1. **海岸環境變遷歷程及影像判識分析**

本項工作需標記出計畫區內現有海岸結構物與自然河道、沙洲、沖積扇等海岸環境變遷歷程。歷史百年地圖所記載的資料為從清朝末期的戰略地圖、日治時期的三角測量繪圖等，因此在台南方面可以清楚地了解海埔新生地生成之年代，以及生成的範圍結果，透過套圖的變遷分析演算出海埔新生地成長的時間年代。

1. 不同尺度的歷史衛星影像與航照影像

雖然衛星影像解析度品質有大幅的提昇，但對於長期變遷的觀測研究而言但卻突顯出同一地區不同時期的衛星影像的解析度不同，而造成不同尺度的觀測數據。以SPOT系列衛星影像解析度為例，從早期的SPOT-1~SPOT-4多光譜影像解析度為20m，全色態影像解析度為5m，融合後的影像解析度為5m；至2002年升空的SPOT-5多光譜影像解析度為10m，全色態影像解析度為2.5m，融合後的影像解析度2.5m；直至2012年的SPOT-6與2014年的SPOT-7陸續升空後，影像解析度提高為多光譜影像6m，全色態影像解析度為1.5m，融合後影像1.5m；直至2014年升空的Pleiades衛星，高解析度影像品質，多光譜影像解析度為2m，全色態影像解析度為0.5m，融合後影像為0.5m。

1. 應用不同尺度航遙測影像辨識海岸環境與海岸線

本團隊將透過影像內插方法提高多光譜的影像解析度與全色態影像進行融合以進行後續海岸環境變遷分析使用。多光譜影像資料處理時首先針對遙測資料波段作選擇，波段選擇愈多則分類所得的資訊愈多，但分類所耗用電腦時間也愈久，有時還會引入雜訊造成錯誤。多光譜衛星影像各波段與地物反射波譜間的關係，三種基本型式的地物：(1)健康的綠色植物、(2)乾燥的土壤、(3)透明的水體等典型的波譜反射率曲線如圖3- 15，這些曲線的內容可做為指標判斷地物的類別及狀態。

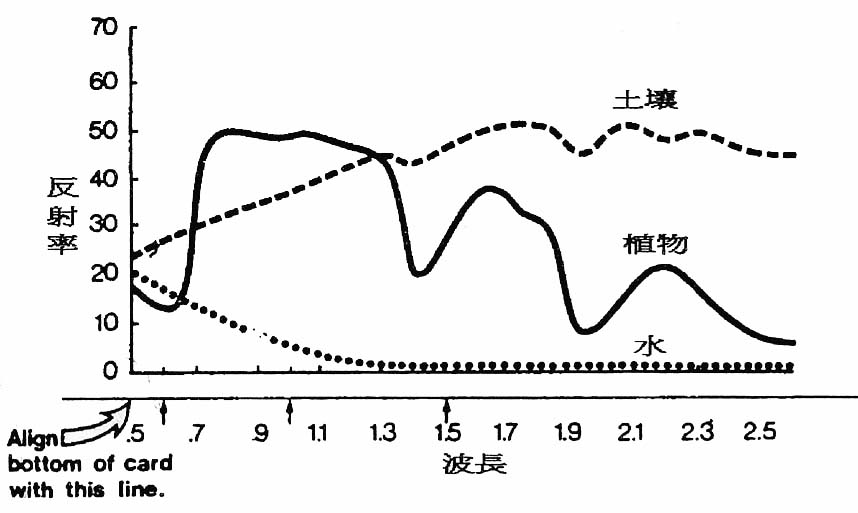


圖3- 15土壤、植物及水體之光譜反射特性曲線

本計畫將使用常用的正規化差異水指數（NDWI）計算多光譜影像資料，正規化差異水指數（Normalized Difference Water Index, NDWI）主要是利用近紅外波段和可見光的綠色波段來表達水的特徵範圍，是红光与近红外波段比值的指数，如公式(3)，NDWI可圈繪出水域邊緣的輪廓線，突顯出含水量高的地方，可提供本工作項目標記出海岸結構物與自然河道、沙洲、沖積扇等海岸環境特性，做為後續海岸環境變遷分析使用。

 (公式1)

NDWI值在-1〜1之間，其值在0附近的區域常為道路和建築物，小於0的區域屬於植物，大於0的區域為水體的訊號，值愈大代表水體訊號越強。本研究利用此特性，可得知河川行水區內含水量與水質混濁度變化情形，若有含水量異常增高與水質異常混濁的情形，須細部觀察其原因。

1. 海岸環境影像判識方法

本項工作將以低潮位線拍攝的歷史衛星影像多光譜與全色態影像、歷史航測影像資料，判釋安平地區（含海域區域）的灘地、河道寬度及彎曲度、沙洲、沖積扇、海岸線進行影像判釋與變遷分析，以瞭解安平地區（含海域區域）淺灘出現的比例、河道與海岸線的變遷過程。

本團隊將同時使用多光譜影像與正射影像，分別進行主動式輪廓擷取（Shape-based Level set Method）、擴張（dilation）、侵蝕（erosion）、骨架化（bone）、去支流（cutline）等影像強化處理，以得知地形或河道的骨幹輪廍及範圍，再以前後期影像進行套疊，根據影像前後期的變化，辨識灘地、河道面積、寬度與海岸線，以進行安平地區海岸環境空間長期的變遷調查，各影像強化處理的目的如下：

1. 色彩模型（Color model）

所謂色彩模型是一種抽象數學模型，通過一組數位資料來描述顏色（例如RGB使用三組8bit資料來表示顏色）。

1. RGB（Red、Green、Blue）

RGB是一種以加色概念來表示顏色的模型，將紅（Red）、綠（Green）、藍（Blue）三原色的色光以不同的比例相加（如圖3-50），以合成產生各種色彩光。RGB的紅、綠、藍都分別為三組8bit的資料，通常以（Red, Green, Blue）的方式表示顏色，由於每一組資料都是 8bit，因此範圍為（0~255, 0~255, 0~255），由於RGB是以三原色相加的概念，因此若要顯示紫色則是利用「紅加藍變紫」的概念，即可寫為（255, 0, 255）。

一張含有 磚塊, 黑暗, 靠近, 光 的圖片

自動產生的描述  
圖3-50三原色RGB示意圖  
（圖源自：https://en.wikipedia.org/wiki/File:RGB\_illumination.jpg）

RGB其特有的概念與結構較基本且易理解，三個分量（Red、Green、Blue）之間的關係極為緊密，因此RGB較適合用在系統中檢測、表示和顯示圖像，比如電視和電腦。

1. HSV（Hue、Saturation、Value）

如果說 RGB 分別代表紅（Red）、綠（Green）、藍（Blue）的話，那麼 HSV 則是分別代表色調（Hue）、飽和度（Saturation）、彩度（Value）。相較於 RGB 來說，HSV 教直觀的表示了顏色的色調、鮮豔程度等等，因此較常用在影像處理上，能較明確的追蹤某種顏色的物體，HSV 通常會用圓柱形的圖示（如圖3-51）來表示模型（敘述改編自：RGB轉換HSV及HSL型 <https://reurl.cc/bXr99o>）。

一張含有 文字, 光 的圖片

自動產生的描述  
圖3-54、HSV概念模型  
（圖源自：https://reurl.cc/bXr99o）

HSV的色調（Hue）、飽和度（Saturation）與彩度（Value）分別有其特性：

1. 色調（Hue）：代表的是顏色，以 0度 ~ 360度 來表示。
2. 飽和度（Saturation）：表示顏色空間中的灰色範圍，範圍從 0% ~ 100% (0~1)。當值為 0 時顏色為灰色，當值為 1 時顏色為原色。一個比較淡的顏色是由於飽和度較低，也就是顏色包含更多的灰色，因此飽和度越高顏色就包含越少灰色。
3. 飽和度（Saturation）：表示顏色空間中的灰色範圍，範圍從 0% ~ 100% (0~1)。當值為 0 時顏色為灰色，當值為 1 時顏色為原色。一個比較淡的顏色是由於飽和度較低，也就是顏色包含更多的灰色，因此飽和度越高顏色就包含越少灰色。

由於HSV 三分量（Hue、Saturation、Value）之間的關聯性較低，因此能夠直觀的表達顏色。相較 RGB 色彩模型來說，HSV更不易受光感影響，也因此較常用於影像處理。RGB與HSV之間的差異與用途如下表一：

一張含有 桌 的圖片

自動產生的描述  
表一、RGB與HSV之間的特性與差異比較

1. 灰階（Gray Scale）

在影像處理中，複雜的圖像資料並不適合計算，也很難聚焦在所要的目標圖像上，因此時常會對圖像進行灰階處理，在灰階圖像中就不像RGB那樣有三組8bit的資料來表示顏色，直白一點來看可以說只有一個值在表示顏色，但也可以說RGB三分量的值都相同，轉換公式如公式一。

Gray = 0.299 \* Red + 0.587 \* Green + 0.114 \* Blue（公式一）

最一開始可能會很直覺地想，彩色轉灰接的公式應該只要把RGB三分量的值加總除以三，但其實人眼對於紅色、綠色及藍色的敏感度並不一樣，因此以公式一的方式轉換灰階才能得到比較適合的灰階影像。（敘述改編自： [OpenCV] 轉換影像為灰階 (Transform Image to Gray Level)　https://reurl.cc/mLj0E9）

1. 二值化（Binarization）與Otsu

對於已經轉換成灰階的影像，可視為每一個 Pixel 都只有用一個值來表示，在這樣的情況下可以透過二值化將所需的資料留下。二值化即是將圖像所有點的灰階資料只分為兩種，0 或 255（也有人稱 0 與 1）而不再是 0 ~ 255 了，透過一個常數閥值將小於閥值的資料都設為 0，大於閥值的則設為255。以這樣的原理將圖像過濾成只有黑、白的圖像。

將圖像進行二值化處理時，對於這張圖像的二值化閥值該如何調整總是我們常常屢試不爽的難題，閥值的範圍可以很大也可以很小，從5至250之間都有可能，但總不能耗費大量的時間一個一個嘗試，因此在完全沒有參考值的情況下，要找出所有圖像的最佳二值化閥值其實是要花非常多時間的。所以 Otsu 演算法便是來試著解決這個問題，它會試圖將圖像的資料分成兩半，等於是先幫訂好一個這張圖像的「可能」最佳二值化閥值，再交由你去調整罰值的大小，當然 Otsu 並不是只能使用在影像二值化上，只要是選擇門檻值都能試著使用 Otsu。

在本研究以K-means方法來實現Otsu，整理成具體步驟表達如下：

1. 設定兩個群心。
2. 將每一個點依據該點的資料離哪一個群心較近，便把資料歸類為該群心範圍。直到將所有點都歸類完。
3. 將兩個群心所被歸類到的資料值加總並取平均，得出的兩個平均值則用來當作新的群心。
4. 若新群心與舊群心不同（表示群心取錯了需位移），那就把新群心在代回第一步驟到第三步驟，直到群心不再需要做更改為止。
5. 直方圖均衡化（Histogram Equalization）

直方圖均衡化主要用於將一幅圖像像素的色彩強度平均分佈，令圖像提高對比度及擁有更豐富的色彩，能使圖像不會過暗或過亮，常用來調整或美化過暗或過亮的圖片。其原理是將原始圖像像素的色彩強度均勻地映射到整個色彩範圍內，得到一個色彩強度分佈均勻的圖像，如圖3-51為最經典的 lena 還未進行直方圖均衡化的RGB分布圖（紅色為R、綠色為G、藍色為B），而經過直方圖均衡化後的圖3-52很明顯地可以觀察到圖像的色彩分布以 x 軸方向均衡拉開。（敘述參考自：Histogram Equalization 超詳細解說　<https://reurl.cc/a9jW6Y>）：

|  |  |
| --- | --- |
|  |  |
| 圖3-51、還未進行直方圖均衡化之原圖 | 圖3-52、進行直方圖均衡化之結果圖 |
| （原圖來源：<https://zh.wikipedia.org/wiki/File:Lenna.jpg>） | |

直方圖均衡化的演算方式介紹如下（敘述參考自：Histogram Equalization 超詳細解說　<https://reurl.cc/a9jW6Y>）：

假設輸入圖像A為5 \* 5的RGB圖像，色彩範圍為0～9。

一張含有 文字, 光 的圖片

自動產生的描述

1. 將每種資料（0 ~ 9）的出現次數進行統計

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 像素資料 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 出現次數 | 2 | 1 | 4 | 1 | 2 | 4 | 4 | 2 | 3 | 2 |

1. 計算出現的機率（圖像的總像素為5 \* 5 = 25，故計算機率時需除以25）

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 像素資料 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 出現次數 | 2 | 1 | 4 | 1 | 2 | 4 | 4 | 2 | 3 | 2 |
| 出現機率 | 2/25 | 1/25 | 4/25 | 1/25 | 2/25 | 4/25 | 4/25 | 2/25 | 3/25 | 2/25 |

1. 從最小的資料開始計算機率累積相對次數

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 像素資料 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 出現次數 | 2 | 1 | 4 | 1 | 2 | 4 | 4 | 2 | 3 | 2 |
| 出現機率 | 2/25 | 1/25 | 4/25 | 1/25 | 2/25 | 4/25 | 4/25 | 2/25 | 3/25 | 2/25 |
| 累計機率 | 2/25 | 3/25 | 7/25 | 8/25 | 10/25 | 14/25 | 18/25 | 20/25 | 23/25 | 25/25 |

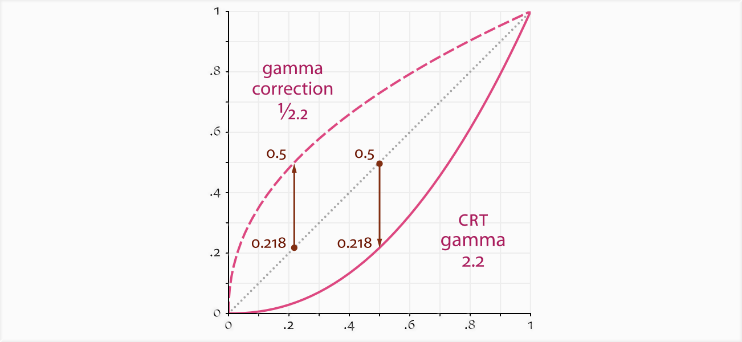
1. 將累積機率值乘上色彩範圍的最大值進行均衡化  
   假設像素資料為2的累積機率為7/25，那麼像素資料2的均衡化值為7/25 \* 9 = 63/25 = 2.52，四捨五入至整數為3

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 像素資料 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 出現次數 | 2 | 1 | 4 | 1 | 2 | 4 | 4 | 2 | 3 | 2 |
| 出現機率 | 2/25 | 1/25 | 4/25 | 1/25 | 2/25 | 4/25 | 4/25 | 2/25 | 3/25 | 2/25 |
| 累計機率 | 2/25 | 3/25 | 7/25 | 8/25 | 10/25 | 14/25 | 18/25 | 20/25 | 23/25 | 25/25 |
| 均衡化值 | 1 | 1 | 3 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

而這個均衡化值也就是指該像素點需調整的新值，也就是說原圖中像素點為 0 的都調整為 1、原圖中像素點為 1 的仍然為 1、而原本像素點為2的都調整為3，以此類推。最後所有像素點都調整過後即可得出直方圖均衡化的圖像。

1. Gamma校正

當在光線不足的幻境下所拍攝的圖像亮度過暗時，會導致目標物件不明顯，這時就需要將圖像的亮度調亮。最直覺的做法就是將每一個像素點的 RGB 三分量都往上加，但這樣會導致原本就落在 255 或接近 255 的值超過值域範圍，就算令超過 255 的停留在 255，也很有可能會使得圖像中原本該凸顯的特徵變得不明顯。這時就要利用 Gamma 校正，其特點就是利用絕對遞增函數來調整圖像（如圖3-53），避免上面提到的細節丟失的狀況。（敘述改編自：【影像處理】伽瑪校正 Gamma Correction　<https://reurl.cc/Q9A6kp>）

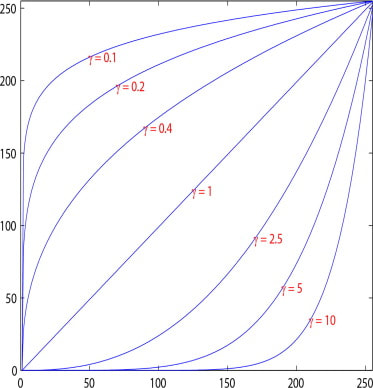
  
圖3-53、Gamma校正絕對遞增函數曲線圖  
（圖源自：What’s the best setting for Gamma with Rec.709 video?　<https://reurl.cc/MA5kgK>）

在 Gamma 校正的絕對遞增函數曲線中，資料越靠近極值（0與255）則增加或減少的幅度越小，然後在中間的區段調整的幅度則是較明顯。根據公式三，可歸納出Gamma校正的具體執行步驟為（敘述改編自：【影像處理】伽瑪校正 Gamma Correction　<https://reurl.cc/Q9A6kp>）：

（公式三）

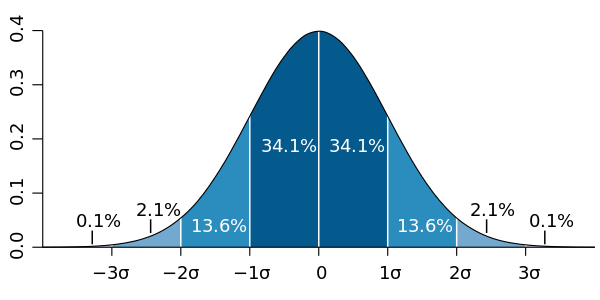
1. 先把每個pixel 點正規化到[0,1]
2. 將每個像素點資料以設定好的 Gamma 值（）做指數項的調整
3. 最後再乘上 Maximum 做還原

公式三裡的 A 為一個常數係數，通常為 1 可不理會，因此圖像的調整主要與 Gamma 值有關。經由不同的 Gamma 值做調整，就會有不同的輸出效果。由圖3-54的曲線可以發現 Gamma 值大於 1 時會使得資料變小圖像變暗，小於 1 則使資料變大圖像變得較亮。

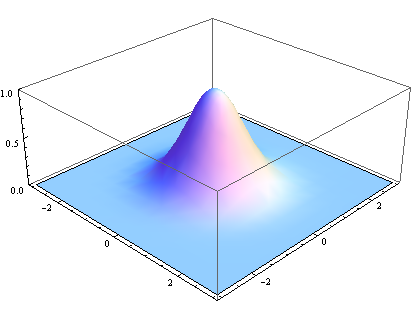
  
圖3-54、不同Gamma值在圖像上的變化曲線  
（圖源自：【影像處理】伽瑪校正 Gamma Correction　<https://reurl.cc/Q9A6kp>）

1. 高斯模糊（Gaussian）

說到圖像的模糊化，最簡單明瞭的方法即為均值濾波（Mean Filter），均值濾波是取像素範圍內的平均值，使得圖像有模糊化的效果。但如果使用只做簡單的取平均，顯然不是那麼公平，因為影像都是連續的，越靠近的點關係越密切，越遠離的點關係越疏遠，所以應該要對範圍內的每個像素分配合理的權重，距離越近的點權重越大，距離越遠的點權重則越小。而高斯模糊是利用常態分布曲線來做權重的分配，如圖3-55是一維的常態分佈曲線。

  
圖3-55、常態分布曲線圖（一維）  
（圖源自：File:Standard deviation diagram.svg　<https://reurl.cc/VEvvG5>）

但在圖像的二維空間中並不適用一維的常態分佈曲線，因此正確來說高斯模糊採用的是二維常態分佈曲線，如圖3-56。

  
圖3-56、常態分布曲線圖（二維）  
（圖源自：高斯模糊（高斯濾波）的原理與演算法　<https://reurl.cc/xGjMZV>）

有了常態分布曲線後，高斯模糊會利用常態分布的密度函式來計算二維空間內的權重，進而透過權重來對圖像中的每個像素資料進行調整，最後即可得出模糊化後的結果。高斯模糊具體的計算概紹如下（敘述改編自：高斯模糊（高斯濾波）的原理與演算法　<https://reurl.cc/xGjMZV>）：

1. 高斯函式

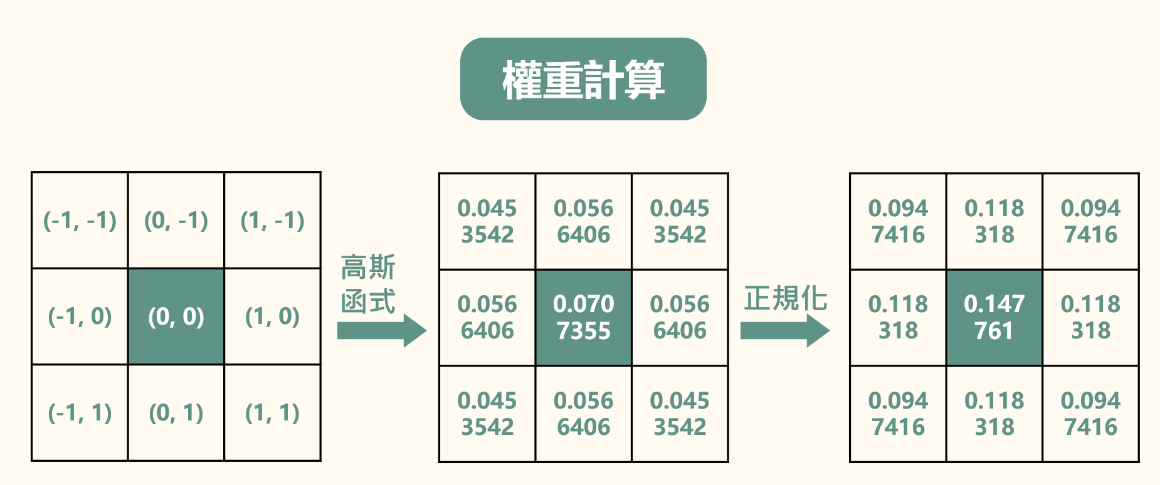
要取得各點的權重前，需要先了解用來計算權重的高斯函式（Gaussian function），常態分佈的密度函式又稱做高斯函式，它的一維公式如公式四。

其中，μ是x的均值，σ是x的方差。因為計算平均值的時候，中心點就是原點，所以μ等於0。將μ代換為0可得公式五。

但公式五仍然為一維的高斯函式，為了應用在二維的圖像資料上，將公式五開平方得二維的高斯函式，如公式六。有了二為的高斯函式後，即可對圖像中的像素計算權重。

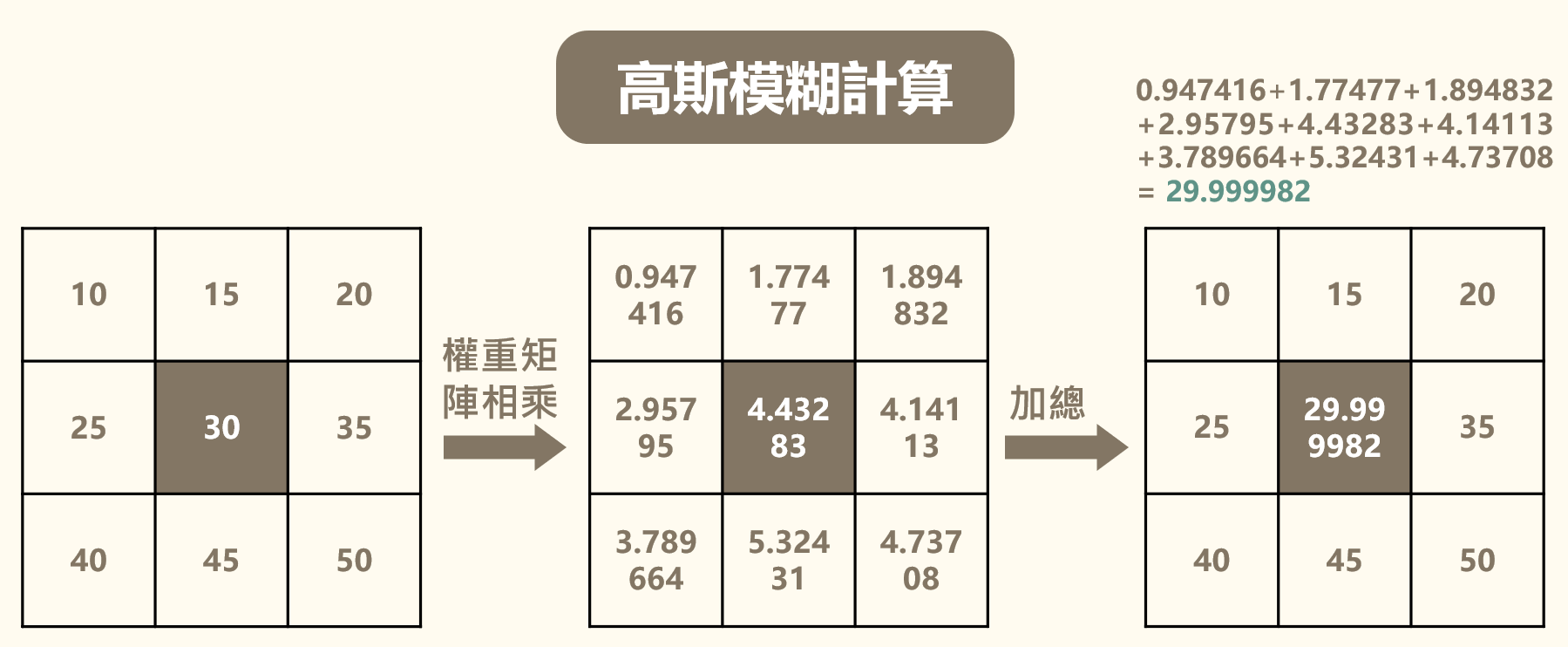
1. 權重矩陣計算

在本研究以3 \* 3的範圍為例計算個座標像素的權重。假設中心點為（0, 0）並將周圍的像素點依序填上x與y之座標關係（如圖3-57-左）。接著透過剛剛得出的二維高斯函式，將所有像素點的x與y帶入函式中得出周圍的各個座標像素之權重（如圖3-58-中），在這裡假定σ= 1.5代入計算。既然為權重，那麼相加的總和須為1，但目前所算出的權重總和0.4787147，因此圖中的9個值需要分別除以0.4787147得出最終的權重矩陣（如圖3-58-右）。

  
圖3-58、高斯模糊權重矩陣之計算步驟

1. 高斯模糊計算

由於已經得出了最終的權重矩陣，在這裡假設某像素點（x, y）其資料為30，以像素周圍的3 \* 3圖像為例。首先取得自身及周圍像素點資料（圖3-59-左），接著分別乘上剛剛算好的權重矩陣，像素中心乘以矩陣中心、像素左上乘以矩陣左上，以此類推，乘完後可得出新的像素資料（圖3-59-中），最後將這九個像素資料全部相加後得出的值即為像素點（x, y）的高斯模糊值（圖3-59-右）。

  
圖3-59、高斯模糊之像素點計算步驟

當圖像中所有像素點皆與算好的權重矩陣相乘過後，即可得出經由高斯模糊過後的圖像。

1. Canny邊緣偵測

Canny 是採用較複合型的方式尋找邊緣，用了多種方法將邊緣提取出來。可概分為四個主要步驟（參考資料：邊緣偵測懶人包-Canny演算法　<https://reurl.cc/W3Qvrx>）：

1. 濾除雜訊

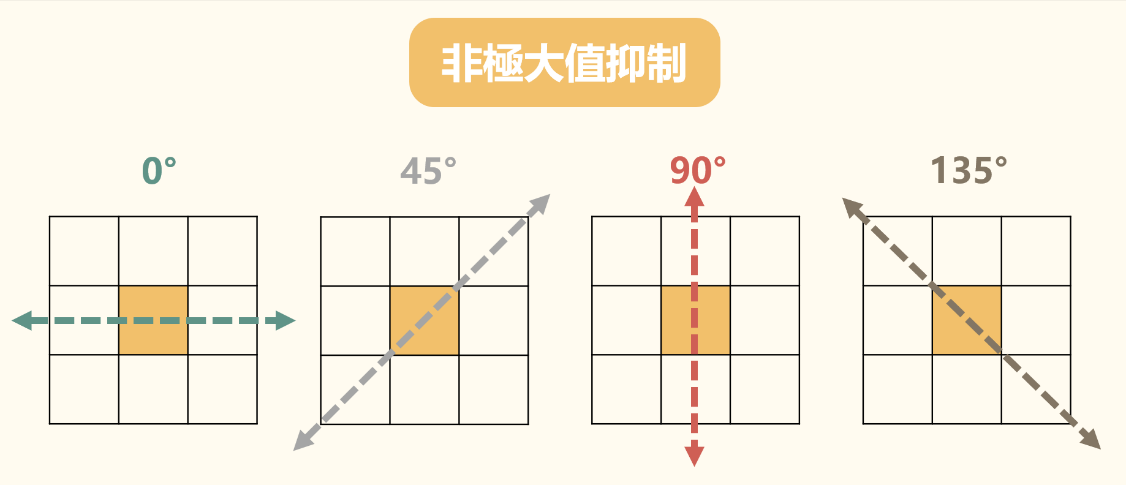
為了減少雜訊的產生而被誤判為邊緣資料，會先進行濾波使得圖像變得模糊並清除雜訊，常見的方法有均值濾波器或高斯模糊。

1. 找出圖像中每個邊緣的強度

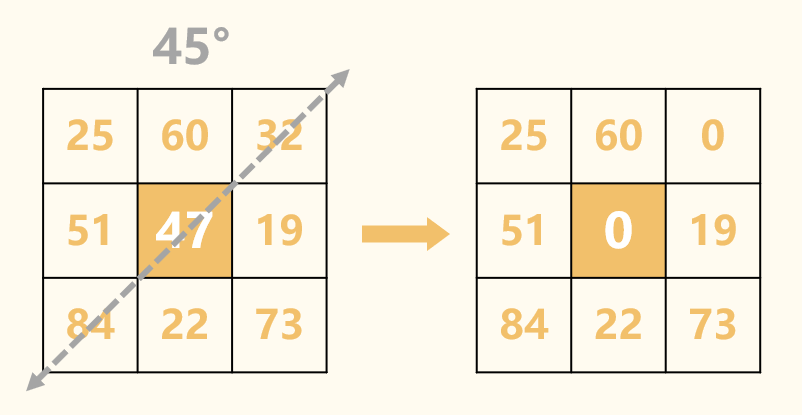
對圖像進行卷積，找出每一個像素點的邊緣強度梯度，梯度越高表示邊緣越明顯。Sobel 為目前較常用的方法。

1. 進行非極大值抑制（Non-Maximum Suppression，NMS）

在第二步驟找出圖像中各點的梯度值後，該點的像素資料越大代表梯度值越高，在Canny 演算法中，非極大值抑制是進行邊緣檢測的重要步驟。非極大值抑制可以理解為尋找像素點區域性最大值，將非極大值點所對應的像素資料設為 0，這樣可以剔除掉一大部分非明確邊緣的點。（敘述改編自：OpenCV: Canny邊緣檢測演算法原理及其VC實現詳解　<https://reurl.cc/lRrnkd>）

  
圖3-60、非極大值抑制四個梯度示意圖

非極大值抑制會利用圖3-60中的四個梯度方向將非極大值的像素點設為 0。以45°梯度方向為例，由中心點以45°角度檢查範圍內中每個像素點的梯度極大值，小於極大值者都會被判定為非邊緣而設為 0（如圖3-61）。以上圖四個角度的方向檢查圖像中的每一個點，完成非極大值抑制演算。

  
圖3-61、以45°梯度方向進行非極大值抑制

1. 設定雙閥值移除不必要的資料

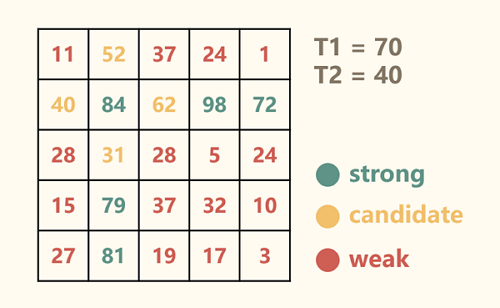
Canny 演算法中減少假邊緣數量的方法是採用雙閾值法。若只根據一個高閾值來過濾圖像，即使含有很少的假邊緣，但因閾值較高而產生的邊緣可能有缺漏或不閉合，為了解決這樣的問題則多採用了另外一個低閾值。（敘述改編自：OpenCV: Canny邊緣檢測演算法原理及其VC實現詳解　<https://reurl.cc/lRrnkd>）

Canny 演算法中，雙閥值的使用方法需先設定一個高閥值T1與低閥值T2，像素點資料以M(x,y)表示，並依照不同的情況將圖像中的各個像素點分為三類（如圖3-62），strong、week與candidate（敘述改編自：陳慶瀚 2004-11-03 - 單元六、邊緣偵測　<https://reurl.cc/Q9AmyM>）：

Case 1：假設M(x, y) >= T1，則該像素點（x, y）即為 strong

Case 2：假設M(x, y) <= T2，則該像素點（x, y）即為 week

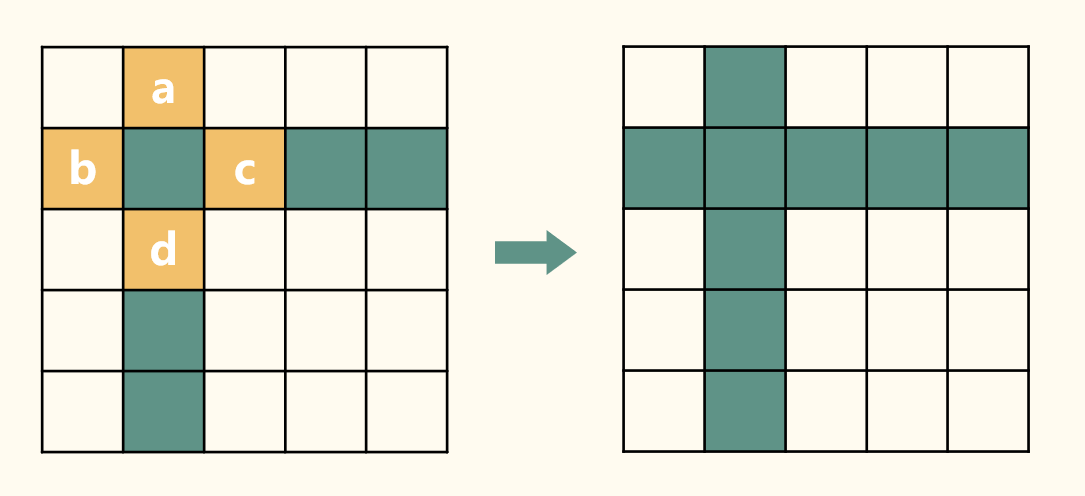
Case 3：其他介於 T1 與 T2之間，即為 candidate

  
圖3-62、Canny演算法以像素資料與雙閥值間之關係進行分類示意圖

設為 strong 的點視為必是邊緣的點，而設為 weak 的點視為一定不是邊緣的點。而 candidate 有另外的的處理方式。（敘述改編自：陳慶瀚 2004-11-03 - 單元六、邊緣偵測）

Case 1：假設存在一個candidate像素(x, y)，若該點附近有一個以上的 strong 像素相鄰，則像素(x, y)同樣被視為邊緣像素，如圖3-63-左的a、b、c與d點。

Case 2：如果存在一個 candidate 像素 (x, y)，判斷附近相連的邊緣像素方向是否有穿過(x, y)，若是則也視為邊緣像素，如圖3-63-左的 c 點與 d 點。

  
圖3-63、Canny演算法雙閥值對於Candidate的處理示意圖

將所有像素點都經由以上四步驟的處理過後，仍留存下來像素點即為Canny運算過後所找尋到的邊緣資料，由於Canny演算法較類似對各式邊緣進行層層過濾的概念，因此所留存下來的邊緣通常較簡單明確。

1. 主動式輪廓擷取（Shape-based Level set Method）

目的以自動化方式擷取出河道輪廓。以OpenCV的findContours實作，其主要參考 Satoshi Suzuk 在 1985 年發表的 Topological Structural Analysis of Digitized Images by Border Following，論文中提供找尋 contour的演算法以及證明。（敘述源自：findContours 實作OpenCV <https://reurl.cc/xG2b1E>）

在介紹演算法前，先概述OpenCV中對於尋找輪廓的考慮因素（源自：findContours 實作OpenCV <https://reurl.cc/xG2b1E），有以下3>點：

1. 影像為黑底白色物體，也就是0代表背景，1代表前景物體。
2. 像素 1 之間連接，以4-connected為定義，如圖3-3，所以像素0之間的相連以8-connected為定義，如圖3-4。（圖片源自：Pixel Connectivity <https://reurl.cc/vq2b4k>）

|  |  |
| --- | --- |
|  |  |
| 圖3-3、4-connected關係示意圖 | 圖3-4、8-connected關係示意圖 |

1. 根據以上的定義，使用Canny產生的影像無法滿足條件，因此需要先經過 Dilation（擴張）處理。

OpenCV主要是以Tree的概念來建構輪廓之間的關係，由於並不是所有的影像都為100%實心填滿，倘若某一輪廓中還有含有輪廓資料，那麼其輪廓與輪廓之間即會構成父子關係，故以Tree的概念表示之。如圖3-5中，塗色的部分代表像素為1之圖像，而塗色部分之邊界稱為outer border，非塗色部分則代表像素0之背景圖像，其邊界稱為hole border。

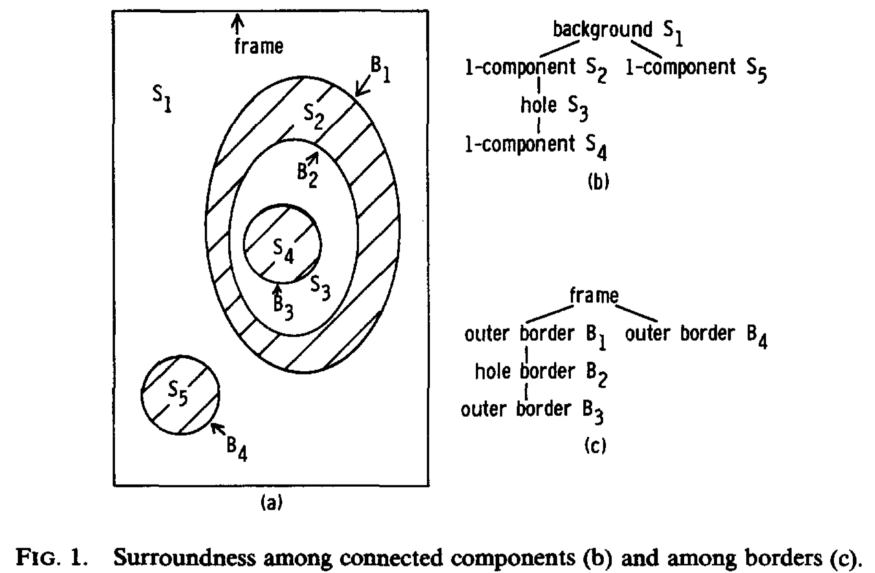
  
圖3-5 圖(a)中的連通分量(b)與輪廓性(c)之樹狀關係  
（源自：Topological Structural Analysis of Digitized Images by Border Following  
 [Satoshi Suzuk, 1985]）

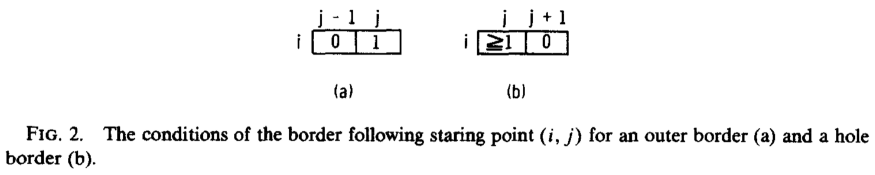
圖3-5中的參數說明如下（敘述源自：findContours 實作OpenCV <https://reurl.cc/xG2b1E>）：

1. outer border/contour（外部之邊界/輪廓）：被0-component（背景）包圍住的1-component（前景圖像）之邊界，定義在 1-component上。
2. hole border/contour（內部之邊界/輪廓）：被1-component（前景圖像）包圍住的0-component（背景）之邊界，也是定義在 1-component 上。
3. frame（圖框）：圖像的邊界。可以想像整張圖是被一個 1-component 包住，因此可以定義 frame 是一個 hole border，而圖像中 1-component 邊界的 parent 就是 frame。
4. NBD（輪廓之編號）：當前計算中的 contour 編號，其中 frame border 是 1。
5. LNBD（前輪廓之編號）：掃描尋找 starting point（Find starting point將在下段作介紹）時最後碰到的 border 編號，初始值是 1。用來判斷新的 contour 和其他 contour 的父子關係。

在了解圖像的建構關係後，進而了解OpenCV的findContours演算法主要分為三個步驟：Find contour starting point（找尋輪廓起點）、Contour following（輪廓追蹤）、Build hierarchy（建立階層關係）。在以下說明對這三個步驟進行說明：

1. Find contour starting point（找尋輪廓起點）

第一步的Find contour starting point表示由圖片左上方（0, 0）pixel開始，依照row-first的順序（先左至右，再由上至下）掃描。可以推斷出現 0/1 component 交界的條件必定是下列兩者之一，如圖3-6，其中 (a) 條件表示還沒被走過的 outer contour；由於在下一步的contour following會改變像素的值，因此 (b) 條件可以代表還未被走過的 hole contour，值 >1是考量到走過的contour會被標上NBD的標號。（敘述源自：findContours 實作OpenCV <https://reurl.cc/xG2b1E>）

  
圖3-6 外部邊界(a)與內部邊界(b)之起點(i, j)的判定條件

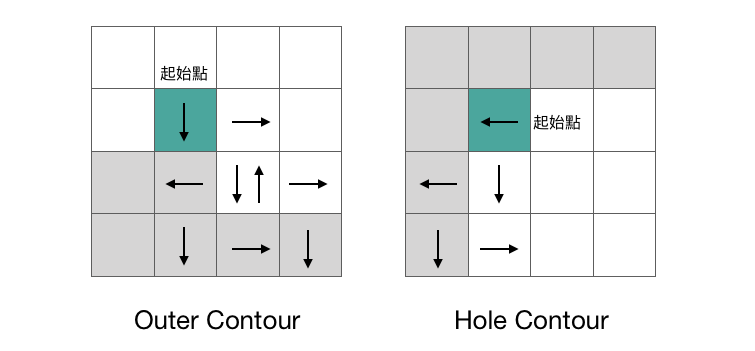
1. Contour following（輪廓追蹤）

在第一步驟的Find contour starting point找到 contour 的起始點後，下一步就是從起始點開始，依次走過並紀錄 1-component 的邊緣，這裡採用Square tracing algorithm（詳細可參閱：Square Tracing Algorithm <https://reurl.cc/5rQxNy> ），此方法只適合 4-connected case。它的步驟如下，如圖3-7（淺色代表0、深色代表1）（敘述源自：findContours 實作OpenCV <https://reurl.cc/xG2b1E>）：

步驟一：想像站在一個 contour 的邊界點（起始點）

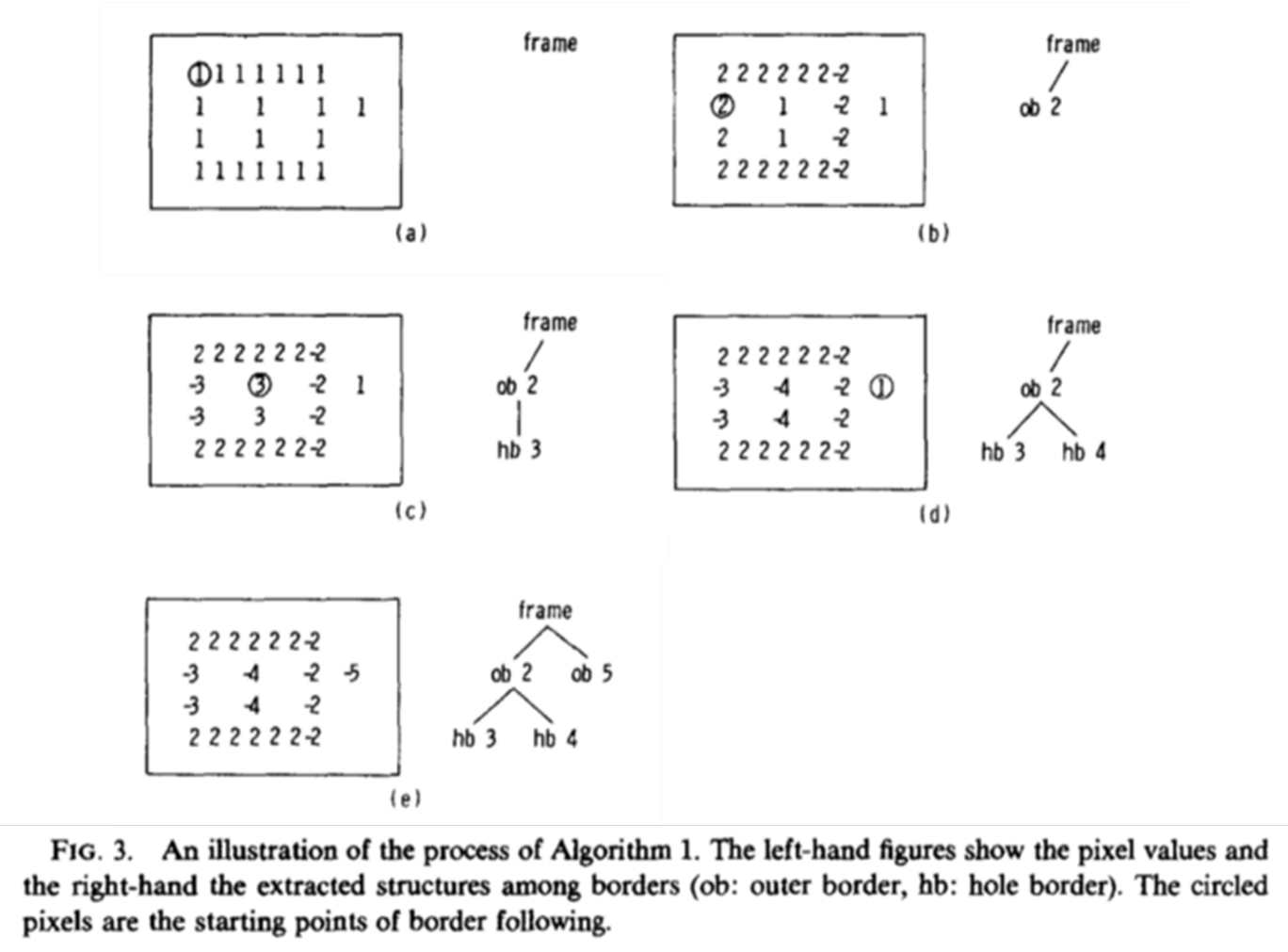
步驟二：如果當前的像素值是 1 時，轉向左邊，並踏出一步。反之，像素值是 0 時，轉向右邊，並踏出一步。紀錄走過的 1-pixel 點。

步驟三：按照 2 的規則持續進行，直到回到起始點。

  
圖3-7 以Square tracing algorithm搜尋圖像示意圖  
（圖源自：findContours 實作OpenCV <https://reurl.cc/xG2b1E>）

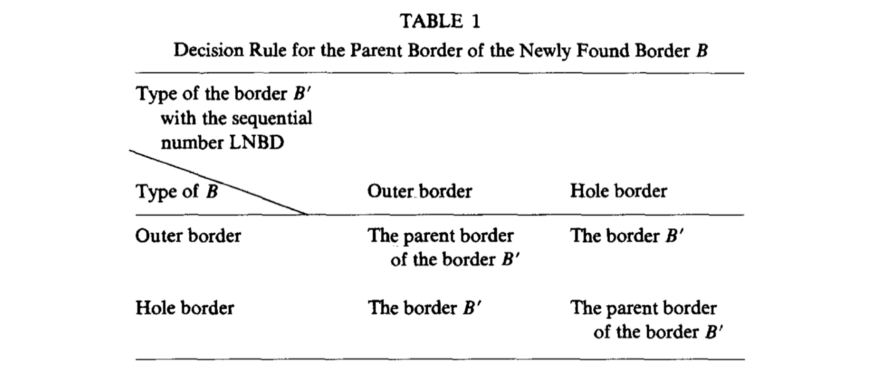
在以Square tracing algorithm進行 contour following 過程中，將會由以下的規則改變像素值：如果當前的 contour 點（p, q）的鄰近點（p+1, q）像素值為 0，則將 (p, q) 的值設成– NBD。除此之外，如果當前點的數值已經是某個 > 1的NBD則保持不變，否則設成 NBD。這樣的規則除了讓該點記錄當前的contour編號之外，也是為了記錄其邊界點。

此Contour following步驟之建構流程如圖3-8，其中共分為(a)至(e)個示意圖，每個示意圖的左側代表的是圖像中的像素資料；右側代表的是輪廓與輪廓間的父子結構（ob代表外部邊界、hb代表內部邊界），注意圖中之 (b) 圖左側的邊界點並不會被設為 –2，只會針對contour的右邊界設為 – NBD。（敘述源自：findContours 實作OpenCV <https://reurl.cc/xG2b1E>）

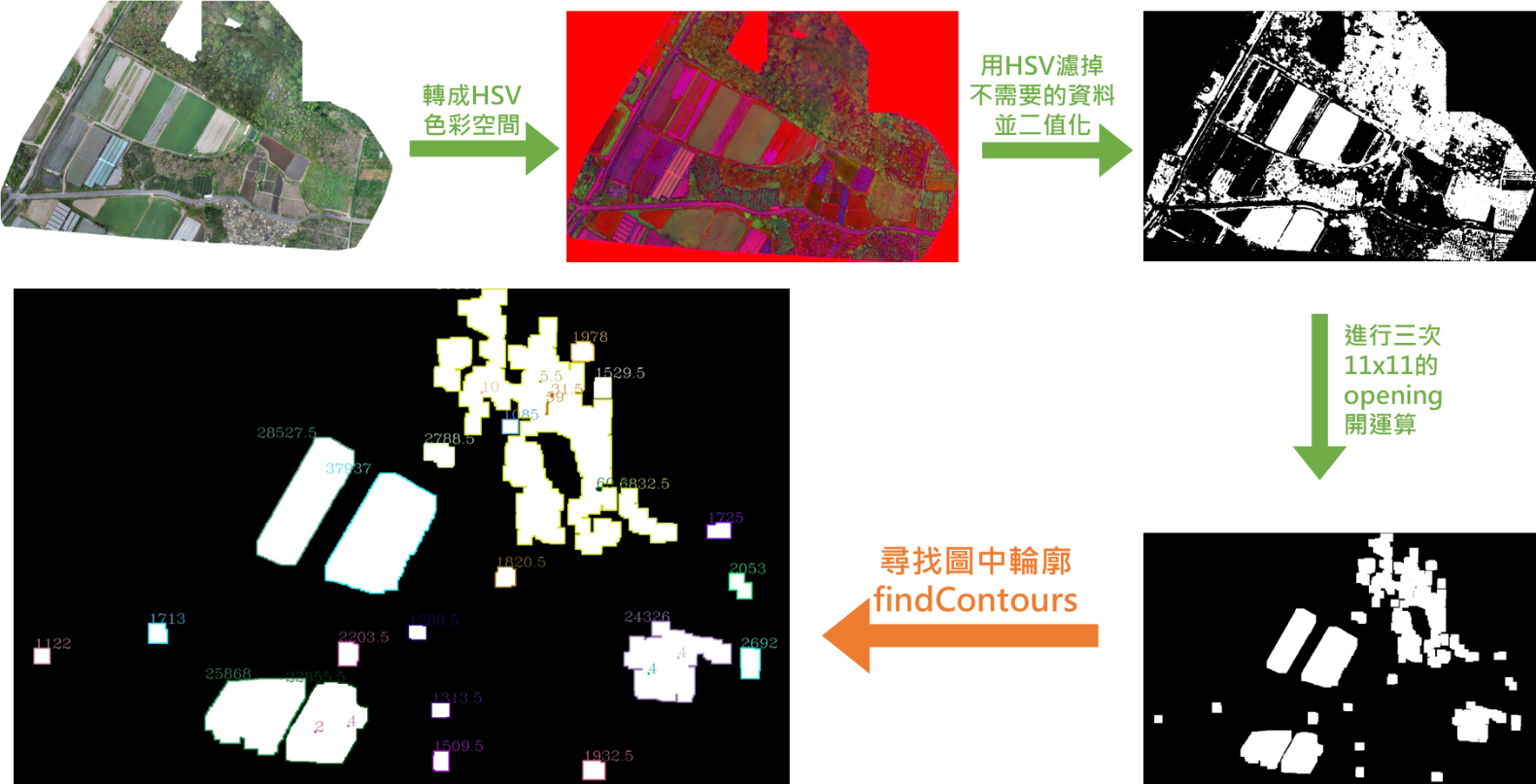
  
圖3-8 進行contour following之標號步驟流程  
（圖源自：Topological Structural Analysis of Digitized Images by Border Following   
[Satoshi Suzuk, 1985]）

1. Build hierarchy（建立階層關係）

最後的步驟為Build hierarchy，目的是找出 contour之間的父子關係。首先我們必須在前面介紹的Find contour starting point步驟掃描過程中紀錄最後碰到的 contour之NBD（無論正負），並紀錄在變數LNBD中。使得掃描輪廓的過程中，進而從新找到的contour以S代表和 LNBD 對應的S’ 得到關係：（如圖3-9）若S與S’ 擁有相同的型別（同為前景圖像outer或同為背景圖像hole），則代表他們擁有相同的父節點，反之則LNBD即為父節點。（敘述源自：findContours 實作OpenCV <https://reurl.cc/xG2b1E>）

  
圖3-9 對於新發現之邊界的父子關係決策規則  
（圖源自：Topological Structural Analysis of Digitized Images by Border Following   
[Satoshi Suzuk, 1985]）

實際運行結果如圖3-10，首先對原圖傳換為hsv色彩空間並過濾不要的資料，轉成二值化圖像後利用Opening運算去除雜訊，最後以OpenCV的findContours找尋圖中所有的contours。為了方便檢驗，通常會在調用OpenCV的drawContours來描繪出找尋到的所有contours，最後再利用contourArea以幾何學的方式計算所尋之輪廓面積，最後標示於contours上方以利後續之調整或研究。



3-10 主動式輪廓擷取findContours運行結果

1. Dilation（擴張）

目的將地形或河道內阻礙Skeletonization（骨架化）之空隙填滿。考慮在空間中的兩個集合 A 集合和 B 集合，當 A 集合被 B 集合膨脹時，可以用「A⊕B」表示，A 為輸入影像，B 為結構元素，當輸入像素及其周圍像素相對於結構元素為 1 的像素值有一個以上為 255 時，則將輸入像素的值設為255。運算結果會使影像看起來擴大，所以此運算稱為擴張運算，擴張的應用是用來做隙縫的填滿，利用適當的結構元素即可填補間隙。（敘述源自：侵蝕與膨脹<https://reurl.cc/838m9X>）

一張含有 文字, 填字遊戲 的圖片

自動產生的描述

圖3-16 Dilation（擴張）示意圖

在擴張運算中（如圖3-16），B 為像素值全為 1 的 3x3 的結構元素（-1 即像素點本身），且若「一個以上」的像素點符合結構元素 B 即設為 255。輸入圖像 A 中的 a 點其下方像素符合結構元素 B，故點 a 設為255，b 點四周皆不符合結構元素 B，因此點 b 會被設為 0。經過一整張的運算圖像會變得較大，且擴張也有填補缺洞的效用。

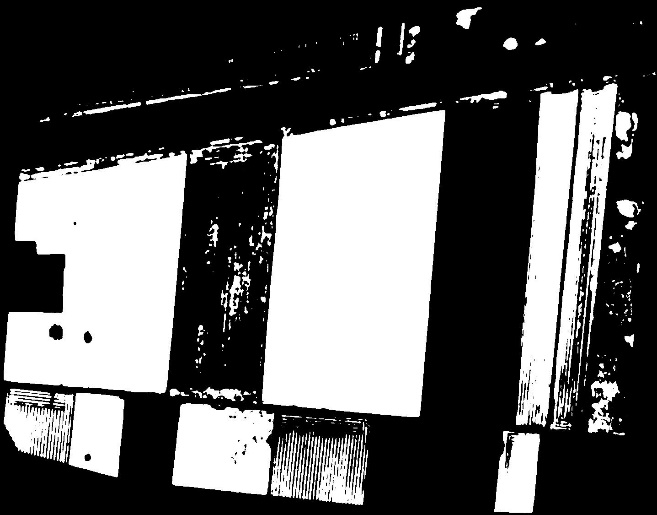
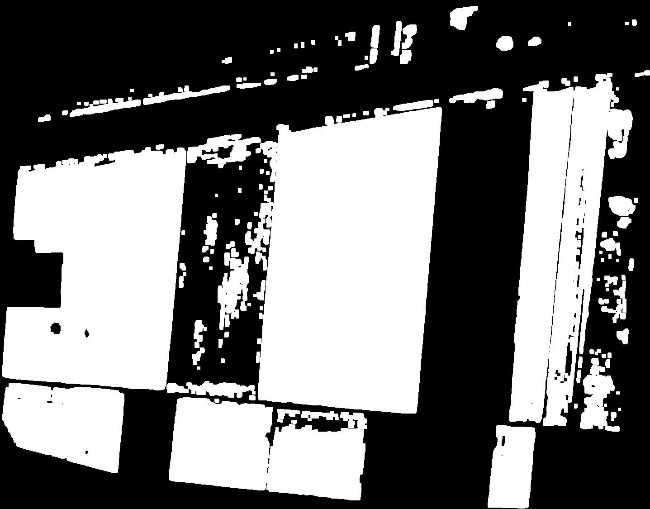
 

圖3-17未進行Dilation之二值化農地圖像 圖3-18進行Dilation之二值化農地圖像

由圖3-17與圖3-18可以明顯發現原先有細線條紋的農地，透過三次Dilation運算後變得較飽和，但有些雜訊同時也被跟著放大了且前景與前景之間的縫隙也跟著變得緊密一些。

1. Erosion（侵蝕）

目的為去除河道與地形周邊之雜訊。考慮在空間中的兩個集合 A 集合和 B 集合，當 A 集合被 B 集合侵蝕，可表示為「A⊝B」。A 為輸入影像， B 為結構元素，當輸入像素及其周圍像素相對於結構元素為 1 的像素值皆為 255，則將輸入像素的值設為 255。運算結果會使影像看起來收縮，所以此運算稱為侵蝕。侵蝕可以利用適當的結構元素將不必要的元素去除掉。（敘述源自：侵蝕與膨脹<https://reurl.cc/838m9X>）

一張含有 文字, 填字遊戲 的圖片

自動產生的描述

圖3-19 Erosion（侵蝕）示意圖

在侵蝕運算中（如圖3-19），B 為像素值全為 1 的 3x3 的結構元素（-1 即像素點本身），且「全部」的像素點符合結構元素 B 即設為255。輸入圖像 A 中的 a 點其上、左及左上方像素不符合結構元素 B，故點 a 設為0，b 點四周皆符合結構元素 B，因此點b會被設為255。經過一整張的運算圖像會變得較小，且侵蝕也有去除雜訊的效用。

一張含有 文字 的圖片

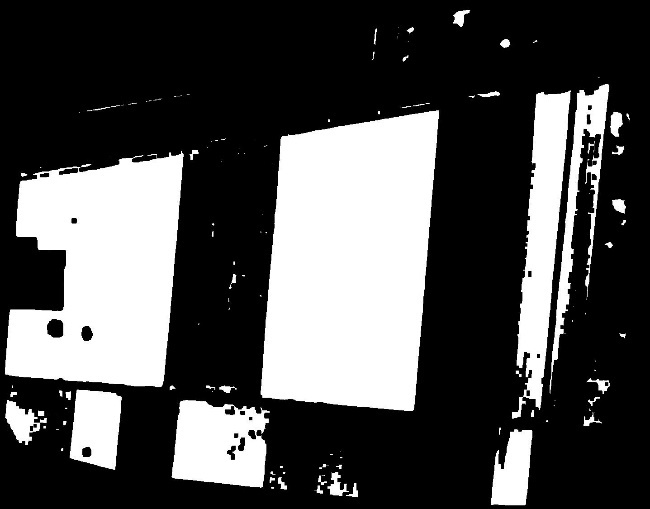
自動產生的描述 

圖3-20未進行Erosion之二值化農地圖像 圖3-21進行Erosion之二值化農地圖像

由圖3-20與圖3-21可以明顯發現有些細小雜訊經由三次Erosion後已經被消除，但有些農地的條紋線因為較細小因此隨著雜訊一起被侵蝕掉，且前景與前景之間的縫隙與坑洞也變大了一些。

1. Opening（開運算）

目的以Erosion（侵蝕）清除河道與地形周邊之雜訊後，再經過Dilation（擴張）之影像侵蝕回符合原圖地形或河道狀況。因此是以先Erosion再Dilation的概念來清除雜訊，先以Erosion清除雜點，但整張圖的前景目標會變得較小，因此再用Dilation來回歸原本像素值，如圖3-22。

一張含有 桌 的圖片

自動產生的描述

圖3-22 Opening（開運算）示意圖

如下圖3-23與圖3-24，由於先進行五次Erosion已經把許多雜訊白點給消除掉，最後又透過五次Dilation將前景像素調整為原大小，因此整體的開運算結果圖少了許多細小的雜訊點。

一張含有 文字 的圖片

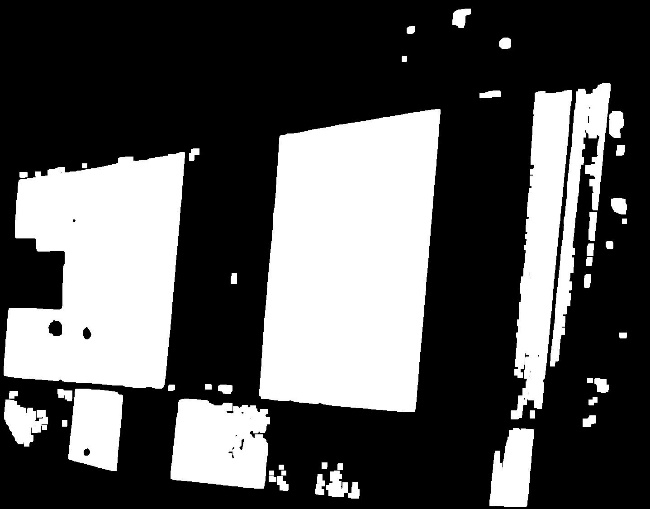
自動產生的描述 

圖3-23未進行Opening之二值化農地圖像 圖3-24進行Opening之二值化農地圖像

1. Closing（閉運算）：目的以Erosion（侵蝕）清除河道與地形周邊之雜訊後，再經過Dilation（擴張）之影像侵蝕回符合原圖地形或河道狀況。因此主要是以先Dilation再Erosion的概念來填補圖像缺洞，先以Dilation將缺洞填滿，但整張圖的前景目標會變得較大，因此再用Erosion來回歸原本像素值。

一張含有 文字, 名片 的圖片

自動產生的描述

圖3-25 Closing（閉運算）示意圖

如下圖3-26與圖3-27，由於先進行三次Dilation已經將細小坑洞與某些農地上的條紋填滿，最後又透過三次Erosion將前景像素調整為原大小，因此整體的閉運算使圖中的前景資料少了一些坑洞。

一張含有 文字 的圖片

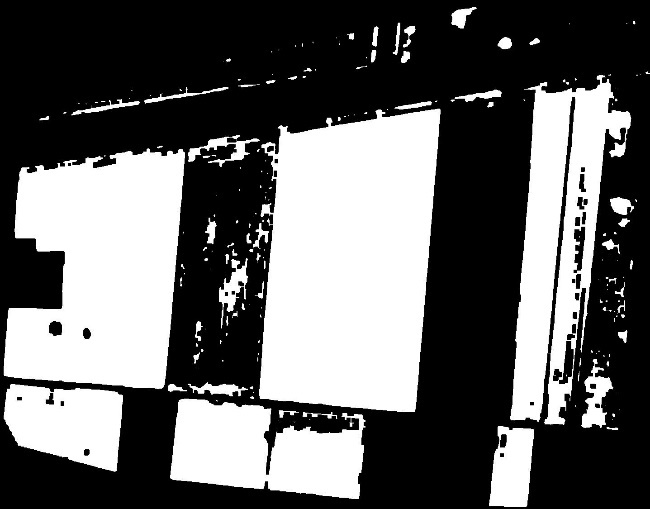
自動產生的描述 

圖3-26未進行Closing之二值化農地圖像 圖3-27進行Closing之二值化農地圖像

1. Skeletonization（骨架化）

目的將地形或河道影像以骨架方式呈現。Skeletonization的步驟理念是不停地對侵蝕過後的圖像做Opening（開運算），並將Opening所消除的像素點都記錄起來，再繼續對原圖做Erosion（侵蝕）與Opening直至圖像的像素點全為0。而在反覆做Opening的過程中，被消除並記錄起來的那些像素點就成為了Skeletonization的結果。

詳細演算步驟如下說明（譯自：Skeletonization in Python using OpenCV　<https://reurl.cc/7rmQNQ>）：

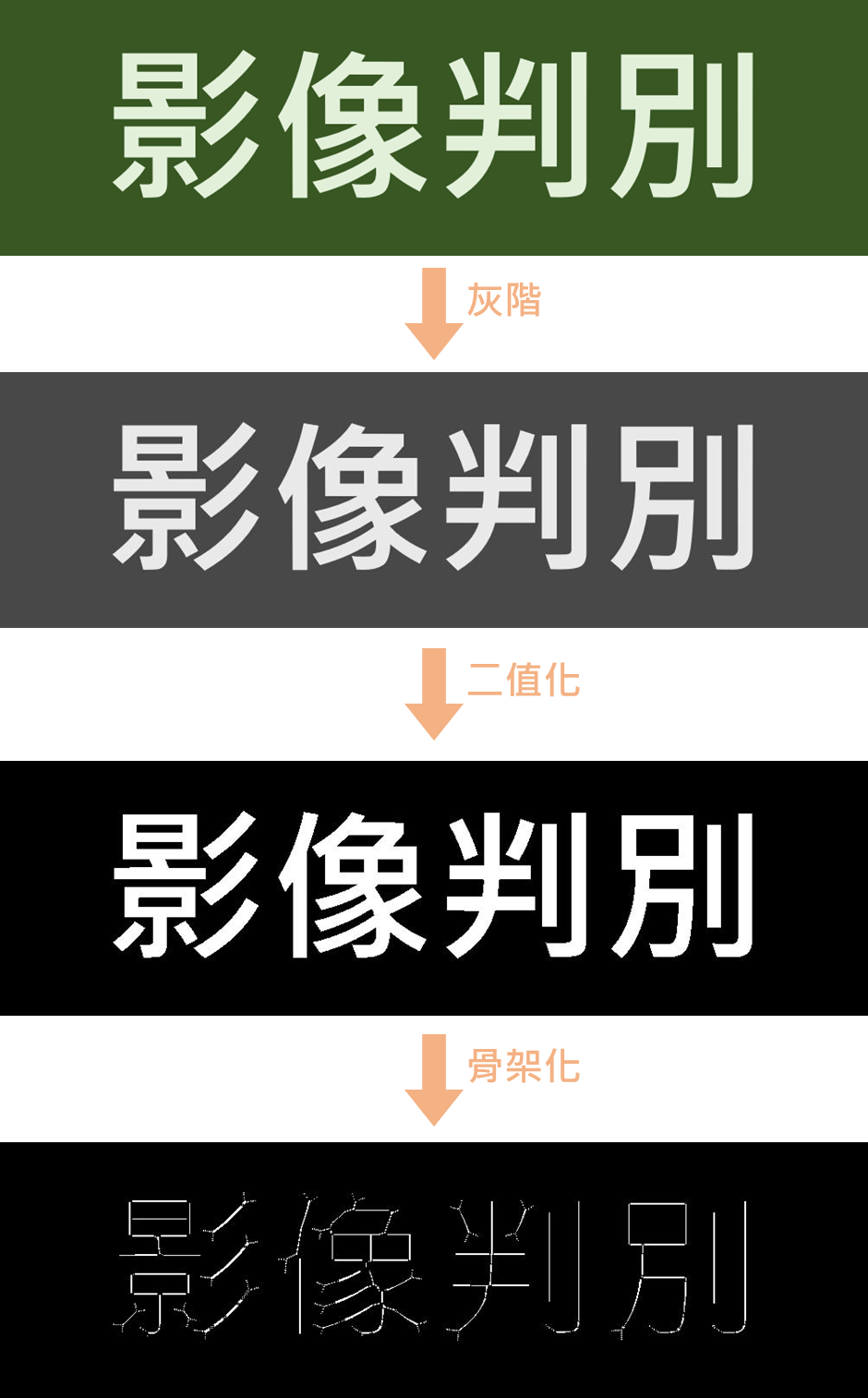
步驟一：先成立一個作為Skeletonization後的輸出圖像名為skel，其像素點需全為空值（0）。

步驟二：對原圖做Opening並存到一個名為Open的圖像中。

步驟三：將原圖減掉圖像Open可以得出原圖做開運算時所消除的點，並存到名為 tmp 的圖像中。

步驟四：對原圖做Erosion，並使圖像skel與圖像tmp做聯集運算。

步驟五：重複 2~4 步驟直到原圖全被侵蝕完畢，得 skel 為輸出結果（如圖3-28）。

  
圖3-28 經Skeletonization骨架化之結果圖像

1. Cutline（去支流）：去除非地形或河道主幹之支流（只留下地形或河道主骨幹）。

過去在擷取地形或海岸灘線的研究中，除了常用影像處理方法的邊緣偵測法之外，還需配合人工判釋將大部份的雜訊濾除，再透過細線化或骨架化將多點的資料細化，得到的灘線往往是片段式，需再使用內插方法將片段灘線連結成全域性的海岸線以進行後續分析，程序複雜且取得的地形或灘線輪廓仍不完全客觀。

本團隊將提出主動式輪廓擷取方法-形狀基礎等位函數法（Shape-based Level set Method），來改善過去常用的輪廓擷取方法，透過使用主動式輪廓（Active Contour）方法來進行影像分割，此方法是以等位函數法為基礎並套上形狀的特徵來進行輪廓擷取。。等位函數法主要是依影像各位置的特徵建立一個高度分佈圖，籍由影像特徵不斷改變高度分佈直至平衡收斂，並由收斂後的高度分佈圖，定義高度大於0者為物件內部；高度小於0者為物件內部，因此高度為0或正負臨界即為物作的輪廓。

本團隊以Chan-Vese1提出的等位函數法為基礎，會以一圓形輪廓為初始輪廓φ值再依公式4的更新程式進行更新演化φ值，直至收斂以擷取影像的輪廓。透過公式(4)中、兩項次的演化過程，可得到圖3- 16的高度等位線演化過程：圖3- 16(a)之紅色高度等位線為初始預設之高度等位線，因此圖中a、b區域的φ值為正，而c、d區域的φ值為負。我們同時可求得預設高度等位線內部(a∪b)影像灰階平均值及外部(c∪d) 影像灰階平均值。由圖可知 [b區域與c區域的影像灰階值] > [] > [] > [a區域與d區域的影像灰階值] （灰階愈暗其值愈小），因此之值在b區域與c區域為正而使得φ值增加（趨向內部）；同理，之值在ａ區域與ｄ區域為負而使得φ值減少（趨向外部），如圖3- 16(b)的標示。最後輪廓將收斂成如圖3- 16(c)的物件輪廓上。

 (公式2)

|  |
| --- |
| home1a.bmp home1b.bmp home3.bmp  (a) (b) (c) |

圖3- 16高度等位線演化之範例

同時加入形狀特徵值對原始等位函數法的影響，圖3- 17中的紅色實線為原始等位函數法的*φ*值分佈，黑色方框表示我們加入的形狀模型內部；而藍色虛線表示加入形狀特徵更新後的*φ*值，由於在形狀愈內部的值愈大（正值），反之在形狀愈外部的值愈小（負值），因此在形狀愈內部，提昇等高線*φ*值的力道愈強，在形狀愈外部，下拉等高線值的力道愈強。加入形狀特徵更新後的*φ*值（藍色虛線）都正確分佈（的區域為輪廓的內部，的區域即為輪廓的外部）。

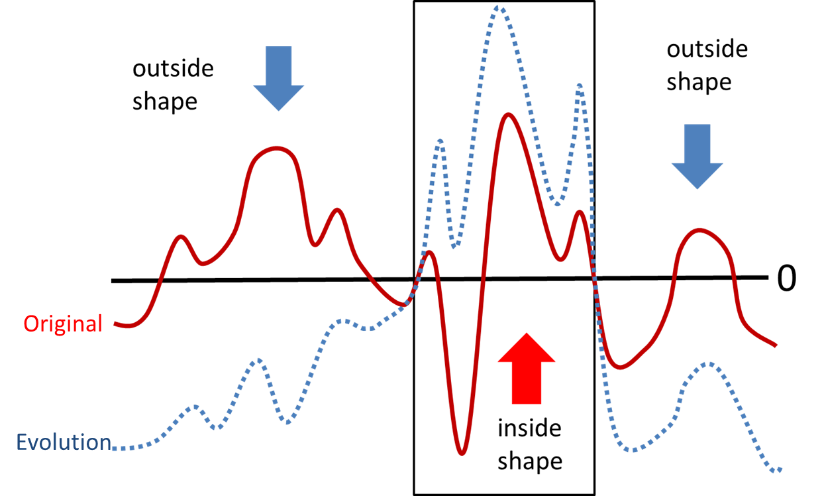
****

圖3- 17加入形狀資訊來改變等高線*φ*值演化區域

1. **影像辨識演算分析海岸線變遷歷程**

本項工作將以影像辨識演算法自動標記出歷年衛星與航照影像海岸線，提供影像辨識演算法以佐證分析海岸線歷年之變遷歷程。

本團隊於海岸環境影像變遷分析方法將使用(1)多時間影像顏色合成之視覺說明法，能快速得到定性變異區域的分析方法；(2)單波段的影像相減法（single-band image differencing）和主成分分析法（PCA），以得到定量變異區域的分析方法；由比較分析前期與後期影像像素點的顏色變化，進行變異監測之分析。

一般變異監測的方法分為二種處理，即前、後期影像變異監測及多時期影像變異監測。前、後期影像變異監測將二張影像視為獨立部份，利用二張影像光譜上的差異進行變異監測，如：影像差異法、影像比例法、分類後比較法，進行變遷的偵測。多時期影像變異監測將不同時期的影像，經影像前處理後，合併數張影像為一張影像，對合併後的影像等進行變異監測。常見的變異監測方法共分為三大類：

1. 數值表示：正規化植被指數（Normalized Difference Vegetation Index,NDVI）、正規化水分指數（Normalized Difference Water Index,NDWI）、正規化砂石指數（Normalized Difference Gravel Index ,NDGI）、近紅外線（Near Infra-red, NIR）及影像相減法。
2. 以數值分析影像如大致上植生NDGI值小於0，房屋建築之NDGI值約為0，開採成為水坑之區域NDGI值大於0.3。
3. 邊線特徵萃取型態以表達紋理：梯度影像法。
4. 梯度影像法需要門檻值界定邊界，能明確繪出開挖砂石坑及運輸便道之邊界。
5. 顏色型態表示：視覺說明法、標準假彩色影像、指數合成彩色影像。
6. 以顏色形態表示之變異監測法都需依照使用者之經驗知識作合適之判釋。

對於不同研究目的雖使用不同的影像分析方法，但為了能同時比對前後期影像圖以進行變異監測，無論是彩色正射影像或是多光譜影像，均使用常用的影像相減法，做為變遷分析探討之用。影像相減法是將經過幾何及輻射較正之前後期影像，所相對應像元的灰度值相減，得到介於-255~255之數值，若此數值接近0代表灰度值接近可視為無變遷，反之接近255或是-255代表變遷極大，如圖3- 18為影像相減法的示意圖。

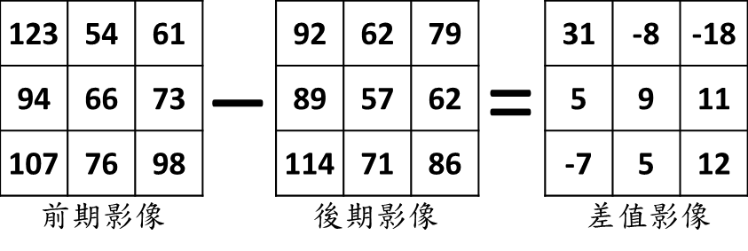


圖3- 18影像相減法示意圖

本團隊為了克服衛星影像長期觀測資料的尺度不一致性，將蒐集安平地區（含海域區域）歷年SPOT-2影像解析度20m、SPOT-5影像解析度10m、SPOT-6像解析度6m及Pleiades像解析度2m等歷史低潮位衛星影像與歷史航照影像，透過內插方法將不冋解析度的衛星影像提高到相同解析度或更高解析度的影像品質，以克服多時期衛星影像解析度不同之課題，提高不同時期衛星影像尺度不一致性對於長期環境變遷觀測之實用性。

常見的影像內插方法有最鄰近（Nearest）內插法、雙線性（Bilinear）內插、雙立方迴旋（Bicubic）內插法進行尺度一致性計算以利後續海岸線變遷分析使用；最近相鄰法不會影響原來的對比，內插後所產生的新灰度值及影像內的一些細微變化不致於損失過大，但原來線性構造如道路，會產生鋸齒狀的構造，優點是計算時間最快。雙線性內插法較最近相鄰法不易造成像元間的損失，其缺點在於處理時間為最近相鄰法的三倍。立方迴旋法可產生較圓滑的結果，雖視覺效果較佳，但影像上的特徵點或特徵線會被圓滑化，造成影像對比的喪失，且計算時間為最近相鄰法的十倍。