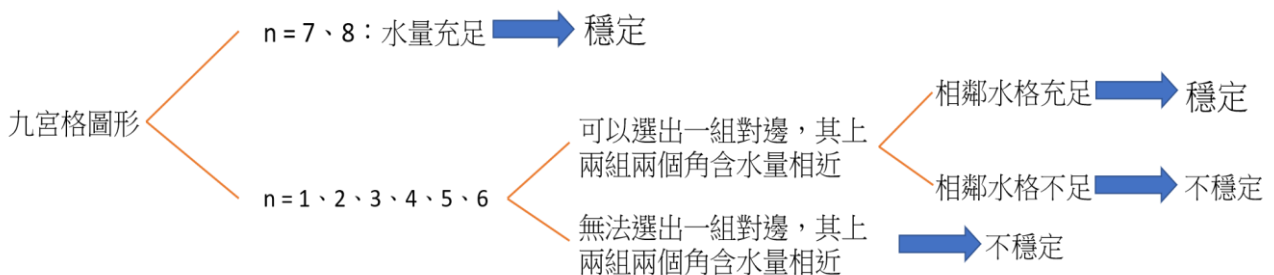
除了水膜水量不足外，水膜的扭曲亦能造成破裂，且扭曲的造成歸因於相鄰水膜之拉力（表面張力）。以圖形 28 為例，當中央水膜四角水量不均勻時，牽動角落之水量分布亦不均勻，水膜即產生形變。若此形變符合某些規律則水膜便容易破裂。此規律於下段第二點解說。



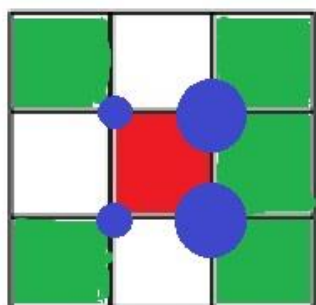
註：角落藍色圓圈大小代表區域推估含水量

經觀察破裂率顯著較高或較低的九宮格圖形，歸納出中央水膜穩定/不穩定存在的條件：

1. 若圖形屬於 $n = 7、8$ ，則因**含水量高**，中央水膜**不易破裂**。
2. 若水膜四個角落能**選擇一組對邊**，其上兩組兩個角各因附近水膜分布產生之表面

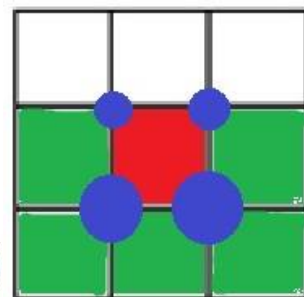


張力相近導致含水量相近，則水膜有機會穩定存在，其先決條件為直接相鄰(非對角)之水膜仍生存，可於不穩定時補充水分。這是因為水膜產生傾斜相對於水膜產生扭曲對水膜穩定性影響較小。



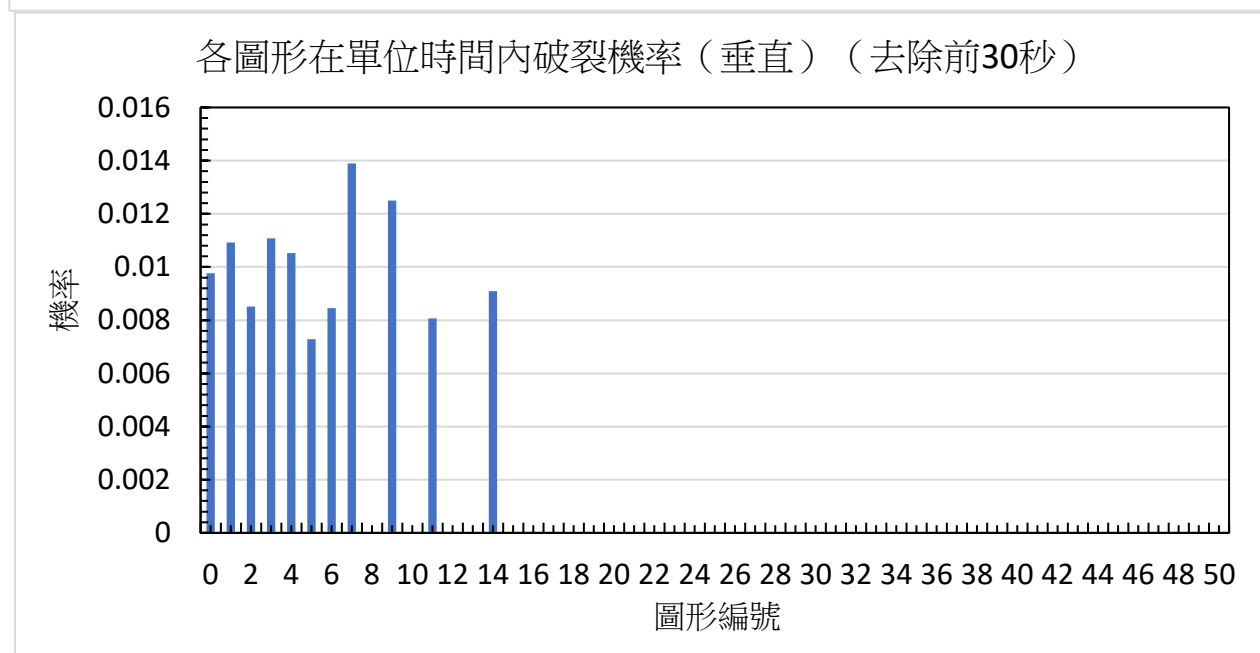
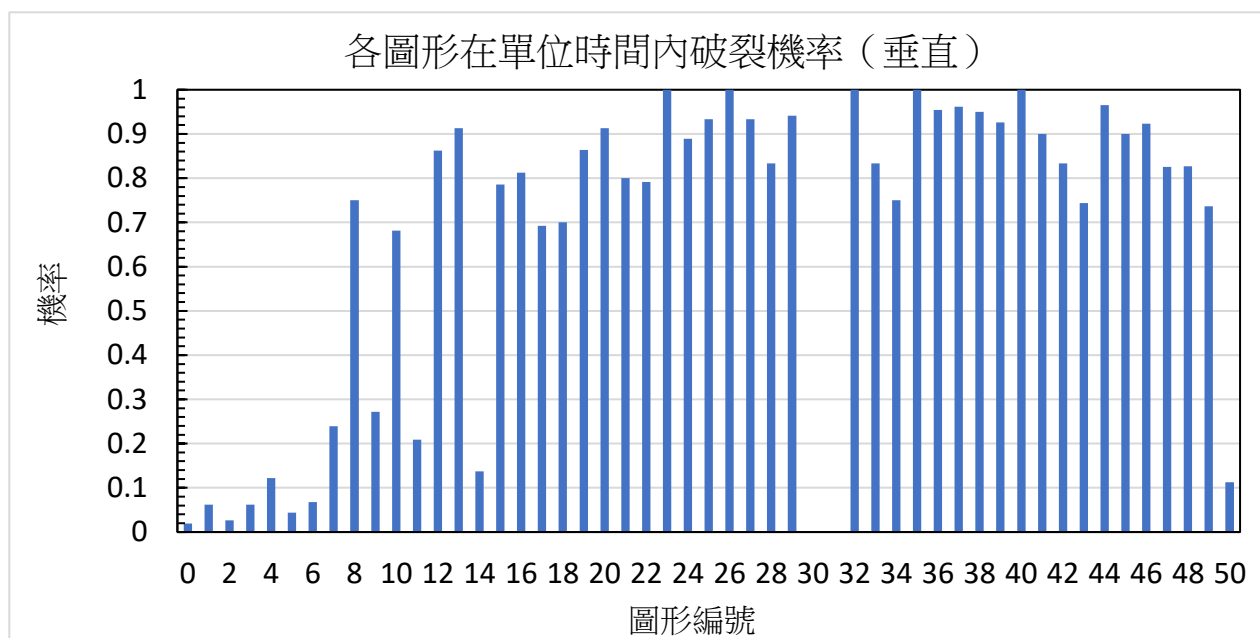
舉例而言，圖形 34 與圖形 41 雖同時滿足「可以選出一組對邊，其上兩組兩個角含水量相近」的條件，但因相鄰可供補充水分格數不同，造成破裂機率的差異。

(34)



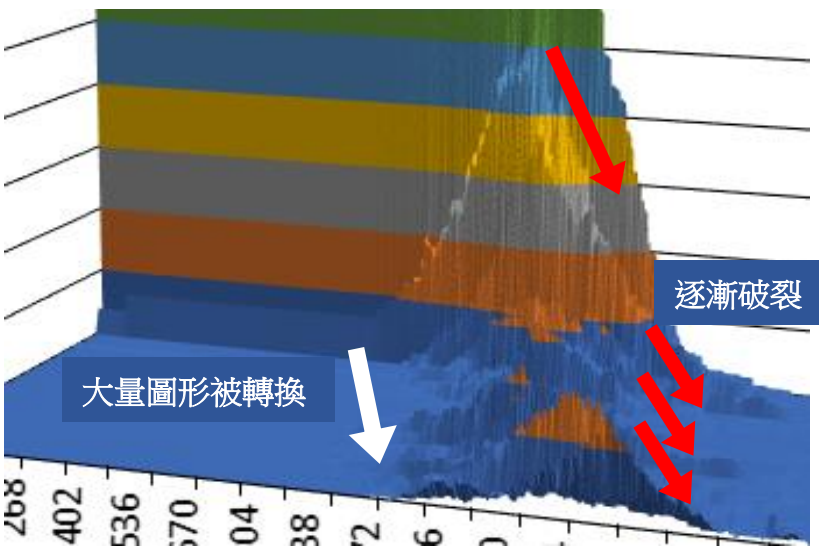
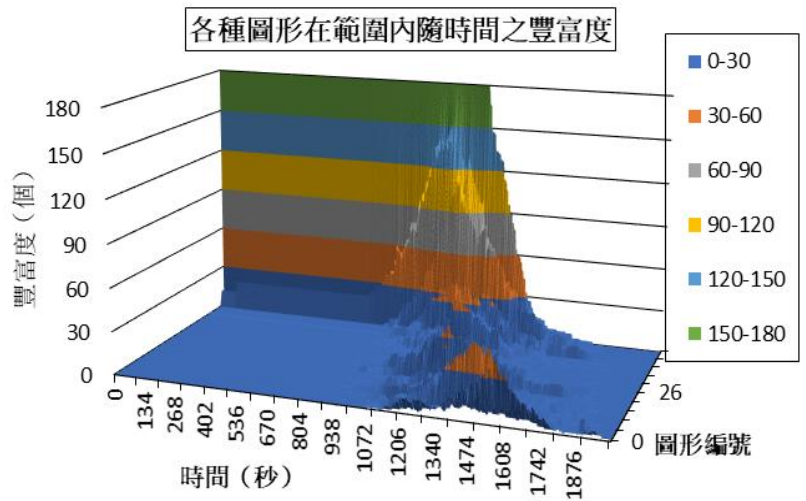
(41)

在鉛直組別中，可以發現破裂機率與水平組別大不相同，且中央水膜破裂機率極高，推測是因為前期紗網上的水流動牽動水膜內水膜破裂，過程並不遵守前段提及的力平衡機制。



圖形出現豐富度時間分布：

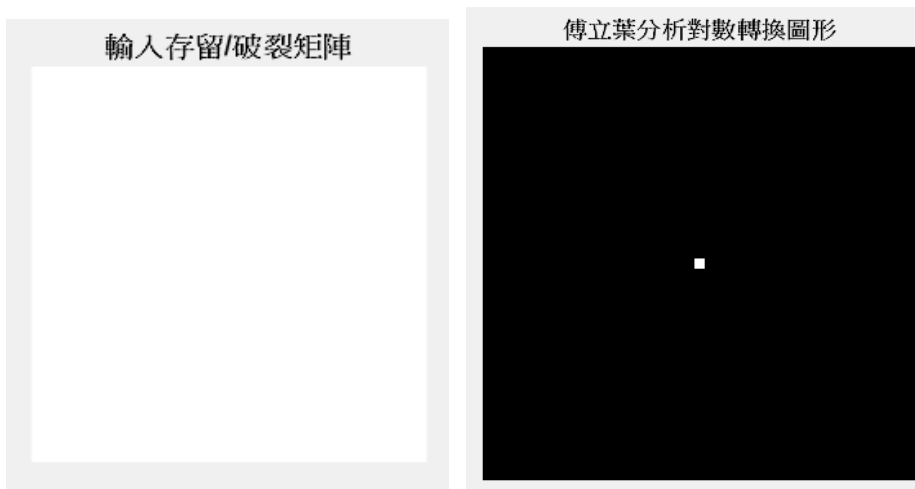
從此圖可以看出，初始狀態下範圍水膜皆處於全滿（shape 50）的狀態。當開始有範圍內水膜破裂後，部分 shape 50 圖形逐漸轉換成其他圖形，同時逐漸破裂形成一個橫向傳遞波峰的趨勢。

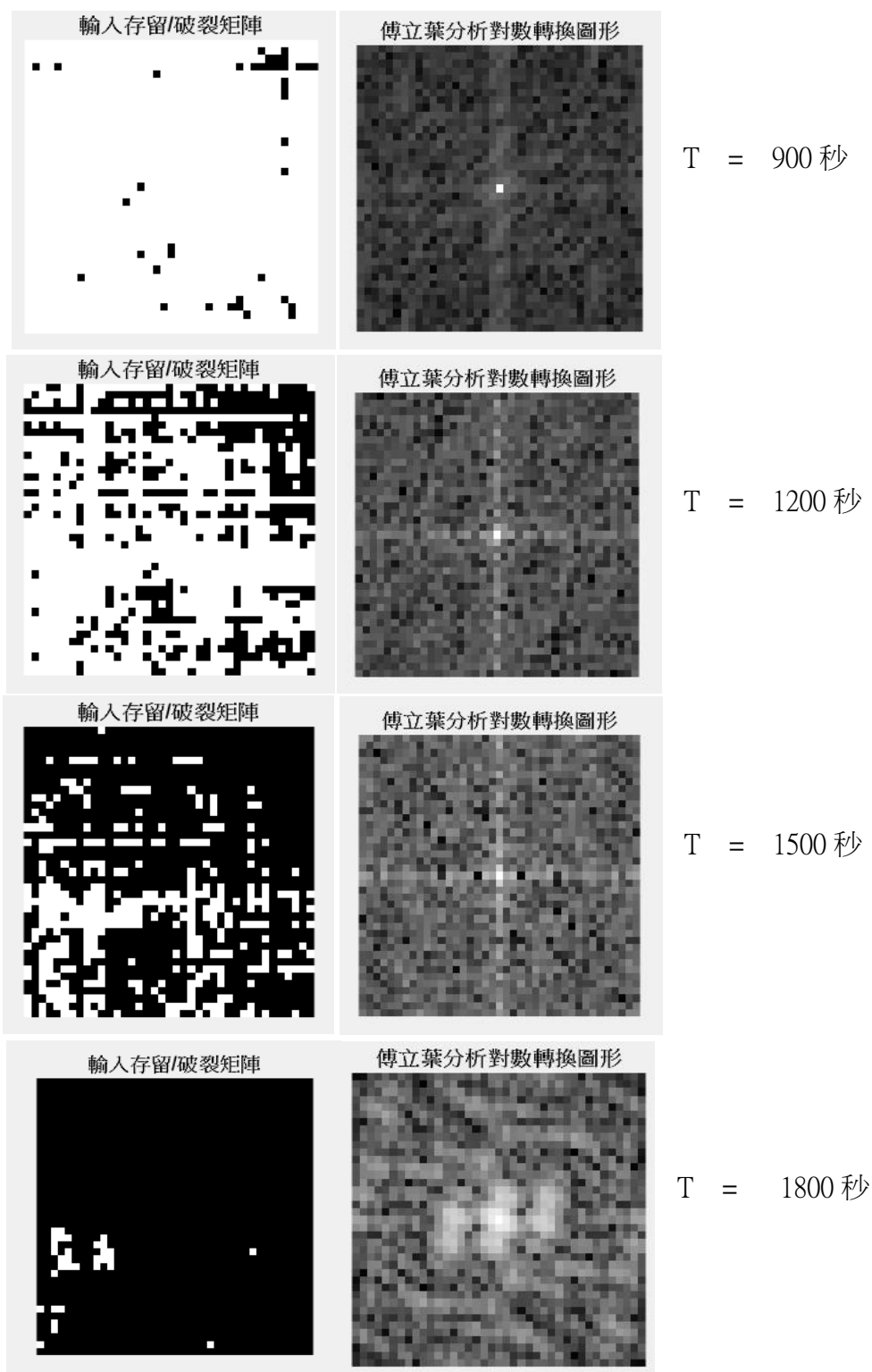


圖表放大圖，可見 shape 50 被迅速消耗以轉換成其他圖形。這些山形區域的時間距也對應到了存留比率快速下降的區域。

D. 傅立葉分析：

針對水平實驗組中 39*39 格的存亡情形，進行不同時間點的二圍傅立葉分析後對數轉換，發現僅兩座標軸出現顯著峰值。





推論：

1. 當破裂水膜數低於總水膜數二分之一時，兩座標軸顯示的峰值對應到許多單一行、列水膜破裂產生的亮度峰值，且無明確週期。兩個方向的破裂關係也無法清楚地以顯著週期波疊合表達。
2. 如 $t = 1800$ 秒的分析所示，當紗窗上僅殘留幾處群聚的水膜，則易因其間隔呈現有幾個亮度峰值的傅立葉分析圖，對比多組實驗仍難找出實質物理意義。

七、結論

A. 水膜行為模式：

1. 紗網上的水分分為水膜本身與水膜以外的水分：
 - i. 在水平組別中上層水分**蒸發**後，水膜才開始破裂。
 - ii. 在鉛直組別中上層水分受重力影響向下流，產生一波流動牽扯部分水膜破裂，形成兩階段式（不同相）的破裂行為。
2. 紗窗水膜**彼此相連**，相鄰之水膜可透過此連結相互傳遞**表面波**及補充水分。
3. 水膜破裂後在四個角形成**四個收縮小水珠**，據推測為水膜水分真正之**基本單位**。

B. 水膜巨觀分析：

1. 相關性分析：

- i. 因相鄰水膜可相互補充水分，水平組別與鉛直組別皆顯示邊相鄰水膜與中央水膜生存時間之相關係數大於對角水膜。
- ii. 因為鉛直組別水膜生存時間包含因**重力流動的水分**牽扯水膜之隨機性，水平組別中鄰近水膜生存時間與中央水膜之相關性**高於鉛直組別**。
- iii. 鉛直組別之存留曲線經分析驗證具有**相點**，分為兩種不同破裂模式，驗證上述相關性分析中的猜測，而較後出現之模式**接近於水平組**。

2. 存留曲線分析：

- i. 定性分析：水膜破裂改變周遭水膜之九宮格圖形，使**破裂機率有所改變**。迅速破裂的水膜使**圖形轉換迅速**，造成水膜更大量快速破裂。
- ii. 定量分析：
 - a、理論（一）指數衰減－圖形轉換同步理論：觀測範圍之水膜破裂主要由**範圍外之水膜破裂**所引發（主要），而蒸發破裂所引發之初始破裂發生較少（次要，隨機）。初始破裂之引發令範圍內之水膜各圖形由其一定之破裂機率行**指數衰減**，配合同步進行之圖形轉換作用，形成所觀測之**存留曲線**。
 - b、理論（二）SIR 模型類比理論：水膜之破裂就好像傳染病一樣，在水膜之間一格格傳染。我們套用傳染病常用的 SIR 模型與水膜存留進行對比，擬合模型中的傳染率並討論水膜傳染之最小單位。我們得到的結果說明，水膜傳染之最小單位為四分之一個水膜，且傳染率之量值可與水膜大規模破裂之情形互相連結。

3. 水膜圖形分析與破裂現象討論：

- i. 水膜破裂為水分蒸發與水膜周遭表面張力各方向**不平衡導致扭曲**所致。
 - ii. $n = 7、8$ 的九宮格圖形之中央水膜因含水量充足不易破裂。
 - iii. 當受力不平衡，但符合「水膜可以選出一組對邊，其上兩組兩個角含水量相近」的條件時，若有相鄰可供給水分之水膜存在，中央水膜亦不易破裂。
4. **傅立葉分析**：隨時間對水膜存亡矩陣進行二維傅立葉分析，發現前期存留的水膜空間上並沒有顯著的週期現象，後期顯示的變化應是**生存水膜群聚**所產生。

八、討論與應用

A. 研究展望：

1. 實驗架設與進行：
 - i. 礙於時間因素，未能利用高速攝影紀錄 51 種九宮格圖形之中央水膜破裂情形。若未來能執行，可進一步驗證力學模型中對破裂機率的解釋。
 - ii. 本研究所有統計數據均以 39*39 格資料組為基準，若能擴大討論範圍，可減少範圍外因素所造成的隨機性。
2. 資料分析與變數定義：
 - i. 若優化現有的 python 程式，應可跳過矩陣轉置步驟直接計算每個水膜的生存時間，大幅提高效率，便於更多實驗數據的累積。
 - ii. 力學模型中的九宮格圖形定義以討論水平狀態為主，因此不同旋轉對稱的圖形會被歸一為同類。探討鉛直組實驗時應加以區分，才能更精準討論物理意義。
 - iii. 若能以更大的數據庫建立各個九宮格圖形準確的破裂函數，應可做出更高「階」的理論函數，更加扣合實驗數據。
3. 可以更多組多數水膜破裂的影格進行交叉比對，即便傅立葉分析無發現直接物理意義，仍可用來排除系統性實驗誤差（例如：初始令水膜全部充水的方式，可能導致單向相接水膜較易破裂，進而影響實驗分析。）

B. 應用與價值：

1. 本研究偏向理論模型的發想與驗證，且現象本身並無法直接應用於生活中。但利用程式分析圖像、進行統計的手法，以及藉由細部現象的分析去討論統計數字背後物理意義的作法，皆有在相關領域的參考價值。
2. 其中水膜破裂模式與傳染病模型之連結實為研究過程中之意外發現，十分有趣且提供原本複雜的計算模型一個相對值觀的定量解釋。

九、參考資料

1. Compartmental models in epidemiology:
https://en.wikipedia.org/wiki/Compartmental_models_in_epidemiology
2. OpenCV-Python Tutorials:
https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_tutorials.html
3. 2-D Fast Fourier Transform:
<https://www.mathworks.com/help/matlab/ref/fft2.html>
4. Correlation Coefficients:
<https://www.mathworks.com/help/matlab/ref/corrcoef.html>

十、附錄

分析程式：

附錄（一）：產生 T 矩陣；python opencv

```
import cv2
import numpy as np
img = cv2.imread("Resources/30-30_1slanted_frame.jpg")
cv2.imshow("org",img)
width, height = 260,260
pts1 = np.float32([[967, 374], [1051, 373], [1051, 477], [963, 478]])
pts2 = np.float32([[0,0],[width,0],[width,height],[0,height]])
matrix = cv2.getPerspectiveTransform(pts1,pts2)
img_warp = cv2.warpPerspective(img,matrix,(width,height))
cv2.imshow("output",img_warp)
img_gray = cv2.cvtColor(img_warp,cv2.COLOR_BGR2GRAY)
cv2.imshow("gray",img_gray)
(thresh,img_WB) = cv2.threshold(img_gray, 130, 255, cv2.THRESH_BINARY)
cv2.imshow("wb",img_WB)
cv2.waitKey(0)
A = np.zeros((13,13),dtype=int)
x = 0 , y = 0 , a = int(width/ 26)
print(a)
b = int(height/ 26)
print(b)
pixel = img_WB[a,b]
print(pixel)
while x <= 12:
    while y <= 12:
        pixel = img_WB[a,b]
        #print(pixel)
        if pixel == 255:
            A[x,y]=0
        else:
            A[x,y]=1
        b =int(b + height / 13)
        y += 1
        #print(A)
    b = int(height/26) , y = 0 , a = int(a + width/13)
    x += 1
```



```

print (A)
import xlswriter
import cv2
import numpy as np
count_a = 1
cap_a = cv2.VideoCapture('Resources/30-30_1slanted_2.mp4')
survive_a = np.zeros((13, 13), dtype=float)
while cap_a.isOpened():
    ret, img_a = cap_a.read()
    if ret == True:
        count_a = count_a + 1
        if count_a % 3 == 0:
            width_a, height_a = 260, 260
            pts1_a = np.float32([[809, 178], [889, 179], [886, 273], [804, 273]])
            pts2_a = np.float32([[0, 0], [width_a, 0], [width_a, height_a], [0, height_a]])
            matrix_a = cv2.getPerspectiveTransform(pts1_a, pts2_a)
            img_warp_a = cv2.warpPerspective(
                img_a, matrix_a, (width_a, height_a))
            img_gray_a = cv2.cvtColor(img_warp_a, cv2.COLOR_BGR2GRAY)
            x_a = 0, y_a = 0, a_a = int(width_a / 26), b_a = int(height_a / 26)
            pixel_a = img_gray_a[a_a, b_a]
            while x_a <= 12:
                while y_a <= 12:
                    pixel_a = img_gray_a[a_a, b_a]
                    if pixel_a <= 132:
                        survive_a[x_a, y_a] = count_a / 30
                        b_a = int(b_a + height_a / 13)
                        y_a += 1
                    b_a = int(height_a / 26), y_a = 0, a_a = int(a_a + width_a / 13)
                    x_a += 1
            else:
                break
        print(int(count_a))
        print(survive_a)
        cap_a.release()
        cv2.destroyAllWindows()
        workbook_a = xlswriter.Workbook('arrays1.xlsx')
        worksheet_a = workbook_a.add_worksheet()
        row = 0
        for col, data in enumerate(survive_a):

```

```

        worksheet_a.write_column(row, col, data)
workbook_a.close()
count_b = 1
cap_b = cv2.VideoCapture('Resources/30-30_1slanted.mp4')
survive_b = np.zeros((13, 13), dtype=float)
while cap_b.isOpened():
    ret, img_b = cap_b.read()
    if ret == True:
        count_b = count_b + 1
        if count_b % 3 == 0:
            width_b, height_b = 260, 260
            pts1_b = np.float32([[804, 273], [886, 273], [882, 372], [796, 371]])
            pts2_b = np.float32([[0, 0], [width_b, 0], [width_b, height_b], [0, height_b]])
            matrix_b = cv2.getPerspectiveTransform(pts1_b, pts2_b)
            img_warp_b = cv2.warpPerspective(
                img_b, matrix_b, (width_b, height_b))
            img_gray_b = cv2.cvtColor(img_warp_b, cv2.COLOR_BGR2GRAY)
            x_b = 0, y_b = 0, a_b = int(width_b / 26), b_b = int(height_b / 26)
            pixel_b = img_gray_b[a_b, b_b]
            while x_b <= 12:
                while y_b <= 12:
                    pixel_b = img_gray_b[a_b, b_b]
                    if pixel_b <= 132:
                        survive_b[x_b, y_b] = count_b / 30
                        b_b = int(b_b + height_b / 13)
                        y_b += 1
                    b_b = int(height_b / 26), y_b = 0, a_b = int(a_b + width_b / 13)
                    x_b += 1
            else:
                break
        print(int(count_b))
        print(survive_b)
        cap_b.release()
        cv2.destroyAllWindows()
        workbook_b = xlsxwriter.Workbook('arrays2.xlsx')
        worksheet_b = workbook_b.add_worksheet()
        row = 0

for col, data in enumerate(survive_b):
    worksheet_b.write_column(row, col, data)

```

```

workbook_b.close()
count_c = 1
cap_c = cv2.VideoCapture('Resources/30-30_1slanted.mp4')
survive_c = np.zeros((13, 13), dtype=float)
while cap_c.isOpened():
    ret, img_c = cap_c.read()
    if ret == True:
        count_c = count_c + 1
        if count_c % 3 == 0:
            width_c, height_c = 260, 260
            pts1_c = np.float32([[796, 371], [882, 372], [876, 477], [787, 476]])
            pts2_c = np.float32([[0, 0], [width_c, 0], [width_c, height_c], [0, height_c]])
            matrix_c = cv2.getPerspectiveTransform(pts1_c, pts2_c)
            img_warp_c = cv2.warpPerspective(
                img_c, matrix_c, (width_c, height_c))
            img_gray_c = cv2.cvtColor(img_warp_c, cv2.COLOR_BGR2GRAY)
            x_c = 0, y_c = 0, a_c = int(width_c / 26), b_c = int(height_c / 26)
            pixel_c = img_gray_c[a_c, b_c]
            while x_c <= 12:
                while y_c <= 12:
                    pixel_c = img_gray_c[a_c, b_c]
                    if pixel_c <= 128:
                        survive_c[x_c, y_c] = count_c / 30
                        b_c = int(b_c + height_c / 13)
                        y_c += 1
                    b_c = int(height_c / 26), y_c = 0, a_c = int(a_c + width_c / 13)
                    x_c += 1
            else:
                break
        print(int(count_c))
        print(survive_c)
    cap_c.release()
cv2.destroyAllWindows()
workbook_c = xlsxwriter.Workbook('arrays3.xlsx')
worksheet_c = workbook_c.add_worksheet()
row = 0
for col, data in enumerate(survive_c):
    worksheet_c.write_column(row, col, data)
workbook_c.close()

```

附錄 (二) : 存留比率曲線 ; C++

```
#include <iostream>
#include <fstream>
#include <string>
#include <sstream>
using namespace std;int norm = 39;
int main(int argc, char const *argv[]){string file = "C:\\Users\\tengw\\Desktop\\special39.csv";cout << file
<< endl;ifstream input(file.c_str());string line;int b[norm][norm], m = 0, max =
0;while(!input.eof()){getline(input, line);if(!input.eof()){int pos2 = -1; for(int i = 0;i<norm;i++){if(i <
norm - 1){int pos = line.find(','); string x = line.substr(pos2 + 1 , pos - pos2 - 1);    stringstream ss;ss <<
x;ss >> b[m][i];line[pos] = (char) 32;pos2 = pos; }else{string y = line.substr(pos2 + 1);stringstream ss;ss
<< y;ss >> b[m][norm - 1];} if(b[m][i] > max){max = b[m][i];} }cout << line << endl;m++;} }ofstream
outData;outData.open("SR.csv", ios::app);    outData << "Time" << "," << "Survival Rate" << endl;cout
<< max << endl;int k = 0;bool yes = true; while(yes){int n = 0;for(int i = 1;i < norm - 1;i++){for(int j = 1;j
< norm - 1;j++){if(b[i][j] >= k/8){n++;}} }outData << (double) k/8 << "," << (double) n/((norm -
2)*(norm - 2)) << "," << n << endl;k++;  if(n == 0){yes = false;} } input.close(); return 0;}
```

附錄(三) : 相關性分析 ; matlab

```
M = xlsread('30-30_2vertical_0.1.xlsx', '39-39');A = zeros(13);Rbase = M(7:33,7:33);
Rbase2=zeros(39);Rvary2=zeros(39;x=1;y=1;pickx=1;picky=1;
while x<=13
    while y<=13
        Rvary=M(x:x+26,y:y+26);
        while pickx<=27
            while picky<=27
                matrixcor=M(pickx:pickx+12, picky:picky+12);
                Avg=sum(sum(matrixcor))/169;
                Rvary2(pickx,picky)=Rvary(pickx,picky)-Avg;
                Rbase2(pickx,picky)=Rbase(pickx,picky)-Avg;
                picky = picky+1;
            end
            picky=1;pickx=pickx+1;
        end
        r=corrcoef(Rbase2, Rvary2);Rbase2=zeros(39);Rvary2=zeros(39);
        A(x,y)=r(1,2); y=y+1; pickx=1;picky=1;
    end
    y=1; x=x+1;
end
```

附錄(四) : 九宮格圖形豐富度 ; C++

```

#include <iostream>
#include <fstream>
#include <string>
#include <sstream>
#include <iomanip>
using namespace std;int norm = 39;
int determine(int num, int in[8]){int count = 0;while(in[0] == 0 && count < 8){int media = 0;int
medium;for(int i = 0;i < 8;i++){medium = in[7 - i];in[7 - i] = media;media = medium;}count++;}if(num
== 0){return 0; }else if(num == 1){if(count % 2 == 0){return 1;}else{return 2;}}else if(num ==
2){if(in[1] + in[7] > 0){return 3;}else if(in[2] + in[6] > 0){if(count % 2 == 0){return 4;}else{return
6;}}else if(in[3] + in[5] > 0){return 7;}else{if(count % 2 == 0){return 8;}else{return 5;}}}}else if(num ==
3){if(in[2] + in[4] + in[6] > 1){if(count % 2 == 0){return 15;}else{return 16;}}else if(in[2] + in[3] == 1
&& in[6] + in[5] == 1){if(in[2] + in[6] == 1){if(count % 2 == 0){return 17;}else{return
18;}}else{if(count % 2 == 0){return 18;}else{return 17;}}}}else if(in[1] + in[2] == 2||in[7] + in[6] ==
2){if(count % 2 == 0){return 9;}else{return 10;}}else if(in[1] + in[7] == 2){return 10;}else if(in[1] + in[7]
== 0){if(in[2] + in[6] == 1){if(count % 2 == 0){return 14;}else{return 11;}}else{if(count % 2 ==
0){return 12;}else{return 13;}}}}else if(in[4] > 0){if(count % 2 == 0){return 12;}else{return 13;}}else
if(in[1] + in[5] == 2||in[7] + in[3] == 2){if(count % 2 == 0){return 13;}else{return 12;}}else{if(in[2] +
in[6] == 1){if(count % 2 == 0){return 14; }else{return 11;}}else{if(count % 2 == 0){return
11;}else{return 14;}}}}else if(num == 4){if(in[2] + in[4] + in[6] == 3){if(count % 2 == 0){return
30;}else{return 31;}}else if(in[4] == 1 && (in[1] + in[5]) % 2 == 0 && in[2] + in[6] == 0){return
25;}else if(in[7] + in[6] == 2 && in[2] + in[3] + in[4] == 0){return 19;}else if(in[1] + in[2] == 2 &&
in[6] + in[5] + in[4] == 0){return 19;}else if(in[1] + in[7] == 1 && in[2] + in[6] == 0 && in[4] ==
1){if(count % 2 == 0){return 24;}else{return 26;}}else if(in[1] + in[7] == 1 && in[1] + in[2] == 1 &&
in[7] + in[6] == 1 && (in[1] + in[5]) % 2 == 0 && (in[3] + in[7]) % 2 == 0){ if(count % 2 == 0){return
26;}else{return 24;}}else if(in[1] + in[7] == 0 && in[2] + in[6] == 0){if(count % 2 == 0){return
23;}else{return 21;}}else if(in[1] + in[2] != 1 && in[6] + in[7] != 1 && in[3] + in[5] == 1){if(count % 2
== 0){return 21;}else{return 23;}}else if(in[2] + in[6] + in[3] + in[5] == 0){return 23;}else if(in[2] + in[6]
+ in[4] == 0 && in[3] + in[5] == 1){return 22;}else if(in[1] + in[2] == 2||in[2] + in[3] + in[4] == 3||in[5]
+ in[6] + in[4] == 3||in[6] + in[7] == 2){if(count % 2 == 0){return 20;}else{return 22;}}else if(in[4] ==
1||in[1] + in[5] == 2||in[2] + in[6]== 2||in[3] + in[7] == 2){if(in[1] + in[7] == 0){if(count % 2 == 0){return
29;}else{return 27;}}else{if(in[4] == 1){if(count % 2 == 0){return 29;}else{return 27;}}else{if(count % 2
== 0){return 27;}else{return 29;}}}}}}else{return 28;}}}}
int opposite(int dig, int ob[8]){for(int i = 0;i < 8;i++){if(ob[i] == 0){ob[i] = 1;}else{ob[i] = 0;}}return 50 -
determine(8 - dig, ob);}
int main(int argc, char const *argv[]){string filename = "C:\\Users\\tengw\\Desktop\\flat3.csv";ifstream
input(filename.c_str());string line;int a[norm][norm];int m = 0, max = 0;while(!input.eof()){getline(input,
line);if(!input.eof()) {int pos2 = -1; for(int i = 0;i < norm;i++){ if(i < norm - 1){int pos = line.find(',');string
x = line.substr(pos2 + 1 , pos - pos2 - 1);stringstream ss;ss << x;ss >> a[m][i];line[pos] = (char) 32;pos2 =
pos;}else{string y = line.substr(pos2 + 1);stringstream ss;ss << y;ss >> a[m][norm - 1];if(a[m][i] >

```



```

max){max = a[m][i];}cout<<endl;cout<<endl;cout<<endl;ofstream
outData;outData.open("shapes over time.csv", ios::app);outData << ",";for(int i = 0;i < 51;i++){outData
<< i << ",";outData << endl; for(int k = 0;k < max;k++){int l = 0;int f[51] = {0};for(int i = 1;i < norm -
1;i++){for(int j = 1;j < norm - 1;j++){int b[8] = {0};int x = a[i][j];int n = 0;if(k <= x){if(a[i - 1][j - 1] >
k){n++;b[0]++; }if(a[i - 1][j] > k){n++;b[1]++; }if(a[i - 1][j + 1] > k){n++;b[2]++;} if(a[i][j - 1] >
k){n++;b[7]++; }if(a[i][j + 1] > k){n++;b[3]++;}if(a[i + 1][j - 1] > k){n++;b[6]++;} if(a[i + 1][j] >
k){n++;b[5]++; }if(a[i + 1][j + 1] > k){n++;b[4]++;}l++;if(n <
5){f[determine(n,b)]++;}else{f[opposite(n,b)]++;}} }outData << k << ",";for(int i = 0;i < 51;i++){outData
<< f[i] << ","; } outData << endl;}input.close();return 0;}

```

附錄(五)：九宮格圖形破裂時間；C++

```

#include <iostream>
#include <fstream>
#include <string>
#include <sstream>
#include <iomanip>
using namespace std;int norm = 39;
int determine(int num, int in[8]){int count = 0;while(in[0] == 0 && count < 8){int media = 0;int
medium;for(int i = 0;i < 8;i++){medium = in[7 - i];in[7 - i] = media;media = medium;}count++;}if(num
== 0){return 0; }else if(num == 1){if(count % 2 == 0){return 1;}else{return 2; }}else if(num ==
2){if(in[1] + in[7] > 0){return 3;}else if(in[2] + in[6] > 0){if(count % 2 == 0){return 4;}else{return
6;}}else if(in[3] + in[5] > 0){return 7;}else{if(count % 2 == 0){return 8;}else{return 5;}} }else if(num ==
3){if(in[2] + in[4] + in[6] > 1){if(count % 2 == 0){return 15;}else{return 16;}}else if(in[2] + in[3] == 1
&& in[6] + in[5] == 1){if(in[2] + in[6] == 1){if(count % 2 == 0){return 17;}else{return
18;}}else{if(count % 2 == 0){return 18;}else{return 17;}} }else if(in[1] + in[2] == 2||in[7] + in[6] ==
2){if(count % 2 == 0){return 9;}else{return 10;}}else if(in[1] + in[7] == 2){return 10;}else if(in[1] + in[7]
== 0){if(in[2] + in[6] == 1){if(count % 2 == 0){return 14;}else{return 11;}}else{if(count % 2 ==
0){return 12;}else{return 13;}} }else if(in[4] > 0){if(count % 2 == 0){return 12;}else{return 13;}}else
if(in[1] + in[5] == 2||in[7] + in[3] == 2){if(count % 2 == 0){return 13;}else{return 12;}}else{if(in[2] +
in[6] == 1){if(count % 2 == 0){return 14; }else{return 11;}}else{if(count % 2 == 0){return
11; }else{return 14;}} } }else if(num == 4){if(in[2] + in[4] + in[6] == 3){if(count % 2 == 0){return
30;}else{return 31;}}else if(in[4] == 1 && (in[1] + in[5]) % 2 == 0 && in[2] + in[6] == 0){return
25;}else if(in[7] + in[6] == 2 && in[2] + in[3] + in[4] == 0){return 19;}else if(in[1] + in[2] == 2 &&
in[6] + in[5] + in[4] == 0){return 19;}else if(in[1] + in[7] == 1 && in[2] + in[6] == 0 && in[4] ==
1){if(count % 2 == 0){return 24;}else{return 26;}}else if(in[1] + in[7] == 1 && in[1] + in[2] == 1 &&
in[7] + in[6] == 1 && (in[1] + in[5]) % 2 == 0 && (in[3] + in[7]) % 2 == 0){ if(count % 2 == 0){return
26;}else{return 24;}}else if(in[1] + in[7] == 0 && in[2] + in[6] == 0){if(count % 2 == 0){return
23;}else{return 21;}}else if(in[1] + in[2] != 1 && in[6] + in[7] != 1 && in[3] + in[5] == 1){if(count % 2
== 0){return 21;}else{return 23;}}else if(in[2] + in[6] + in[3] + in[5] == 0){return 23;}else if(in[2] + in[6]
+ in[4] == 0 && in[3] + in[5] == 1){return 22;}else if(in[1] + in[2] == 2||in[2] + in[3] + in[4] == 3||in[5]

```

```

+ in[6] + in[4] == 3||in[6] + in[7] == 2){if(count % 2 == 0){return 20;}else{return 22;}}else if(in[4] ==
1||in[1] + in[5] == 2||in[2] + in[6]== 2||in[3] + in[7] == 2){if(in[1] + in[7] == 0){if(count % 2 == 0){return
29;}else{return 27;}}else{if(in[4] == 1){if(count % 2 == 0){return 29;}else{return 27;}}else{if(count % 2
== 0){return 27;}else{return 29;}}}}else{return 28;}}
int opposite(int dig, int ob[8]){for(int i = 0;i < 8;i++){if(ob[i] == 0){ob[i] = 1;}else{ob[i] = 0;}}return 50 -
determine(8 - dig, ob);}
int main(int argc, char const *argv[]){string filename = "C:\\Users\\tengw\\Desktop\\flat3.csv";ifstream
input(filename.c_str()); string line; int a[norm][norm];int m = 0, max = 0;while(!input.eof()){getline(input,
line);if(!input.eof()){int pos2 = -1;for(int i = 0;i < norm;i++){if(i < norm - 1){int pos = line.find(',');string
x = line.substr(pos2 + 1, pos - pos2 - 1);stringstream ss;ss << x;ss >> a[m][i]; line[pos] = (char) 32;pos2
= pos;}else{string y = line.substr(pos2 + 1);stringstream ss;ss << y;ss >> a[m][norm - 1]; if(a[m][i] >
max){max = a[m][i];}}cout<<line<<endl;m++;}} cout<<max<<endl; ofstream
outData;outData.open("rupture shape over time.csv", ios::app);outData << ",";for(int i = 0;i <
51;i++){outData << i << ",";outData << endl; for(int k = 0;k < max;k++){int l = 0;int f[51] = {0}; for(int
i = 1;i < norm - 1;i++){for(int j = 1;j < norm - 1;j++){int b[8] = {0};int x = a[i][j];int n = 0;if(k ==
x){if(a[i - 1][j - 1] > k){n++;b[0]++; }if(a[i - 1][j] > k){n++;b[1]++; }if(a[i - 1][j + 1] >
k){n++;b[2]++;}if(a[i][j - 1] > k){n++;b[7]++;}if(a[i][j + 1] > k){n++;b[3]++;}if(a[i + 1][j - 1] >
k){n++;b[6]++; }if(a[i + 1][j] > k){n++;b[5]++;}if(a[i + 1][j + 1] > k){n++;b[4]++;}l++;if(n <
5){f[determine(n,b)]++;}else{f[opposite(n,b)]++;}} } }outData << k << ",";for(int i = 0;i < 51;i++){outData
<< f[i] << ","; } outData << endl;}input.close(); return 0;}

```

附錄(六)：九宮格圖形破裂機率；C++

```

#include <iostream>
#include <fstream>
#include <string>
#include <sstream>
using namespace std;int norm = 39;
int determine(int num, int in[8]){int count = 0;while(in[0] == 0 && count < 8){int media = 0;int
medium;for(int i = 0;i < 8;i++){medium = in[7 - i];in[7 - i] = media;media = medium;}count++;}if(num
== 0){return 0; }else if(num == 1){if(count % 2 == 0){return 1;}else{return 2;}}else if(num ==
2){if(in[1] + in[7] > 0){return 3;}else if(in[2] + in[6] > 0){if(count % 2 == 0){return 4;}else{return
6;}}else if(in[3] + in[5] > 0){return 7;}else{if(count % 2 == 0){return 8;}else{return 5;}}}}else if(num ==
3){if(in[2] + in[4] + in[6] > 1){if(count % 2 == 0){return 15;}else{return 16;}}else if(in[2] + in[3] == 1
&& in[6] + in[5] == 1){if(in[2] + in[6] == 1){if(count % 2 == 0){return 17;}else{return
18;}}else{if(count % 2 == 0){return 18;}else{return 17;}}}}else if(in[1] + in[2] == 2||in[7] + in[6] ==
2){if(count % 2 == 0){return 9;}else{return 10;}}else if(in[1] + in[7] == 2){return 10;}else if(in[1] + in[7]
== 0){if(in[2] + in[6] == 1){if(count % 2 == 0){return 14;}else{return 11;}}else{if(count % 2 ==
0){return 12;}else{return 13;}} }else if(in[4] > 0){if(count % 2 == 0){return 12;}else{return 13;}}else
if(in[1] + in[5] == 2||in[7] + in[3] == 2){if(count % 2 == 0){return 13;}else{return 12;}}else{if(in[2] +
in[6] == 1){if(count % 2 == 0){return 14;}else{return 11;}}else{if(count % 2 == 0){return

```

```

11;}else{return 14;}}}}else if(num == 4){if(in[2] + in[4] + in[6] == 3){if(count % 2 == 0){return
30;}else{return 31;}}else if(in[4] == 1 && (in[1] + in[5]) % 2 == 0 && in[2] + in[6] == 0){return
25;}else if(in[7] + in[6] == 2 && in[2] + in[3] + in[4] == 0){return 19;}else if(in[1] + in[2] == 2 &&
in[6] + in[5] + in[4] == 0){return 19;}else if(in[1] + in[7] == 1 && in[2] + in[6] == 0 && in[4] ==
1){if(count % 2 == 0){return 24;}else{return 26;}}else if(in[1] + in[7] == 1 && in[1] + in[2] == 1 &&
in[7] + in[6] == 1 && (in[1] + in[5]) % 2 == 0 && (in[3] + in[7]) % 2 == 0){ if(count % 2 == 0){return
26;}else{return 24;}}else if(in[1] + in[7] == 0 && in[2] + in[6] == 0){if(count % 2 == 0){return
23;}else{return 21;}}else if(in[1] + in[2] != 1 && in[6] + in[7] != 1 && in[3] + in[5] == 1){if(count % 2
== 0){return 21;}else{return 23;}}else if(in[2] + in[6] + in[3] + in[5] == 0){return 23;}else if(in[2] + in[6]
+ in[4] == 0 && in[3] + in[5] == 1){return 22;}else if(in[1] + in[2] == 2||in[2] + in[3] + in[4] == 3||in[5]
+ in[6] + in[4] == 3||in[6] + in[7] == 2){if(count % 2 == 0){return 20;}else{return 22;}}else if(in[4] ==
1||in[1] + in[5] == 2||in[2] + in[6]== 2||in[3] + in[7] == 2){if(in[1] + in[7] == 0){if(count % 2 == 0){return
29;}else{return 27;}}else{if(in[4] == 1){if(count % 2 == 0){return 29;}else{return 27;}}else{if(count % 2
== 0){return 27;}else{return 29;}}}}}}else{return 28;}}
int opposite(int dig, int ob[8]){for(int i = 0;i < 8;i++){if(ob[i] == 0){ob[i] = 1;}else{ob[i] = 0;}}return 50 -
determine(8 - dig, ob);}
int main(int argc, char const *argv[]){string filename = "C:\\Users\\tengw\\Desktop\\ver1.csv";ifstream
input(filename.c_str());string line;int a[norm][norm];int m = 0, max = 0;while(!input.eof()){getline(input,
line);if(!input.eof()) {int pos2 = -1;for(int i = 0;i < norm;i++){if(i < norm - 1){ int pos = line.find(',');string
x = line.substr(pos2 + 1 , pos - pos2 - 1);stringstream ss;ss << x;ss >> a[m][i];line[pos] = (char) 32;pos2 =
pos;}else{ string y = line.substr(pos2 + 1);stringstream ss;ss << y;ss >> a[m][norm - 1];} if(a[m][i] >
max){max = a[m][i];} m++;}}ofstream outData;outData.open("Cabundance.csv", ios::app);outData << ",
<< "Abundance" << ", " << "Rupture Occurence" << ", " << "Corrected prob" << endl;int g[51];for(int i =
0;i < 51;i++){g[i] = 0;}int o[51] = {0}; int e[norm - 2][norm - 2];for(int i = 1;i < norm - 1;i++){for(int j =
1;j < norm - 1;j++){int f[51] = {0};bool hi = true;int k = a[i][j];e[i - 1][j - 1] = -1;while(hi){int b[8] =
{0};int n = 0;if(a[i - 1][j - 1] > k){b[0] = 1;n++;}if(a[i - 1][j] > k){b[1] = 1;n++;}if(a[i - 1][j + 1] >
k){b[2] = 1;n++;}if(a[i][j - 1] > k){b[7] = 1;n++;}if(a[i][j + 1] > k){b[3] = 1;n++;}if(a[i + 1][j - 1] >
k){b[6] = 1;n++;}if(a[i + 1][j] > k){b[5] = 1;n++;}if(a[i + 1][j + 1] > k){b[4] = 1;n++;}if(n < 5){int xyz =
determine(n,b); if(e[i - 1][j - 1] == xyz){f[xyz]++;}else if(e[i - 1][j - 1] == -1){f[xyz]++;e[i - 1][j - 1] =
xyz;}else{hi = false; }}else{int xyz = opposite(n,b); if(e[i - 1][j - 1] == xyz){f[xyz]++;e[i - 1][j - 1] =
xyz;}else if(e[i - 1][j - 1] == -1){f[xyz]++;e[i - 1][j - 1] = xyz;}else{hi = false; }}k--;if(k < 0){ hi =
false;}}g[e[i - 1][j - 1]] += f[e[i - 1][j - 1]];int x = a[i][j];int p = 0;int h[8] = {0};if(a[i - 1][j - 1] > x){h[0]
= 1;p++;} if(a[i - 1][j] > x){h[1] = 1;p++; }if(a[i - 1][j + 1] > x){h[2] = 1;p++;}if(a[i][j - 1] > x){h[7] =
1;p++;}if(a[i][j + 1] > x){ h[3] = 1;p++;}if(a[i + 1][j - 1] > x){ h[6] = 1;p++;}if(a[i + 1][j] > x){h[5] =
1;p++;}if(a[i + 1][j + 1] > x){h[4] = 1;p++;}if(p <
5){o[determine(p,h)]++;}else{o[opposite(p,h)]++;}}for(int i = 0;i < 51;i++){outData << i << ", " << g[i]
<< ", " << o[i] << ", " << (double) o[i] / g[i] << endl; cout << i << ", " << g[i] << endl;} input.close();return
0;}

```

附錄(七)：二維傅立葉分析；matlab

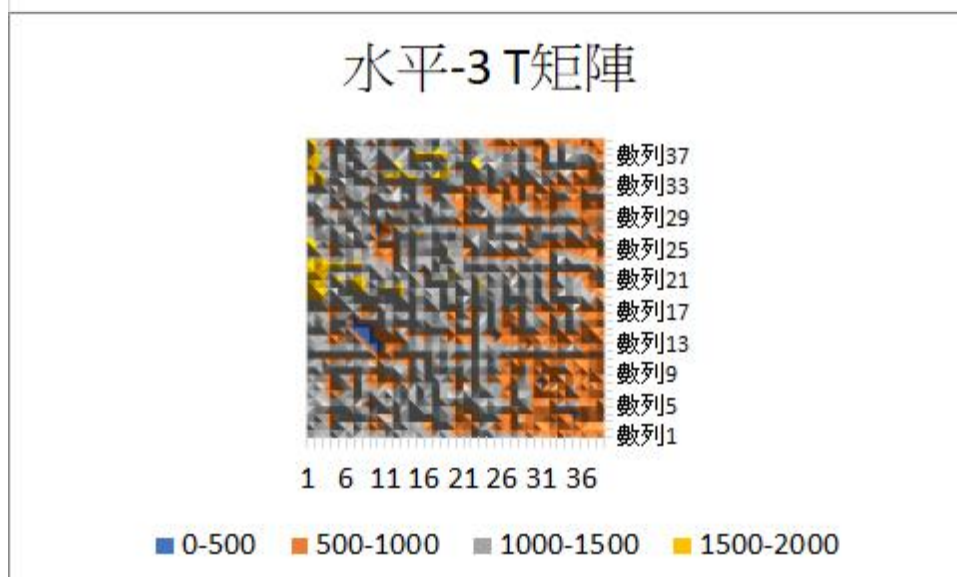
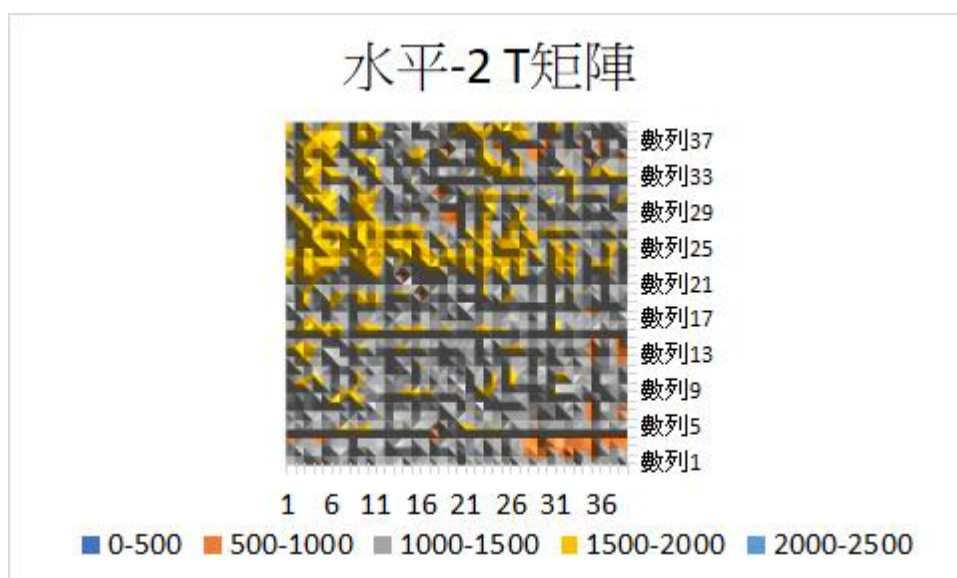
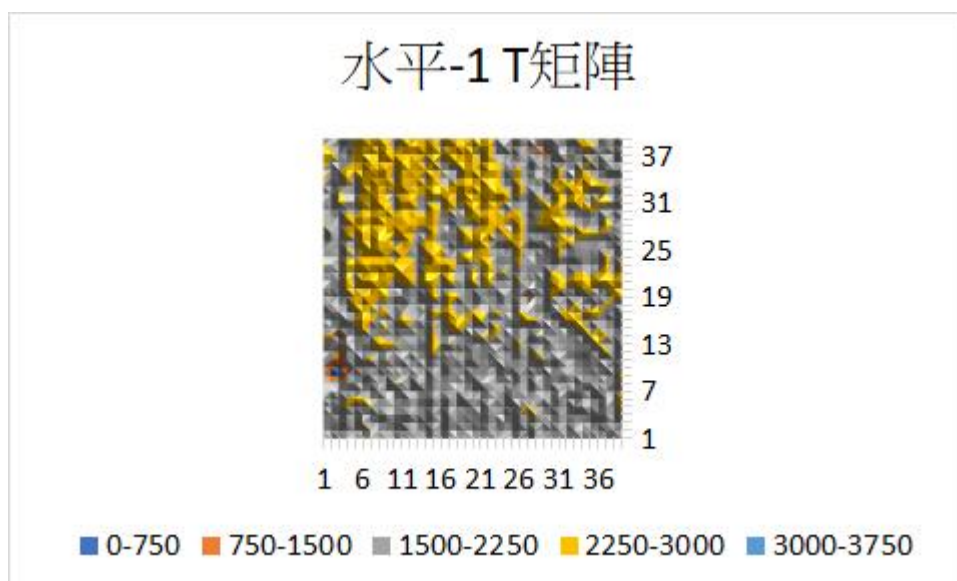
```

matrixbase = xlsread('30-30_2flat_0.5.xlsx', '39-39');x=1;y=1;t=0;M=zeros(39);
maximum=max(max(matrixbase));
while t<= maximum
    while x<=39
        while y<=39
            if matrixbase(x,y) <t
                M(x,y)=0;
            else
                M(x,y)=1;
            end
            y=y+1;
        end
        y=1; x=x+1;
    end
    filename= ['survive',num2str(t),'.xlsx'];xlswrite(filename, M);t=t+300;x=1; y=1;
end
clear all; close all; clc
matrix = xlsread('30-30_2flat_0.5.xlsx', 'test1800');
F = fft2(matrix);
% Fourier transform of an image
S = abs(F);
figure(1);imshow(S,[]);title('Fourier transform of an image');
%get the centered spectrum
Fsh = fftshift(F);
figure(2);imshow(abs(Fsh),[]);title('Centered fourier transform of Image')
%apply log transform
S2 = log(1+abs(Fsh));figure(3);imshow(S2,[]);title('傅立葉分析對數轉換圖形')
%reconstruct the Image
F = ifftshift(Fsh);f = ifft2(F);figure(4);imshow(f,[]);title('輸入存留/破裂矩陣')

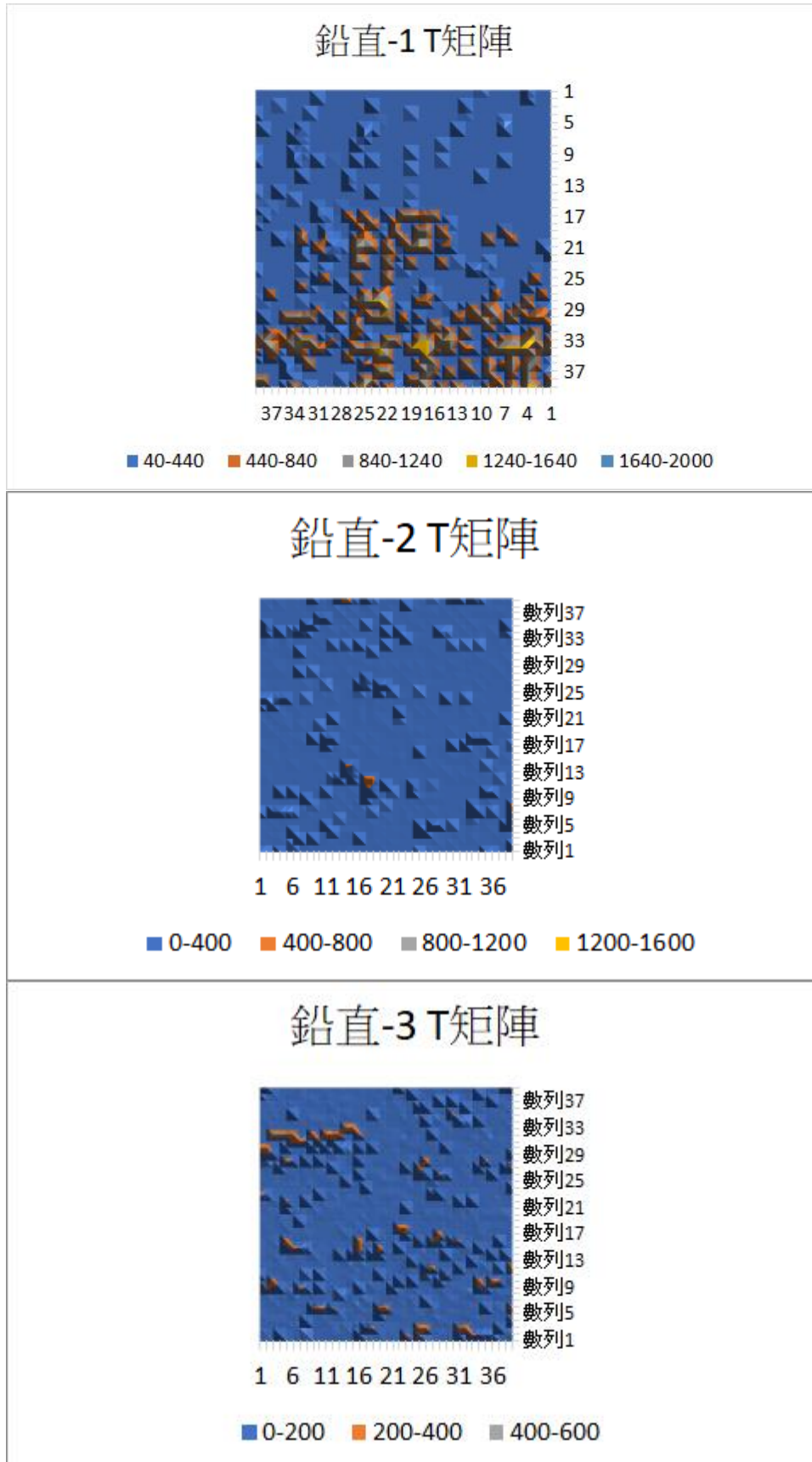
```

實驗數據：

T 矩陣—水平：

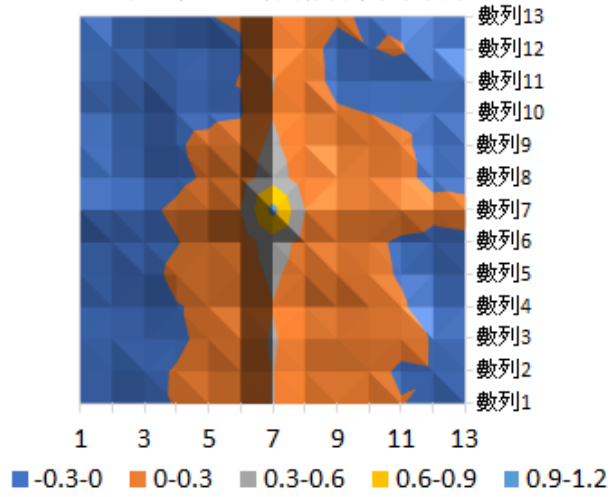


T 矩陣—鉛直：

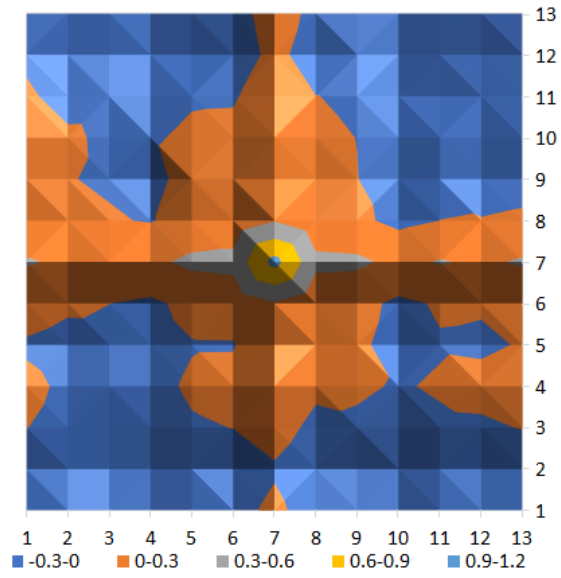


相關係數矩陣—水平：

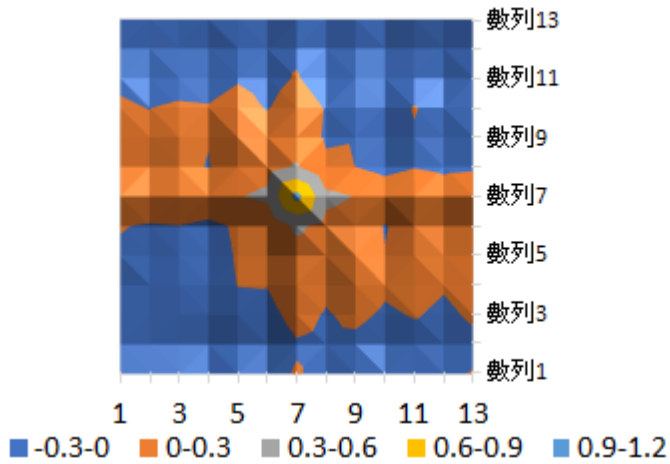
水平-1 相關係數值



水平-2 相關係數值



水平-3 相關係數值



相關係數矩陣—鉛直：

