

# 國立交通大學

多媒體工程研究所

碩士論文

基於無母數直方圖之快速背景模型建構

Efficient Background Modeling Using Nonparametric Histogramming

研究生：施立宸

指導教授：莊仁輝 教授

林泓宏 教授

中華民國一〇三年七月

基於無母數值方圖之快速背景模型建構

Efficient Background Modeling Using Nonparametric Histogramming

研究生：施立宸

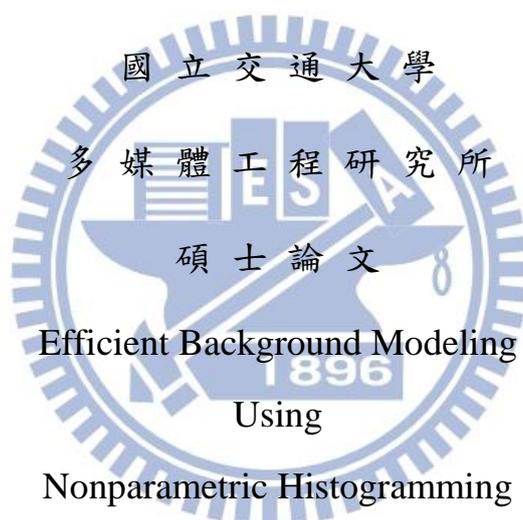
Student：Li-Chen Shih

指導教授：莊仁輝

Advisor：Jen-Hui Chuang

指導教授：林泓宏

Advisor：Horng-Horng Lin



Efficient Background Modeling  
Using

Nonparametric Histogramming

for the Degree of

Master

in

Computer Science

July 2014

Hsinchu, Taiwan, Republic of China

中華民國 一〇三年 七月

# 基於無母數值方圖之快速背景模型建構

研究生：施立宸

指導教授：莊仁輝  
林泓宏

國立交通大學多媒體工程研究所

## 摘要

監控設備不斷的發展進步，物件檢測效能上亦必須相對應的提升。而在物件檢測的技術上，背景相減法(Background Subtraction)在該技術上佔有舉足輕重的地位。目前以背景相減法技術來說，主流的背景模型為高斯混合模型(GMM)，該模型有許多相關的論文被提出。高斯混合模型可以對前景判斷保持一定的精確度，但其在監控設備不斷的進步下卻仍無法作出高解析度即時的前景偵測。本篇論文針對高斯模型不足之處發展出一套基於單像素上的背景模型，此模型解決高斯混合模型在模型更新上運算耗時的問題，並克服了高斯混合模型描述的限制，並可完整描述測量時間內每個顏色分布範圍的背景機率值。此外，由於此演算法製作簡易，因此可在定點運算(fixed-point)下進行，在不需其它加速模組的協助(CUDA、Intel SIMD 或是 Intel IPP library)，以及不採用多執行緒或取樣(Sub-sampling)的情況下，我們的演算法可以在 $1920 \times 1080$ 全彩影像下，硬體設備為 Intel Core i7 3.4GHz CPU 達到 23.25 fps 的速度。而我們的實驗結果也顯示，我們的演算法計算速度比 GMM 快上約 3.7 倍，並且能作出與 GMM 相近的結果。

# 目錄

第一章 簡介.....	1
第二章 相關研究.....	3
2.1 混合高斯模型(GMM).....	4
2.1.1 背景模型更新.....	6
2.1.2 前景像素判定.....	7
第三章 方法.....	9
3.1 直方圖背景模型.....	9
3.1.1 初始化.....	12
3.1.2 背景模型更新.....	13
3.1.3 前景像素判定.....	14
3.2 多重映射.....	15
3.2.1 雙重映射的背景模型更新.....	17
3.2.2 雙重映射的前景像素判定.....	18
3.3 系統加速.....	19
3.3.1 週期性背景機率調降.....	22
3.3.2 系統加速的機率調降與系統加速的機率提升.....	22
第四章 實驗結果.....	25
4.1 精確度與處理速度.....	25
4.2 前景切割應用.....	30
4.2.1 鐵道違規偵測系統：.....	30
4.2.2 紅外線打光器夜視系統.....	32
第五章 結論與未來展望.....	34
第六章 參考文獻.....	35

# 圖目錄

圖 1-1 在背景相減法的技術中，系統可從連續影像中的某張圖中(如左圖)得到該影像中屬於前景的部分(如右圖，黑色部分為背景，白色部分為前景).....	1
圖 1-2 圖(a)為輸入影像，而圖(b)為該影像的背景擷取圖片，圖(c)為圖(b)紅色點的像素之背景模型，該模型得到像素目前的顏色強度值資料並且整合過去幾張影像中該像素的強度值分佈。.....	2
圖 2-1 高斯模型為一機率統計模型，在標準差 $\sigma$ 平均為 $\mu$ 的情況下，我們假定該機率模型如上圖.....	3
圖 2-2 高斯混合 (GMM)，在機率分佈上使用了多個高斯模型來描述一個事件，在多個物件的識別上多了許多彈性.....	4
圖 2-3 高斯模型也需要因應新的資料點作更新的動作，如中圖的紅點，會影響高斯模型的分布情況。.....	5
圖 2-4 混合高斯模型(Mixture of Gaussian)運作流程圖.....	5
圖 2-5 給予前景適當的判斷式子以求出影像中前景的部分.....	7
圖 3-1 每個像素都有一個背景模型，該模型為一像素所對應到的直方圖，縱軸為背景機率值，而橫軸為 bin 的索引值。.....	9
圖 3-2 Bin 索引值化分範圍與起始點 $b_0$ ，而每個 Bin 具有寬度 $\Delta b$ 。.....	10
圖 3-3 以每個像素為單位的直方圖背景模型，綠色框線為目前該像素顏色強度值所對應到的 Bin 索引值，紅色虛線為閾值。.....	11
圖 3-4 直方圖背景模型的流程圖，系統會將第一張圖該像素的顏色強度值設定為背景作為初始化，接下來的每一張圖進行背景模型的更新與像素判定。.....	12
圖 3-5 第一張影像的初始化。圖(a)為原影像，而圖(b)為第一張影像的前景切割結果圖，圖(c)為圖(a)紅點處的直方圖背景模型，初始化令所有像素顏色強度值鎖定到的 bin，背景機率皆為 100%。.....	13
圖 3-6 背景模型更新有兩大步驟，圖(a)為在 $t$ 時間點時某像素的背景模型，在	

作對應 Bin 背景機率提升之前，我們需將所有的 Bin 進行機率調降，如圖(b)。

接下來再作對應 Bin 背景機率提升，如圖(c)。對應 Bin 以綠框表示。 .....13

圖 3-7：背景模型的更新後，在前景像素判定步驟每個像素皆會將目前像素對  
應到的 Bin (以綠色矩形框起) 之背景機率值，與閾值 $TF$ 相比較，高於此閾值  
則判定為背景 (圖中黑色部分)，反之則標上白色記號以示為前景。 .....15

圖 3-8 上圖為兩個時間點時，相似的颜色強度值可能會受到少量雜訊的干擾，  
使原先該被分配到相同 Bin 的颜色值，被判定為兩個不同的 Bin 的情況。 .....16

圖 3-9  $k_0$ 為原先的 Bin 畫分範圍，加入多重映射後，除 $k_0$ 外，我們令 $k_1$ 為額  
外的映射 Bin。 .....17

圖 3-10 加入雙重映射的方法，我們修改原圖 3-4 的流程圖，紅色部分為有修  
改處.....18

圖 3-11 系統以背景機率較大的 Bin 作為判定背景像素的依據 .....19

圖 3-12 代表目前時間點的 $gt$ 與最後更新時間點的 $lk, t$ 相減可得到該 Bin 需要調  
降的次數 $d$  .....20

圖 3-13 代表目前時間點的 $gt$ 與最後更新時間點的 $lk, t$ 相減可得到該 Bin 需要調  
降的次數 $d$  .....20

圖 3-14 經過系統加速後的流程圖，與雙重映射的流程圖相比，修改過的部分為  
背景模型更新處(紅色框).....21

圖 3-15 週期性調降兩步驟：(1) 根據 $gt$ 與 $lgt, t$ 來調降第 $gt$ 個 Bin 的背景機率值  
與(2)將 $lgt, t$ 設為目前系統計數器的值 $gt$  .....22

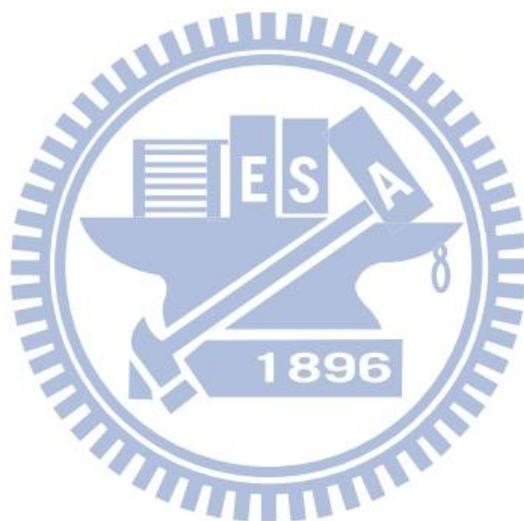
圖 3-16 系統加速後的背景模型更新與前景像素判定的例子 .....23

圖 4-1 前景切割結果比較。列 1：輸入影像；列 2：標準前景切割結果；列 3-  
6：GMM、ZGMM、我們的方法(Fixed-point 實作、 $N = 16$ )以及我們的方法  
(Fixed-point 實作、 $N = 32$ )之前景切割結果(a)&(b)：video 1.  
(c)&(d):video 2. (e)&(f): video 3. (g)&(h)：video 4. ....27

圖 4-2 前景切割結果精確度比較。列 1：輸入影像；列 2：標準前景切割結果；列 3-7：GMM、ZGMM、我們的方法(Fixed-point 實作、 $N = 32$ )，我們的方法(Fixed-point 實作、 $N = 16$ )及我們的方法(Fixed-point 實作、 $N = 8$ )之前景切割結果。.....28

圖 4-3 使用背景模型抓出某區段時間內車留路徑與截取違規路徑，圖(a)為某區段內所有被截取出來的車輛、行人移動路徑，而圖(b)則保留違規路徑部分。.....31

圖 4-4 紅外線打光器偵測較適當距離的前景，擷取前景 4 個頻道為紅外線打光器由暗至亮的四種亮度，橫排為輸入影像與兩模型的背景相減法之輸出影像..32



# 表目錄

表 4-1 測試影片特性 .....	26
表 4-2 背景模型計算速度比較，以 GMM 的速度作為基準速度(100%)進行比較。單位：FPS.....	26
表 4-3 測試影片特性 .....	27
表 4-4 本方法在精確度與速度上和 GMM, Z-GMM 作比較 .....	29
表 4-5 本方法在精確度與速度上和 GMM, Z-GMM 作比較 .....	30
表 4-6 鐵道違規偵測下使用原始 GMM 與本方法在精確度上的比較。 .....	31
表 4-7 鐵道違規偵測下使用原始 GMM 與本方法在速度上的比較。單位：FPS .....	31
表 4-8 紅外線打光器前景切割下使用 GMM 與本方法在速度上的比較。 .....	33



# 第一章 簡介

近年來隨著監控裝置技術的進步，影像格式已從 NTSC/PAL/VGA 提升至高解析度(High-Definition)以上，此兩種格式畫素差異至少約為 6.75 倍。在這樣的技術進步下，高效率、即時智慧型分析(Intelligent video analysis, IVA)的需求呼之欲出。而背景相減法(Background Subtraction)在智慧型分析的架構下是一個較為基礎的技術，該技術可從動態影像中分析出前景與背景，如圖 1-1。利用背景相減法得到影像中的前景與背景後，系統即可從獲得的前景或背景影像中繼續作分析或後處理的動作。因此在智慧型分析系統下，背景相減法占有舉足輕重的地位。而影響背景相減法最關鍵的因素便是，如何選用正確的模型來描述目前的前景。

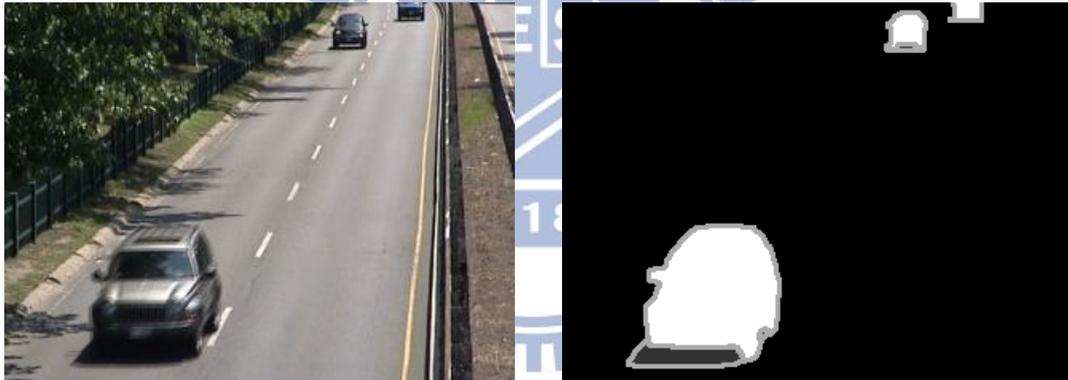


圖 1-1 在背景相減法的技術中，系統可從連續影像中的某張圖中(如左圖)得到該影像中屬於前景的部分(如右圖，黑色部分為背景，白色部分為前景)

但在傳統的背景模型上，存在著計算上的效率問題。GMM (Gaussian Mixture Model) [1]為較主流的背景模型，該背景模型雖然穩定，但由於須維護每個像素中高斯模型的平均值與變異量，並且判斷每個時間點每個像素的顏色強度值屬於哪個高斯模型，這樣的計算複雜度較高。因此該方法仍然無法在高解析度的畫素下做到即時分析的效能。另一方面，目前監控裝置的數目日增，在要求 IVA 系統的情境下需要更快速的系統出現。此外，近年來越來越多的 IVA 與硬體作結合，但傳統的背景模型多使用浮點數運算(Floating-point)，與定

點(Fixed-point)運算趨向的硬體結合不易。也因此，在此趨勢下，有利於定點運算的背景模型因應而生。

考量影像解析度的提昇、攝影監控頻道的增加與硬體 IVA 模組設計的需求，本研究提出一創新的背景模型估計演算法，此演算法以無母數直方圖為基礎，每個像素皆擁有一個背景模型的情況下，可以快速計算出一全像式的場景背景模型描述，如圖 1-2 所示。此演算法運算簡易，因此可以實作定點(Fixed-point)模式進行計算。實驗結果顯示，在不採用任何加速器(如 CUDA, Intel SIMD 或 Intel IPP library)，亦不作多執行序或取樣(Sub-sampling)運算的協助下，我們所實作的新式演算法可對 Full-HD 影片作背景模型建構與前景分析，在 Core i7 3.4GHz CPU 上達到約 23.25 fps 的運算效率。此外，實驗比較結果顯示我們的方法比傳統 GMM 至少快了 3.7 倍以上。

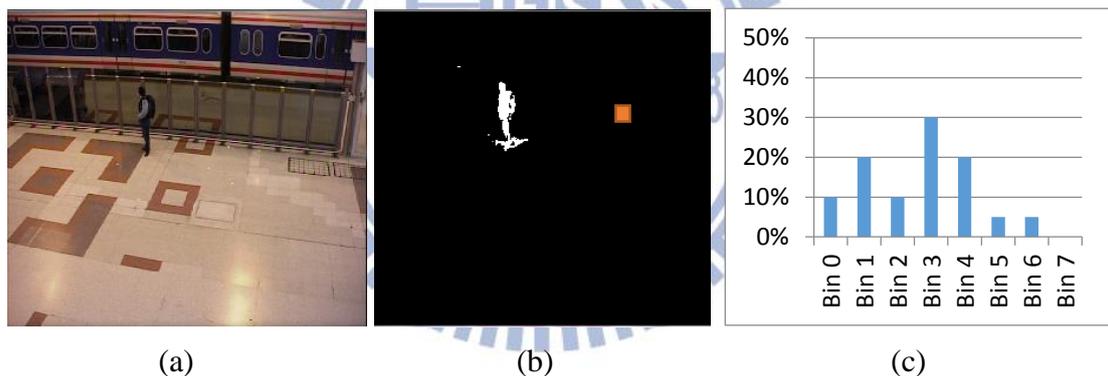


圖 1-2 圖(a)為輸入影像，而圖(b)為該影像的背景擷取圖片，圖(c)為圖(b)紅色點的像素之背景模型，該模型得到像素目前的顏色強度值資料並且整合過去幾張影像中該像素的強度值分佈。

## 第二章 相關研究

由於背景模型的研究上我們使用機率來描述影像中一個像素顏色強度值在一段時間內的變化，因此我們可以將其看作是一種靜態密度估測的研究。在該研究領域上，較為主流的技術為 GMM [1]，在介紹 GMM 前，我們先對高斯 (Gaussian) 模型進行介紹。

高斯分佈，又稱常態分佈 (Normal distribution)，在統計上與許多方面有著重

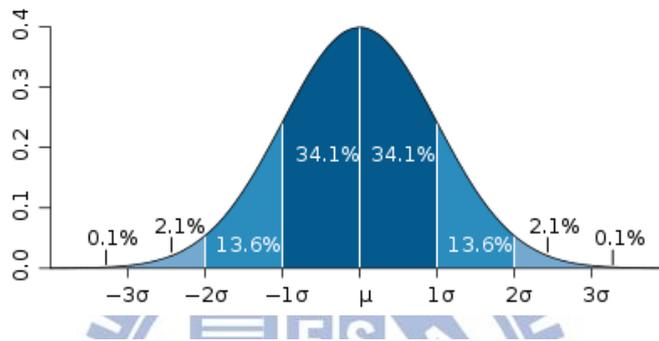


圖 2-1 高斯模型為一機率統計模型，在標準差  $\sigma$  平均為  $\mu$  的情況下，我們假定該機率模型如上圖

大的影響力。該模型主張在標準差  $\sigma$ ，平均值為  $\mu$  的情況下，我們可以得出一個機率模型來描述該數值分佈，如圖 2-1。若有一個隨機變量  $x$ ，遵守高斯模型的分佈的話，其機率密度函數為  $f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$ ，在高斯分佈下，我們就可以知道該隨機變量  $x$ ，在該高斯分布下的機率。而簡單的介紹完高斯模型後，接著將介紹較為複雜的情形—高斯混合模型 (GMM)。

## 2.1 混合高斯模型(GMM)

高斯混合模型(GMM)使用了多個高斯模型來描述整個機率分佈，如圖 2-

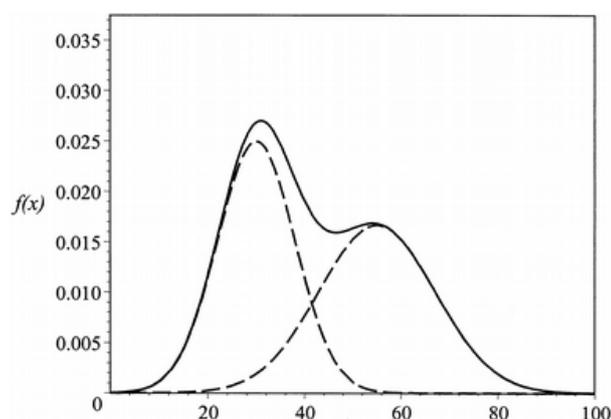


圖 2-2 高斯混合 (GMM)，在機率分佈上使用了多個高斯模型來描述一個事件，在多個物件的識別上多了許多彈性

2。以我們要處理的情境來說，在一個動態影像中，某個像素可能在某個時間點為白雲，下個時間點為太陽的不同情況。由於不同物件的顏色強度值會不一樣，因此我們需要為不同物件建立不同的高斯模型。這樣的特點使得高斯混合(GMM)模型成為背景模型中較為主流的方式。而在現實條件下，一個像素會出現的物件會隨著蒐集的動態影像長度與複雜度而增加，舉例來說：某像素在一個動態影像中，蒐集了一整天的影像，有可能會出現綠地、人物、小狗、小貓等不同物件，但實作上該模型不會將這些物件所屬的高斯模型作無限期的保留。在系統上可同時記錄的高斯數目是有限的，當有新的顏色強度值進入，系

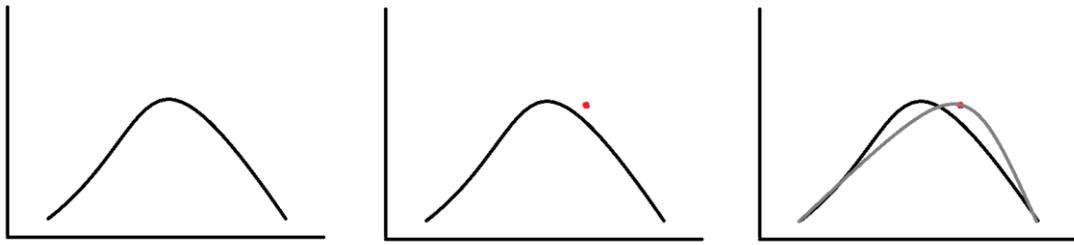


圖 2-3 高斯模型也需要因應新的資料點作更新的動作，如中圖的紅點，會影響高斯模型的分布情況。

系統發現並沒有任何的高斯模型符合該顏色強度值，且高斯模型的記錄已達到上限，此時就會剔除掉最久沒更新的高斯模型。反之，如果有發現符合的高斯模

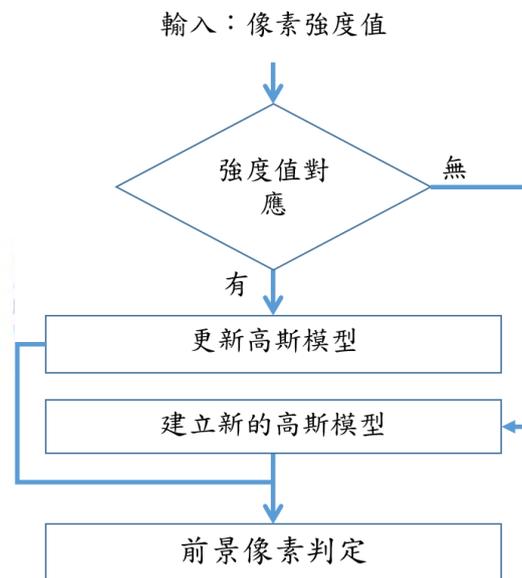


圖 2-4 混合高斯模型(Mixture of Gaussian)運作圖

型，系統也需要因為新進入的資料值而更新高斯模型，如圖 2-3。此外我們設立一個權重值來表示該高斯模型更新頻繁程度，常出現的顏色強度值會使得該強度值所對應到的高斯模型有較高的權重。整理上述內容，GMM 的處理流程應如圖 2-4，我們更具體的描述整個高斯混合模型(GMM)的運作方式。

### 2.1.1 背景模型更新

背景模型更新包含了更新高斯模型與建立新的高斯模型。我們使用  $K$  個高斯分布來描述像素隨著時間  $t$  的強度值  $I_t, I_{t-1}, \dots, I_0$ ，因此有公式如式(1)：

$$P(I_t) = \sum_{k=1}^K \omega_{t-1,k} N(I_t; \mu_{t-1,k}; \sigma_{t-1,k}^2) \quad (1)$$

$N(I; \mu, \sigma^2)$  代表著高斯密度函數， $\mu$  為平均值， $\sigma^2$  為變異量， $\omega_k$  為第  $k$  個高斯的加權值。假定目前像素的強度值  $I_t$  符合第  $m$  個高斯分布，代表符合條件

$|I_t - \mu_{t-1,m}| < T_\sigma \sigma_{t-1,m}$ ，在此處  $T_\sigma$  為預先定義的閾值。若找到第  $m$  個高斯模型符合該條件式。則該高斯模型可以下兩式子更新平均值與變異量，如式(2), (3)

$$\mu_{t,m} = (1 - \rho)\mu_{t-1,m} + \rho I_t. \quad (2)$$

$$\sigma_{t,m}^2 = (1 - \rho)\sigma_{t-1,m}^2 + \rho(I_t - \mu_{t,m})^2. \quad (3)$$

在此處參數  $\rho$  為模型的學習率。在[2]中有記載，加入第二的學習率  $\eta$  給高斯權重的更新，這樣的背景模型在景物變化的情況下比較具備強健性，權重的更新公式如式(4)

$$\omega_{t,k} = (1 - \eta)\omega_{t-1,k} + \eta I[k = m]. \quad (4)$$

而在上式中的  $I[.]$  為指示函數 (indicator function)，意即當括號中式子成立時為 1。

接下來我們再來探討，當在像素強度值  $I_t$  找不到合適的高斯模型時，我們需要建立新的高斯模型給這個強度值，創立新的高斯但要維持高斯模型總數的一致，因此我們可能需要剔除舊的高斯模型，因此在剔除方面我們選擇剔除最低加權值的高斯模型。

## 2.1.2 前景像素判定

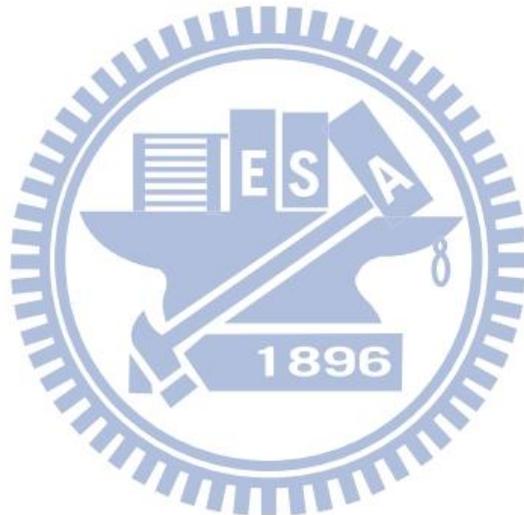
最後，我們要分辨目前的像素為前景或背景，因此我們給予此像素一個標籤  $L(I_t) \in \{0, 1\}$ ，當此像素為背景則為 0，前景則為 1，我們可以靠以下幾種方法來判斷前景背景：(1)比較閾值  $T_F$  與加權值  $\omega_m$ [2]、(2)與正規化後的加權值  $\omega_m/\sigma_m$ [22]、(3)與相似機率(likelihood probability)  $P(I_t)$ (比較是進行在模型更新之後)，或者與  $P(I_t)$  加上模型標籤上的標記當作先決條件導出的事後機率 (posterior probability) 進行比較[3]。而標示出的結果如圖 2-5，前景處為 1，背景處為 0。



圖 2-5 給予前景適當的判斷式子以求出影像中前景的部分

不管 GMM 中數學公式描述的精確性如何，這當中所需要的計算量較為複雜，因此會導致效率低落。原因有三，第一個原因是在第一個式子中的指示函式(indicator function)  $I[.]$ ，需要找到一個最符合輸入像素強度的高斯分布需要一些計算量；第二，模型更新時的每個迭代的迴圈，都會更新  $K$  個高斯的平均值  $\mu_m$  與變異量  $\sigma_m^2$ ，這些更新的計算包含了許多的浮點數加法與乘法運算。第三，這些等式無法用有效率的整數運算且不失精確性的方式來達成。而進來有越來越多的研究文獻著重於將該模型加速，如參考文獻[4]與[5]中，Zivkovic 提出新的估測方式能夠將背景高斯模型數量作動態調整。而接下來 Later, Gorur

與 Amrutur [6]則提出使用整數來模擬浮點數運算。除了在演算法上作改良外，也有文獻[7]使用取樣(sub-sampling)的方式來對 GMM 作加速。



# 第三章 方法

在本篇論文提供一方法改善 GMM 程式運作效率，在 GMM 模型中最為耗時的處理流程便是找尋與目前像素強度值相對應的高斯模型並處理平均值 $\mu$ 與變異量 $\sigma^2$ 。我們使用離散的方法來克服這個處理耗時的流程，進而發展出一套更有效率的模型，即為直方圖背景模型，以下為該方法的流程與詳細介紹。

## 3.1 直方圖背景模型

本篇論文討論的直方圖背景模型是以像素為一個單位進行運算。我們在每個像素中建立模型來描述影像中一個像素截至 $t$ 時間點時的統計資料。而描述的方式我們使用 $N$ 個 Bin 的直方圖，縱軸為每個 Bin 的背景機率值，橫軸為 Bin 的索引值 $k \in \mathbb{N}$ 為正整數，如圖 3-1。每個 Bin 涵蓋某範圍顏色強度值，系統運作

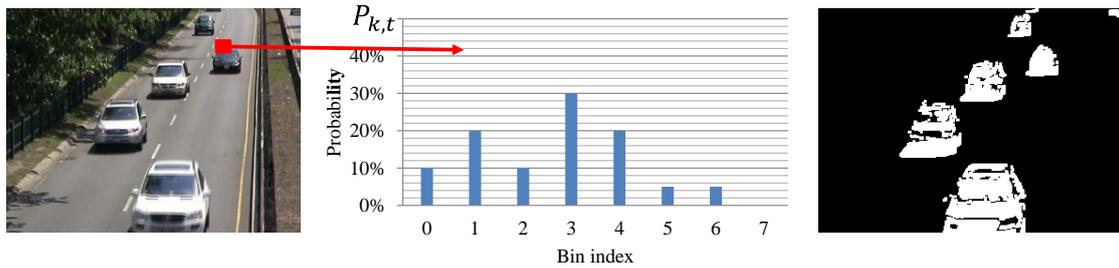


圖 3-1 每個像素都有一個背景模型，該模型為一像素所對應到的直方圖，縱軸為背景機率值，而橫軸為 bin 的索引值。

前需要定義每個 Bin 的寬度 $\Delta b$ ，以及 Bin 的起始位置 $b_0(= 0)$ ，並規定第 $k$ 個 Bin 所劃分的範圍為 $[b_k, b_{k+1})$ ， $b_k = b_0 + k \cdot \Delta b$ 為 Bin 間的分界亮度，每個索引值代表 Bin 所劃分的顏色強度值範圍，如下圖 3-2。

而接下來我們規劃映射函數 $B(I_t): \mathcal{L} \rightarrow \mathcal{B}$ ，代表在時間點 $t$ 時，能夠將顏色強度值 $\mathcal{L} = \{0, \dots, 255\}$ 映設到 Bin 索引值 $\mathcal{B} = \{0, \dots, N - 1\}$ 上，因此 $B(I_t) = k$ ， $k$ 為 Bin 索引值。給了這些變數後，舉例來說：一個 $N = 8$ 的直方圖與顏色強度值

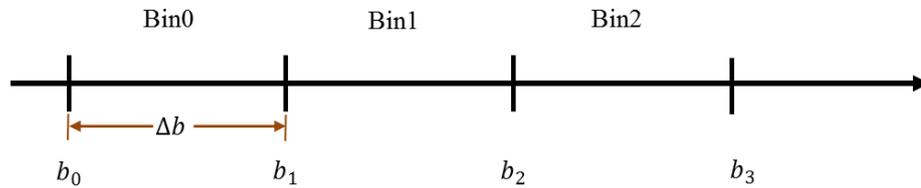


圖 3-2 Bin 索引值化分範圍與起始點 $b_0$ ，而每個 Bin 具有寬度 $\Delta b$ 。

共有 256 色時，Bin-0 代表 $[0, 32)$ ，Bin-1 代表 $[32, 64)$ 等...。而在系統加速上透過函數 $B(I_t)$ ，顏色強度值 $\mathcal{L}$ 與對映的索引值 $\mathcal{B}$ 可以做成一張表格減少計算次數。我們的背景模型可表示為各個 Bin 背景機率值的集合 $P$ ，可表示為式(5)。

$$P = \{P_{k,t} | k \in \{0, \dots, N - 1\}\} \quad (5)$$

上述的 $P_{k,t}$ 為影像中一個像素在第 $t$ 個時間點的第 $k$ 個 Bin 之背景機率值。在連續影像中，我們將每個像素獨立記錄。假設每個時間點可得到該像素的顏色強度值，記錄 $t$ 個時間後我們即可得到每個像素累積至時間點 $t$ 的顏色分佈直方圖，並擁有截至時間點 $t$ 時，每個 Bin 所劃分的顏色出現的頻率，這樣的頻率我們稱之為背景機率值。有了這個直方圖即可對新進入的（第 $t + 1$ 個時間點時）顏色強度值進行比對，比對後可估測第 $t + 1$ 個時間點時該顏色強度值在 $t$ 時間內出現的頻率（背景機率值）。得到 $t + 1$ 時間點時，該像素顏色強度值所屬的背景機率值後，我們可設定一閾值來決定該像素目前的顏色強度值是否為背景。如下圖的兩個像素，綠色框起的 Bin 為目前該像素顏色強度值所對應到的 Bin，當該 Bin 的背景機率值超過指定閾值（紅色虛線），則判定該像素為背景，反之則為前景，如圖 3-3。

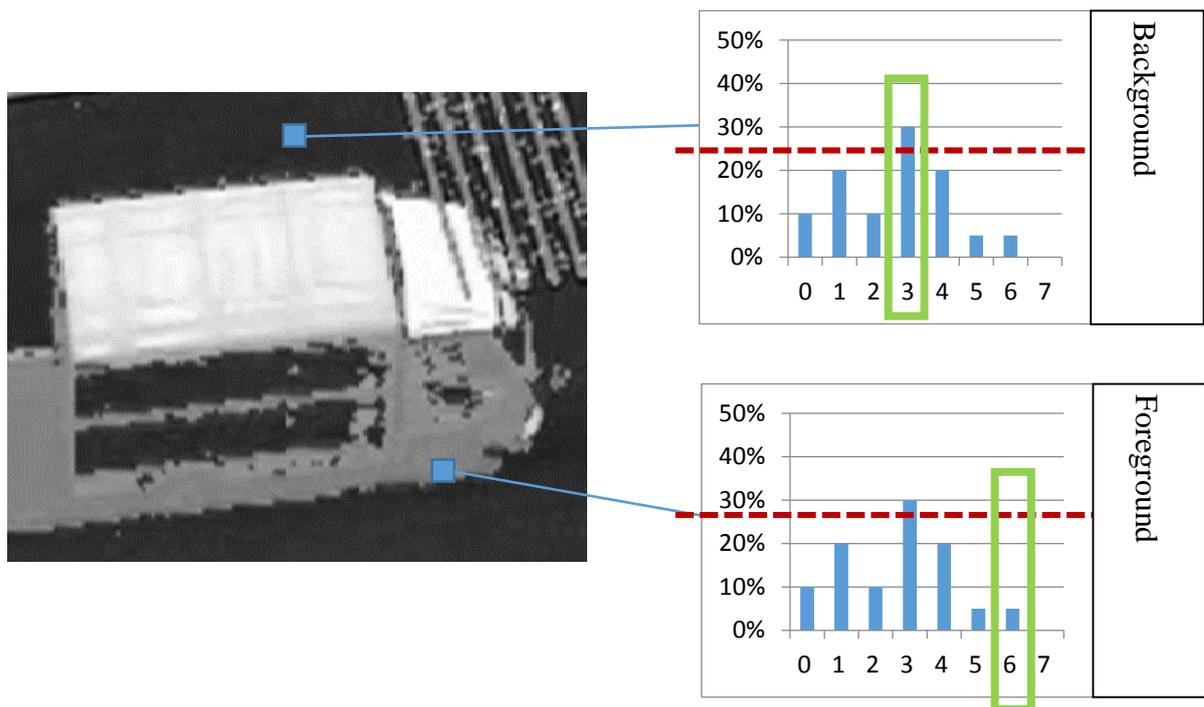


圖 3-3 以每個像素為單位的直方圖背景模型，綠色框線為目前該像素顏色強度值所對應到的 Bin 索引值，紅色虛線為閾值。

如此即為直方圖背景模型的基本精神。而因為背景會隨時間而改變，所以必須隨著時間調整背景模型。以下介紹此模型的運作流程，流程圖如圖 3-4。

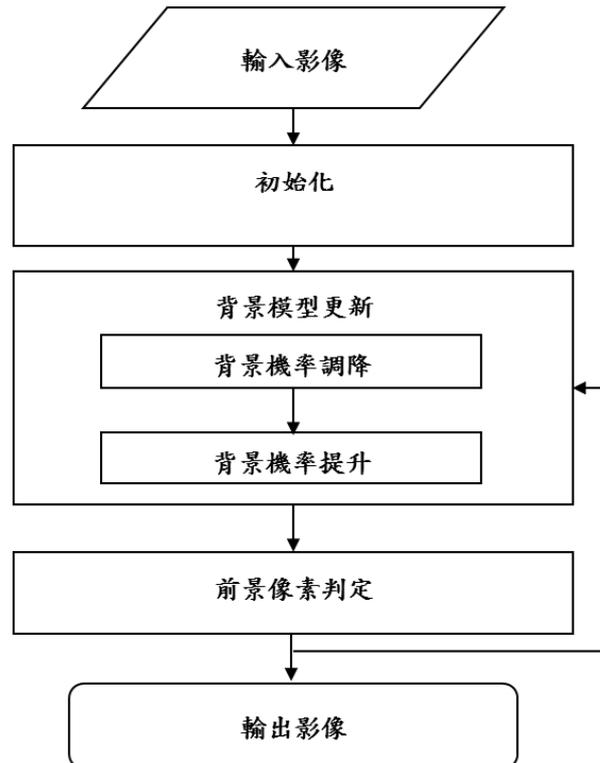


圖 3-4 直方圖背景模型的流程圖，系統會將第一張圖該像素的顏色強度值設定為背景作為初始化，接下來的每一張圖進行背景模型的更新與像素判定。



### 3.1.1 初始化

在第一張影像進入時，某像素的顏色強度值為 $I_{t=0}$ ，我們將此強度值當作該像素的背景，而 $B(I_t)$ 則為該顏色強度所對應的 Bin 索引值，因此 $\forall k \in \{0, \dots, N-1\}$ ，初始化動作如式(6)：

$$P_{k,t=0} = \begin{cases} 1 & \text{if } k = B(I_{t=0}) \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

經過該式的處理後，像素所對應到的 Bin 之背景機率值應如下圖 3-4，綠色框線為目前像素強度值對應到的 bin，第一張的強度值我們將該 bin 背景機率直接設為 100%，如圖 3-5。

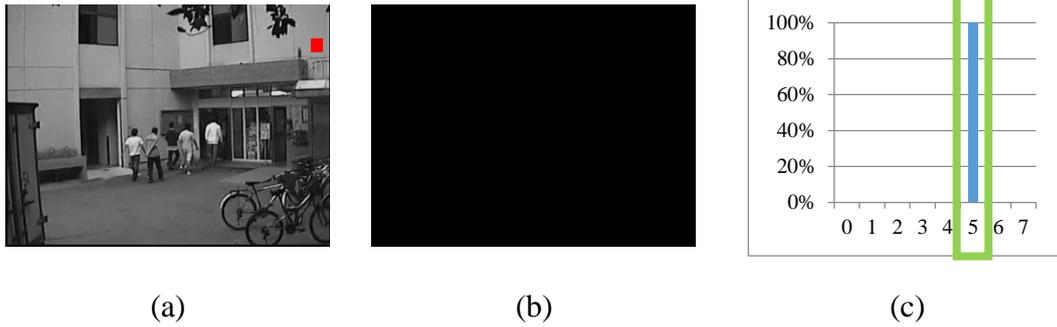


圖 3-5 第一張影像的初始化。圖(a)為原影像，而圖(b)為第一張影像的前景切割結果圖，圖(c)為圖(a)紅點處的直方圖背景模型，初始化令所有像素顏色強度值鎖定到的bin，背景機率皆為100%。

上述步驟將每個像素的第一個接收到的顏色強度值當作是背景，因此該顏色強度值對應的 Bin 背景機率值設為 100%，如此，即完成初始化的動作。

### 3.1.2 背景模型更新

而針對接下來的所有影像，每個時間點都要進行背景模型的更新，而在更新的部份我們將其區分為兩個步驟，背景機率調降與背景機率提升。每個循環先調降每個 bin 的背景機率值接下來再對當前顏色強度值對應到的 bin 進行背景機率提升。如圖 3-6，在背景模型更新時，每個像素的背景模型會先將所有的

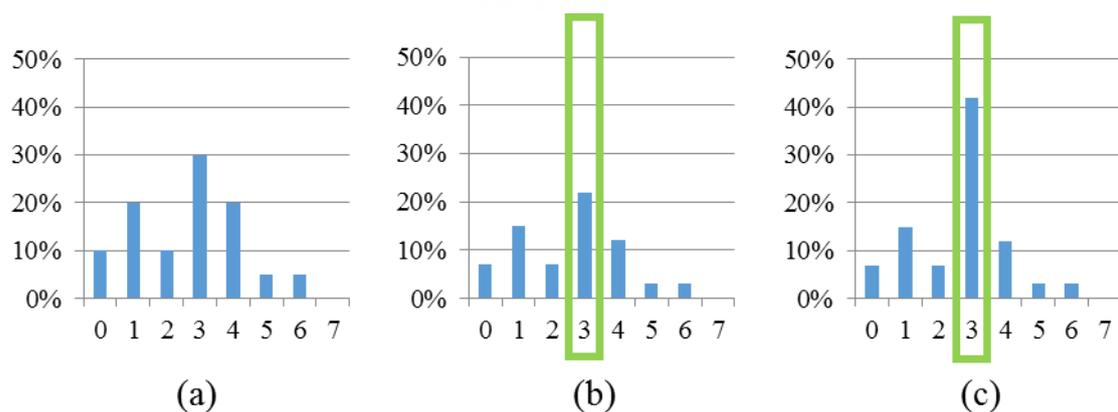


圖 3-6 背景模型更新有兩大步驟，圖(a)為在 $t$ 時間點時某像素的背景模型，在作對應 Bin 背景機率提升之前，我們需將所有的 Bin 進行機率調降，如圖(b)。接下來再作對應 Bin 背景機率提升，如圖(c)。對應 Bin 以綠框表示。

bin 的背景機率值作調降，再對與目前像素顏色強度值相對應的 bin 作背景機率的提升。

上述的解說為背景更新的概念，以下詳述背景模型更新中，機率調降與提升此兩步驟。第一階段的背景機率調降：由於越不常出現的對應 Bin 其背景機率值應該要越來越小，因此我們將每個 Bin 的背景機率值在每個循環中降低。調降公式如下

$$P_{k,t} = \beta P_{k,t-1}, \forall k, \quad (7)$$

我們將每個 Bin 的背景機率值乘上一個調降因子  $\beta$ ， $\beta \in [0,1]$ 。此步驟將全部的 Bin 調降背景機率值。接下來第二階段為對應 Bin 背景機率提升，在一個像素上將目前時間點的顏色強度值所對應到的 Bin 作背景機率提升的動作，因此兩步驟相結合即為下列公式

$$P_{k,t} = \beta P_{k,t-1} + \alpha I[k = B(I_t)], \forall k, \quad (8)$$

與公式(4)相比後面多加了一段機率提升的加總。而此處的  $\alpha = 1 - \beta$ ， $I[]$  為指示 (indicator) 函數，當函數中的式子成立則為 1。意即當  $k$  為顏色強度值對應到的 Bin 索引值  $B(I_t)$  時，整個背景機率值才會加上一個  $\alpha$  值。設定這樣的因子是為了最後步驟不需額外再將 Bin 背景機率值作一般化(Normalize)的動作，用這個手續可保證機率調降與提升兩步驟後所有 Bin 相加仍舊會等於 100%。

### 3.1.3 前景像素判定

接下來，在背景模型更新後，我們可給予系統閾值  $T_F$  來決定該像素是否為前景，判定方式如下

$$L(I_t) = \begin{cases} 0, & \text{if } P_{B(I_t),t} \geq T_F \\ 1, & \text{otherwise} \end{cases} \quad (9)$$

此處的的閾值  $T_F$  決定了該像素是否為前景或背景，當該對應 Bin 機率值大於此閾值，則判定為背景，反之為前景。如圖 3-7，每個像素皆會依照閾值來對目前像素的顏色強度值所對應到的 Bin 進行判定，並在一個二值化的圖片上標上白

色記號。以上步驟即為直方圖模型一個像素判斷前景與背景的流程。

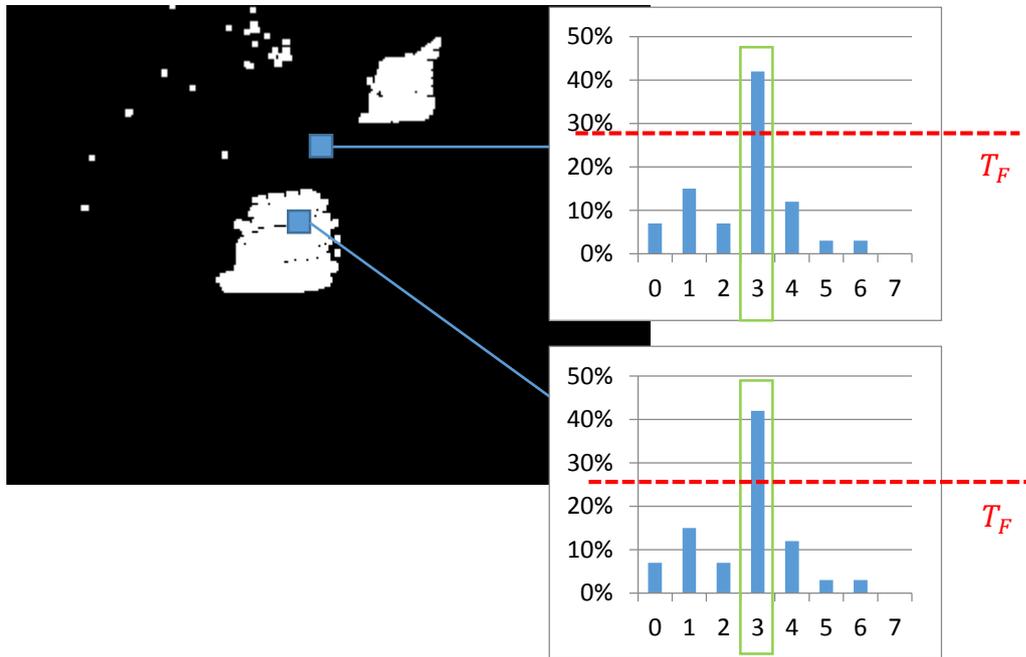


圖 3-7：背景模型的更新後，在前景像素判定步驟每個像素皆會將目前像素對應到的 Bin（以綠色矩形框起）之背景機率值，與閾值 $T_F$ 相比較，高於此閾值則判定為背景（圖中黑色部分），反之則標上白色記號以示為前景。



### 3.2 多重映射

如此一個系統即為完整的直方圖背景模型。然而，在這樣的模型下會因為記錄離散化的資訊而產生一個問題。現實的情況下可能會因為雜訊，使得某像素的顏色強度值在 Bin 的劃分邊緣震盪，此時的機率提升會被分化掉，我們可能會低估該 Bin 為背景的機率。例如：一個 8-Bin 的背景模型，如果一個像素的顏色強度值因為雜訊而在 30 與 34 兩值來回震盪，以顏色強度值 32 為一單位的劃分情況下，我們希望 30 與 34 這兩個相近的值會落在同一個 Bin 中，但事實上 30 與 34 分別落再 Bin-0 與 Bin-1 兩個 Bin 上，如圖 3-8。圖 3-8 描述當兩

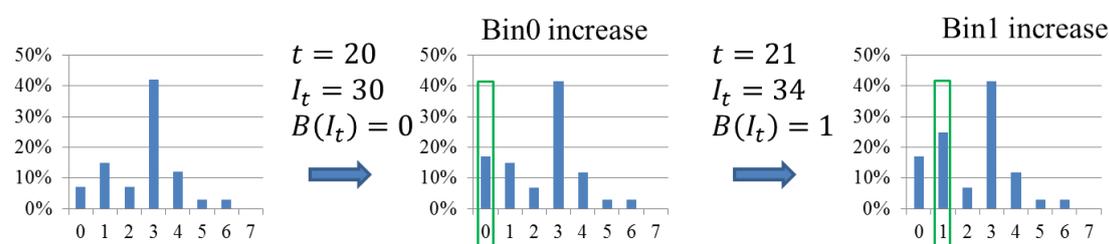


圖 3-8 上圖為兩個時間點時，相似的顏色強度值可能會受到少量雜訊的干擾，使原先該被分配到相同 Bin 的顏色值，被判定為兩個不同的 Bin 的情況。

個時間點  $t = 20$  時，某像素當前顏色強度值為 30，對應到 Bin0，依照直方圖背景的流程，應該先調降所有的 Bin，接下來提升相對應的 Bin0 之背景機率值。接下來下個時間點  $t = 21$ ，因為雜訊的影響，顏色值變為 34。同樣的物體因為剛好落在分界線附近，所以被分配到兩個不同的 Bin 上導致可能會被當作是兩個不同的物體，在背景機率值的增加上也可能會被分化掉。

為解決此問題，我們再將我們的直方圖模型加入一些改變。我們將一個像素由原先對應到的 1 個 Bin 改為 2 個 Bin，一個像素強度值除對應到原先所設定好的 Bin 外，我們亦將離該強度值最近的第二個 Bin，設為對應 Bin。則由原先的  $B(I_t) = k$  變為  $B(I_t) = \{k_0, k_1\}$ ，原先所對應到的一個 bin 我們當作  $k_0$ 。我們設定次要對應 Bin 索引值  $k_1$  示意圖如圖 3-9，而目前像素顏色強度值所對應到的 Bin 索引值的計算方式為下式

$$k_1 = \begin{cases} k_0 - 1, & \text{if } I_t \in [b_{k_0}, b_{k_0} + \frac{\Delta b}{2}) \\ k_0 + 1, & \text{if } I_t \in [b_{k_0} + \frac{\Delta b}{2}, b_{k_0+1}) \end{cases} \quad (10)$$

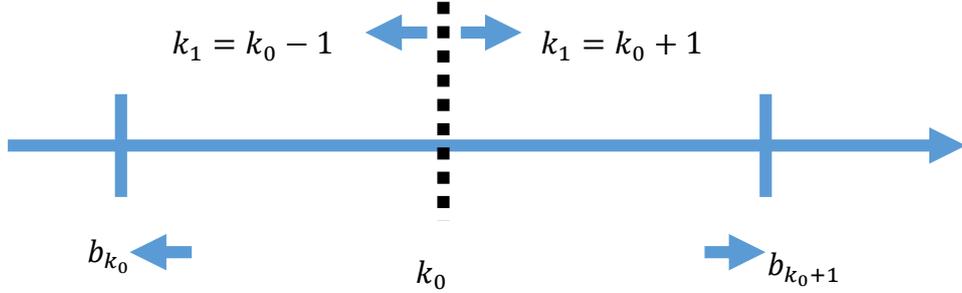


圖 3-9  $k_0$  為原先的 Bin 畫分範圍，加入多重映射後，除  $k_0$  外，我們令  $k_1$  為額外的映射 Bin。

理解了這個原理後，我們僅需修正圖 3-4 背景模型更新與前景像素判定兩步驟，改進後的流程圖如圖 3-10。

### 3.2.1 雙重映射的背景模型更新

由於機率調降是每個 Bin 都需要去調整，無關對應 Bin 增加，因此機率調降此步驟我們無須改變，僅需修改機率提升步驟。修改公式如下

$$P_{k_m,t} = \beta P_{k_m,t-1} + \alpha_m, \forall m \in \{0,1\} \quad (11)$$

原先的因子  $\alpha$  改為  $\alpha_0$  與  $\alpha_1$ ，而  $\alpha_1$  為次要對應的 Bin，由於  $\alpha_1$  為非直接對應的 Bin，重要性不比  $\alpha_0$ 。因此在加權上我們令  $\alpha_1 = 0.5\alpha_0$ 。

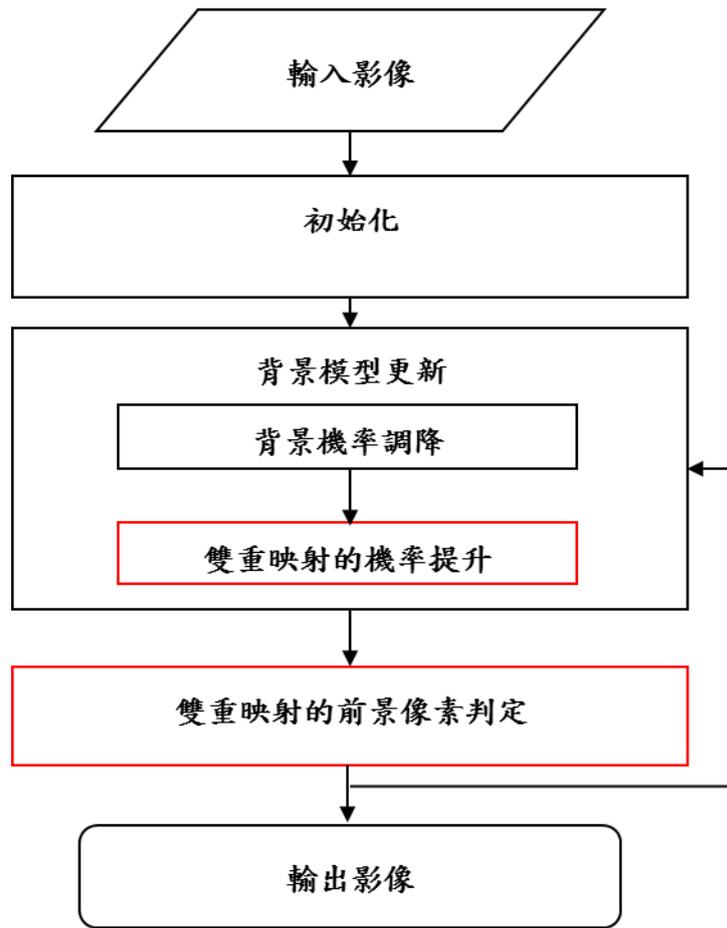


圖 3-10 加入雙重映射的方法，我們修改原圖 3-4 的流程圖，紅色部分為有修改處

### 3.2.2 雙重映射的前景像素判定

在更新上作新的改變後最後一步驟前景判別也需被改寫。一個像素是否為前景之計算方式更改如下

$$\hat{P}_t = \arg_{m \in B(I_t)} \max P_{k_m, t} = \max\{P_{k_0, t}, P_{k_1, t}\} \quad (12)$$

一個像素由於對應到兩個 Bin，因此我們將對應的背景機率改為  $\hat{P}_t$ ，上述式子在計算該像素是否為前景或背景時，會檢查當下像素強度值所對應到的兩個 Bin 的背景機率值，當兩 Bin 只要有一個 Bin 超過閾值  $T_F$ ，則判定該像素為背景像素。舉例來說，如圖 3-13，一個  $N = 8$  的直方圖，在  $t = 21$  的時間點出現了  $I_t = 30$  的像素強度值，該強度值對應到  $k_0$  與  $k_1$ ，此兩 Bin 中，背景機率值最大者為 Bin1，該 Bin 超過閾值  $T_F$ ，因此系統判定該像素為背景像素，示意圖如圖

3-11。

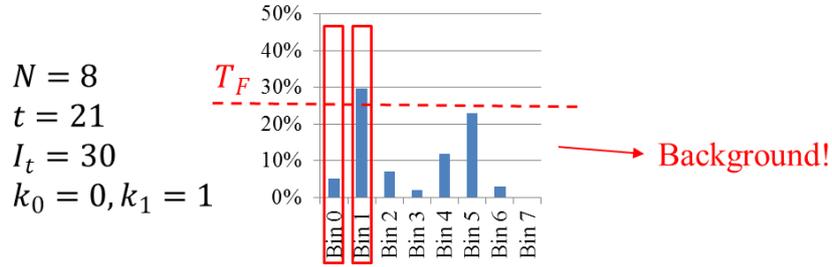


圖 3-11 系統以背景機率較大者的 Bin 作為判定背景像素的依據

### 3.3 系統加速

增加多重映射後，因為雜訊而跳動的顏色強度值可被放入相同的 Bin 中進而降低誤判率。接下來我們來探討這個系統的計算效率，機率調降階段每個像素每個時間點由於需要調降全部的 Bin，因此在機率調降此步驟有多少 Bin 就得重新計算多少次背景機率值。由於決定目前像素為前景背景的關鍵在於該像素對應的兩個 Bin，因此在新的系統中，我們希望系統不需要每個循環都調降所有 Bin，僅針對目前像素強度值所對應到的兩個 Bin 作以下兩步驟：(1)觀察相對應的兩個 Bin 已有幾次沒作調降，並先將該兩 Bin 的機率值作調降動作與 (2)提升相對應兩 Bin 的背景機率值。

這樣的方法需要用到兩個計數器，分別是計數器  $\mathcal{L} = \{l_{k,t} | k \in \{0, \dots, N - 1\}\}$ ， $t$  為目前的時間點而  $k$  則為 Bin 的索引值，記錄每個 Bin 最後一次被更新的時間點，每個 Bin 都有相對應的計數器  $l_{k,t}$  計數器的初始值為  $l_{k,t=0} = 0, \forall k$ 。另一個計數器為系統計數器  $g_t$ ，該計數器初始狀態為  $g_t = 0$ ，會不斷往上累加為  $g_{t+1} = (g_t + 1)$ ，以記錄目前時間點。利用這兩個計數器根據現在與最後一次被更新的時間點的差值  $d$  決定該 Bin 的背景機率值該調降幾次，如下式

$$P_{k_m,t} = \beta^d P_{k_m,t-1}, \quad d = (g_t - l_{k_m,t-1}) \quad (13)$$

在此處的 $k_m$ 為相對應的 Bin 索引值，而上式的示意圖則如圖 3-12，我們利用代

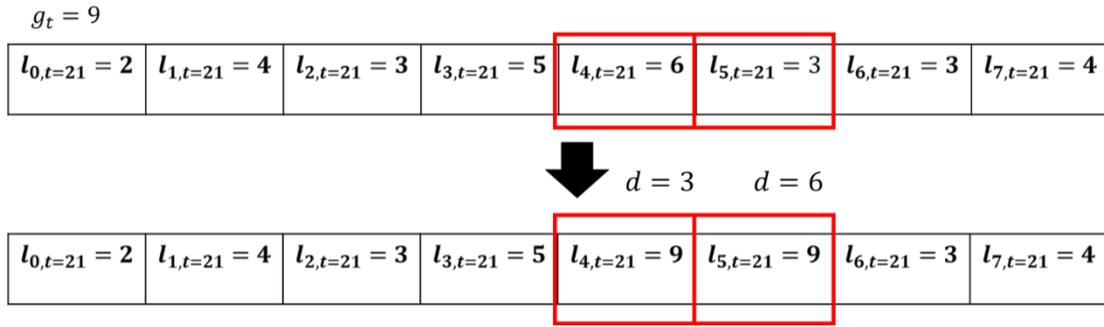


圖 3-12 代表目前時間點的 $g_t$ 與最後更新時間點的 $l_{k,t}$ 相減可得到該 Bin 需要調降的次數 $d$

表目前時間點的計數器 $g_t$ ，與每個 Bin 最後一次更新的計數器 $l_{k,t}$ 的差值來算出該 Bin 距離上次更新已經過多少個循環，這個數字我們以 $d$ 來表示。

這樣的方法省去了每個循環都要調降所有 Bin 的計算，但衍生出一個新的問題：當某個 Bin 的 $l_{k,t}$ 與系統計數器 $g_t$ 相差太多，會成為另外一個運算上的負擔。舉例來說：一個 $N = 8$ 的直方圖模型，該模型所對應的 $l_{k,t}$ 如圖 3-13 所述，此時系統需要將第四個 Bin 調降 88 次，因此 $\beta^d$ 的 $d$ 值會高達 88。

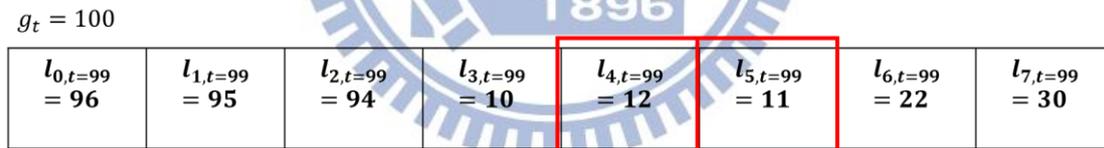


圖 3-13 代表目前時間點的 $g_t$ 與最後更新時間點的 $l_{k,t}$ 相減可得到該 Bin 需要調降的次數 $d$

我們希望將此方法再精簡，因此在每個循環調降相對應的 Bin 之前，再加上一步驟—週期性背景機率調降。該步驟需將 $g_t$ 的累加規則改為下式 $g_{t+1} = (g_t + 1) \bmod N$ 。並在週期性背景機率調降時檢查第 $g_t$ 個 Bin，算出該 Bin 的 $d$ 並調降。會在背景更新的一開始循序的選擇一個 Bin，每 $N$ 個循環又會輪回到原來的 Bin，檢查該 Bin 與目前時間點的差 $d$ ，並調降該 Bin 的背景機率值，接著才檢查相對應的 Bin 有多久未更新，調降兩相對應 Bin 後再作提升。接著會詳述改良後系統加速的流程，而改良後的方法有個優點，週期性調降背景機率可

保證每次找到相對應的兩 Bin 時的  $d \leq N$ ，有了這樣的條件後， $\beta^d$  可建一個查閱而不需計算。而改良後的直方圖背景模型流程圖如圖 3-14。在雙重映射的方法之後，改變背景模型更新步驟，最後在使用雙重映射時的像素判定方法如式 (12)，即為本論文所提出的無母數直方圖背景模型。而接著詳述背景更新的三個步驟。

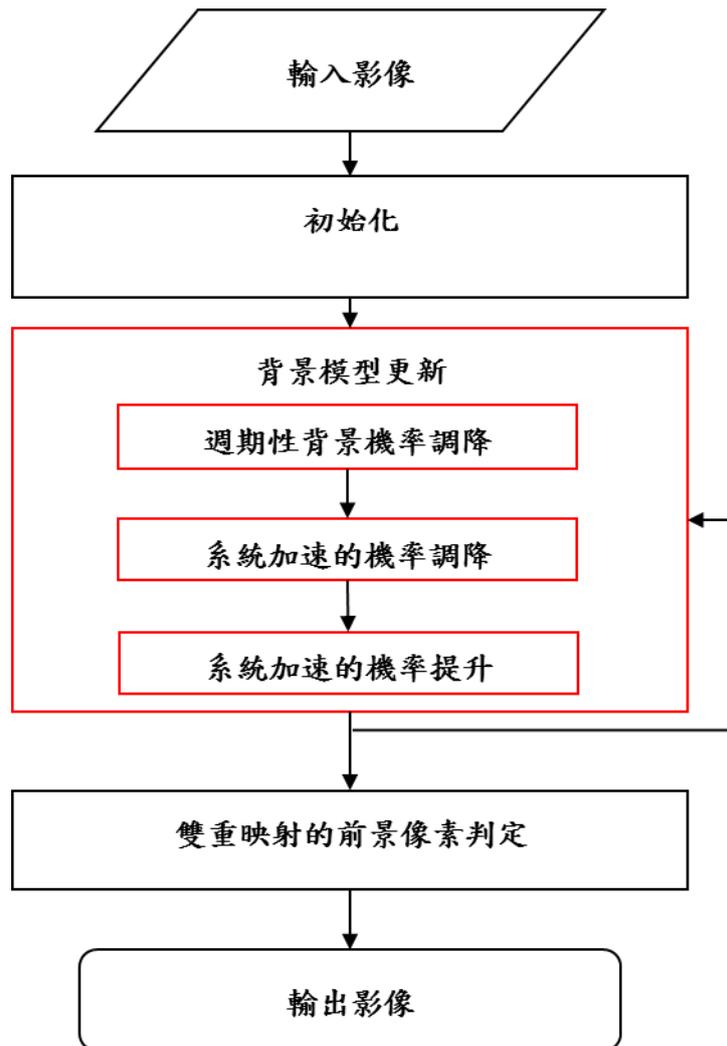


圖 3-14 經過系統加速後的流程圖，與雙重映射的流程圖相比，修改過的部分為背景模型更新處(紅色框)

### 3.3.1 週期性背景機率調降

系統加速後的直方圖背景模型，會有系統計數器 $g_t$ 記錄目前時間，而其更新公式如下式：

$$g_{t+1} = (g_t + 1) \bmod N$$

在週期性背景機率調降此步驟，會進行下列步驟，(1)挑第 $g_t$ 個 Bin 的背景機率值作調降的動作。該指定 Bin 的調降公式為 $P_{g_t,t} = \beta^d P_{g_t,t-1}$ ，其中指數 $d$ 的計算方式如下

$$d = \begin{cases} N, & \text{if } g_t = l_{g_t,t-1} \\ (g_t - l_{g_t,t-1} + N) \bmod N & \text{otherwise} \end{cases} \quad (13)$$

(2)調降後的指定 Bin 更新計數器為 $l_{g_t,t} = g_t$ 。示意圖如圖 3-15。而由於先前所述在此步驟不可能會有 Bin 超過 $N$ 次沒更新，因此 $d$ 只有 $N$ 種可能，所以 $\beta^d$ 我們可以作一個查詢表方便計算。

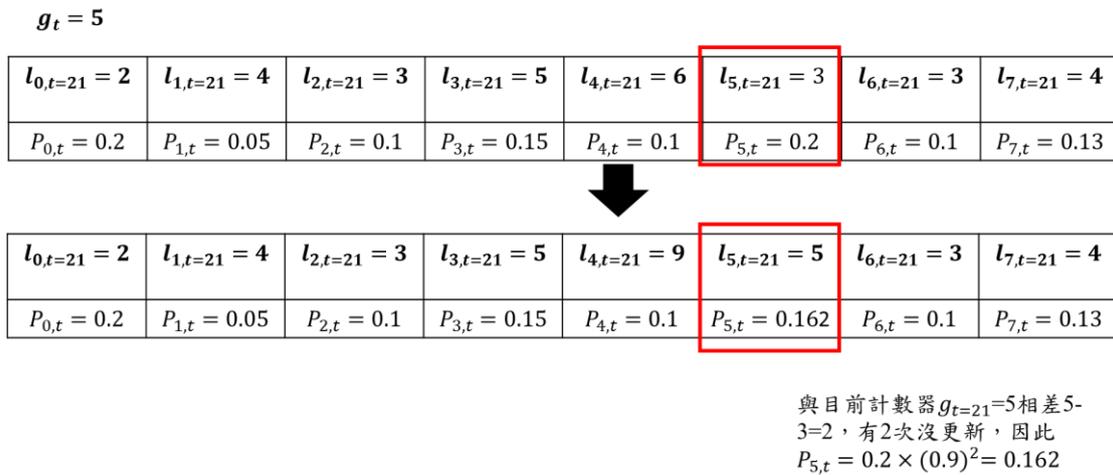


圖 3-15 週期性調降兩步驟：(1) 根據 $g_t$ 與 $l_{g_t,t}$ 來調降第 $g_t$ 個 Bin 的背景機率值與(2)將 $l_{g_t,t}$ 設為目前系統計數器的值 $g_t$

### 3.3.2 系統加速的機率調降與系統加速的機率提升

接下來處理目前時間點 $t$ ，與像素顏色強度值相對應的兩個 Bin 的機率調降與機率提升。假設相對應的 Bin 索引值為 $k_0$ 與 $k_1$ 。在機率提升步驟前我們需要

檢查此兩 Bin 多久沒更新，先行調降，如下公式。

$$P_{k,t} = \begin{cases} P_{k,t-1} \cdot \beta^d & \text{if } k \in \{k_0(\neq g_t), k_1(\neq g_t)\} \\ P_{k,t-1} & \text{otherwise} \end{cases} \quad (14)$$

而完成調降後，則進行兩對應 Bin 機率提升，機率提升的方式則未做更動，如同前面多重映射中式(11)所述。

整理這個系統加速的章節與計數器的概念，在背景模型更新部分我們重新舉一個完整的例子如圖 3-16：圖(a)描述在時間點  $t=21$  時，某像素的所有 Bin 背景機率分佈。由於目前  $g_t = 5$ ，所以在該時間點時我們檢查 Bin 索引值為 5 的 Bin。取出該 Bin 的計數器  $C$  與系統計數器  $g_t$  相減，並調降該 Bin 背景機率值  $P_{5,t}$ ，得到(b)。而在右側兩張圖則代表著背景調降與調升步驟。由於目前該像素背景強度值為 30，因此相對應的 Bin 為 Bin0 與 Bin1，如圖(c)。而在作背景調升之前，我們需要先確定該兩 Bin 有多久沒更新，我們需要作背景調降動作，

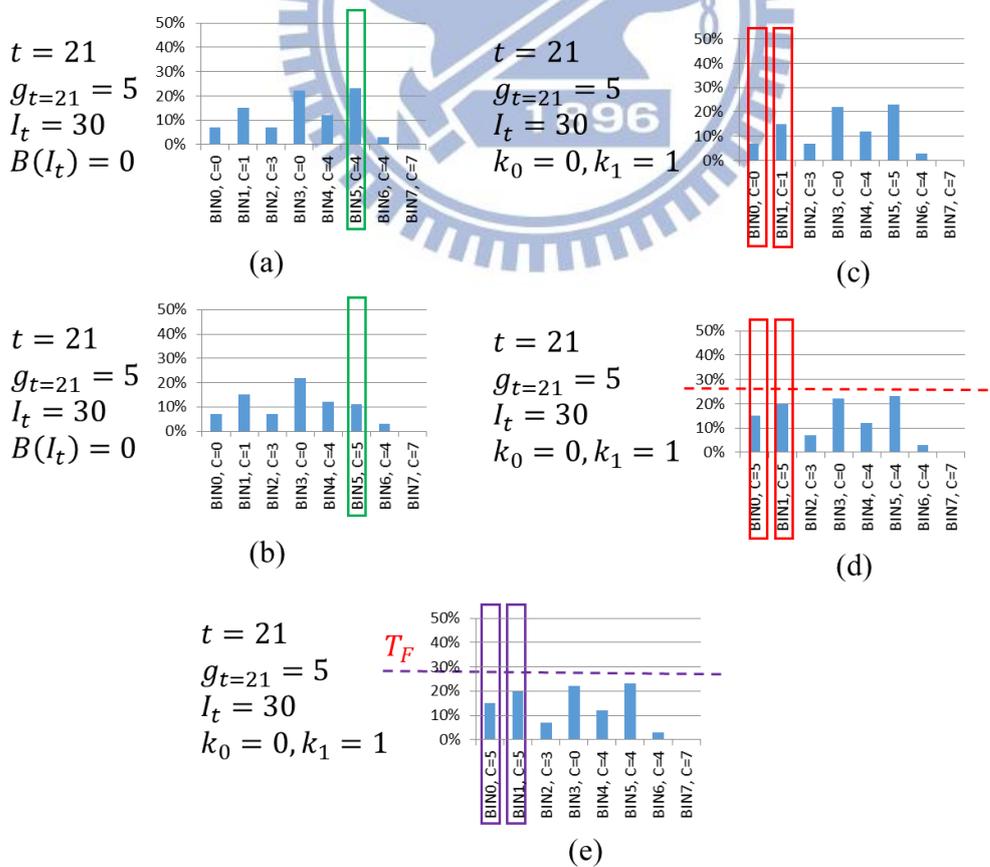
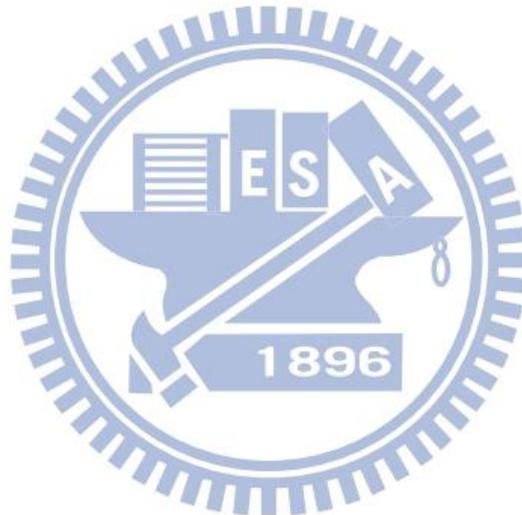


圖 3-16 系統加速後的背景模型更新與前景像素判定的例子

調降後再進行背景機率值的提升如圖(d)，在兩 Bin 中找尋背景機率最高者，與閾值 $T_F$ 相比較，以此例來說，背景機率值最大的 Bin1，低於 $T_F$ ，因此該時間點該像素被判定為前景，如圖(e)。如此變完成了整個背景模型的更新並且做前景像素判定的步驟，而此方法的執行速度與精確度將於下個章節討論。



## 第四章 實驗結果

而在實驗上我們分為兩個部分，第一段我們使用常用的樣本資料來測試我們的方法與一般常用的 GMM 之速度與精確度比較。第二段的部分我們將本論文的方法推廣至應用層面，在應用上我們將本實驗的前景切割效果使用在鐵道偵測、夜視的紅外線打光器與多攝影機人物定位上。

### 4.1 精確度與處理速度

在效能評估上，我們實作兩種版本的演算法進行實驗，分別為浮點數另一個與定點(fixed-point)的運算。並將這兩個實作方式與 GMM [1]進行比較，除此之外也與 Zivkovic 版本的 GMM [2](在本實驗中縮寫為 Z-GMM)進行比較。每個程式模組皆由 MS Visual C++進行編譯並且使用 single process 執行，整個實驗所用的設備為 Intel Core i7-2640M(2.8GHz)。而在實驗數據上，我們將兩個 GMM 模組的參數設定為  $K = 3$ ， $T_\sigma = 3$ ，且  $\eta = 0.01$ ，並且在我們的方法上參數設定為  $\Delta b = 256/N$ ， $\alpha_0 = 0.01$ ， $\alpha_1 = \alpha_0/2$ ， $\beta = 1 - \alpha_0$ 。而前景的閾值在此兩方法上皆設定為  $T_F = 0.25$ 。

為了觀察實際監控系統的前景物件分割效能，這兩個方法上我們使用侵蝕(Erosion)與膨脹(Dilation)作為後處理。如表 4-1 所述，我們使用了許多的影像資料來做為我們評估效能的依據。而在圖 4-1 中的第一行為測試影像資料的快照。如同圖一所示，本篇實驗的前景切割結果是可以比得上 GMM 的。

如同表 4-2 所提到，本研究的實作定點的版本能夠達到最佳的效率，此處理效率比 Z-GMM 快 1.3 倍而比 GMM 快 3.7 倍。而這樣的結果是因為我們使用記憶體空間來換取處理時間。在程式執行期間，會有較頻繁的記憶體空間存取的現象。主記憶體與 CPU 之間的資料傳輸不容易在演算法分析實評估，這樣的現象其實正是本方法處理速度的瓶頸。

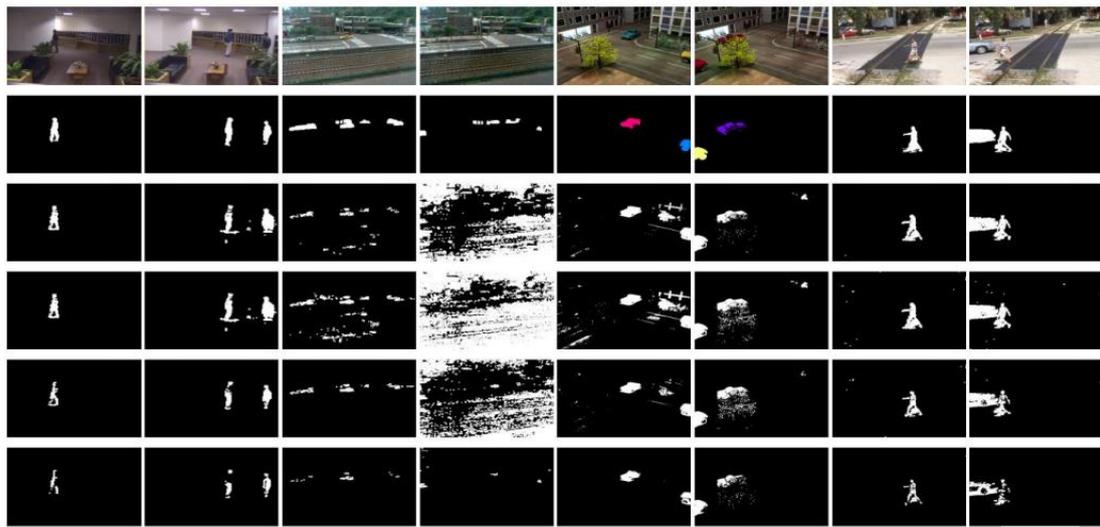
表 4-1 測試影片特性

影像	張數	解析度	場景	特色
1[8]	1546	160 × 128	室內	突然間出現光線變化
2[2]	760	320 × 240	室外	快速光線變化
3[9]	600	800 × 600	室外	人工模擬場景
4[10]	500	360 × 240	室外	攝影機抖動

表 4-2 背景模型計算速度比較，以 GMM 的速度作為基準速度(100%)進行比較。單位：

FPS

GMM( $K = 3$ )	25.34 (100%)	5.93 (100%)	100%
Z-GMM( $K = 3$ )	68.77 (273%)	17.03 (287%)	280%
Histogram-floating ( $N = 16$ )	56.97 (224%)	13.99 (236%)	230%
Histogram-floating ( $N = 8$ )	65.43 (258%)	16.75 (282%)	270%
Histogram ( $N = 16$ )	81.62 (322%)	21.22 (358%)	340%
Histogram ( $N = 8$ )	88.76 (350%)	23.25 (392%)	371%



(a) (b) (c) (d) (e) (f) (g) (h)

圖 4-1 前景切割結果比較。列 1：輸入影像；列 2：標準前景切割結果；列 3-6：GMM、ZGMM、我們的方法(Fixed-point 實作、N = 16)以及我們的方法(Fixed-point 實作、N = 32)之前景切割結果(a)&(b)：video 1. (c)&(d):video 2. (e)&(f): video 3. (g)&(h)：video 4.

表 4-3 測試影片特性

影像	張數	解析度	場景	描述
Highway	1546	360 × 240	室外	高速公路車流
Pedestrians	760	360 × 240	室外	街上人群行走
PETS2006	600	512 × 720	室外	地下鐵人潮流動
Office	500	360 × 240	室內	辦公室內

除比較速度外，我們亦與許多的 dataset 作比較。表 4-3 為我們進行比較本方法與 GMM 的精確度實驗所使用的 CDW 2012 [11]的樣本描述。而表 4-4 與表 4-5 則為本論文所提出的方法，分為 Bin 寬度為 8, 16 與 32 版本，與原始 GMM，ZGMM 的比較。比較的精確度算法使用查全率(Recall)與查準率

(Precision)如下式

$$\text{Recall} = \frac{\#correct}{\#correct + \#miss\ detection} \quad (15)$$

$$\text{Precision} = \frac{\#correct}{\#correct + \#false\ alarm} \quad (16)$$

程式的執行速度與精確度比較如表 4-4 與表 4-5。可從實驗結果觀察出本實驗與原始 GMM 相比，速度上得到了大幅度的提升，而在精確度上  $N = 16$  與  $N = 32$  的版本也可得到與 GMM 相似的效果。由於我們所提出的方法在判定前景時會挑選兩個 Bin 中擁有較大的背景機率者取閾值並做判定，比較不易有應為背景卻被判定為前景的現象，因此在 false alarm 的數目上較小，所以在查準率 (Precision) 上較高，而相反的，此方法也較容易讓前景的像素判定為背景，因此在查全率上會比較低。而與加強後的 ZGMM 相比，在速度上也獲得了一定程

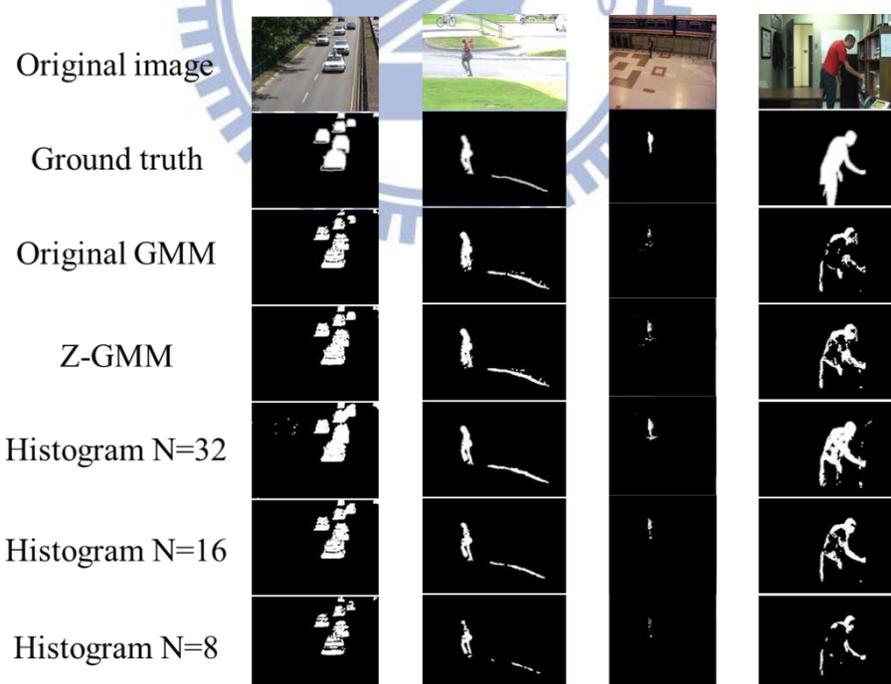


圖 4-2 前景切割結果精確度比較。列 1：輸入影像；列 2：標準前景切割結果；列 3-7：GMM、ZGMM、我們的方法(Fixed-point 實作、 $N = 32$ )、我們的方法(Fixed-point 實作、 $N = 16$ )及我們的方法(Fixed-point 實作、 $N = 8$ )之前景切割結果。

度的改善。值得一提的是，在 Office 的資料上，該影片有一段時間前景物體靜止不動，因此當人物慢慢被當作背景時，則誤判率會大大的提升。而本實驗高於其它兩模型的原因可能為高斯模型對於前景所建立新的高斯模型，其預估的變異量初始化較大，本實驗的 Bin 寬度較小，因此在  $N = 16$  與  $N = 32$  對於顏色變化的敏銳度較高，遇到人物的停留（人物仍有輕微的晃動）的時候，收斂較慢。而比較的資料在視覺化呈現上列於圖 4-2。

表 4-4 本方法在精確度與速度上和 GMM, Z-GMM 作比較

	Highway dataset 320*240			PEDESTRIANS dataset 360*240		
	Precision	Recall	FPS	Precision	Recall	FPS
Original GMM	0.73	0.73	103	0.72	0.83	108
Z-GMM	0.72	0.73	287	0.71	0.83	291
Histogram $N = 32$	0.77	0.72	289	0.78	0.81	272
Histogram $N = 16$	0.88	0.56	366	0.91	0.72	357
Histogram $N = 8$	0.95	0.30	404	0.96	0.57	374

表 4-5 本方法在精確度與速度上和 GMM, Z-GMM 作比較

	PET2006 dataset 512*720			OFFICE dataset 360*240		
	Precision	Recall	FPS	Precision	Recall	FPS
Original GMM	0.73	0.73	19	0.69	0.28	100
Z-GMM	0.72	0.73	72	0.71	0.36	282
Histogram $N = 32$	0.77	0.72	62	0.73	0.57	263
Histogram $N = 16$	0.88	0.56	88	0.83	0.39	325
Histogram $N = 8$	0.95	0.30	102	0.83	0.24	341

## 4.2 前景切割應用

在應用上我們將本論文使用的方法使用在鐵道違規偵測、紅外線打光器夜視功能、多攝影機人物定位上，並與原先的方法作比較。

### 4.2.1 鐵道違規偵測系統：

首先，簡單介紹鐵道違規偵測系統。在鐵路的攝影機應用上，我們常使用攝影機來監視目前行人或汽機車違規的情形，而在論文 Vision-based Detection for Railway Level Crossing Violations 中，該論文使用原始 GMM 來截取鐵道影像中的移動物，並經過許多的處理找出違規的行人與車輛，如圖 4-3。我們將該 OpenCV 版本為 2.4，在 100 組影像中找出違規的情況。該 100 組的影片中有 7 組屬於違規影片，我們比較 OpenCV 所使用的 GMM，與本論文提出的方法。並將本論文的 Histogram 的 Bin 寬度分為寬度 8 與寬度 16 兩種。OpenCV 所採用的 GMM[14]，參數採用  $K = 5$ ， $T_{\sigma} = 2.5$ ，且  $\eta = 0.01$ ，而本論文的參數採用  $\alpha_0 = 0.01$ ， $\alpha_1 = \alpha_0/2$ ， $\beta = 1 - \alpha_0$ ，並使用兩種 Bin 的寬度，一組為 8，另

一組為 16 兩寬度進行比較。而兩種 model 取出違規影片的實驗結果如表 4-6。我們發現在置換成直方圖背景模型下，偵測出來的判斷率會近似或更勝於使用原始 GMM 的判斷率，而我們的估測方式如式(15)與(16)。



圖 4-3 使用背景模型抓出某區段時間內車留路徑與截取違規路徑，圖(a)為某區段內所有被截取出來的車輛、行人移動路徑，而圖(b)則保留違規路徑部分。

表 4-6 鐵道違規偵測下使用原始 GMM 與本方法在精確度上的比較。

	偵測遺漏	誤判	Precision	Recall
OpenCV-GMM[14]	1	1	87.5%	87.5%
HistogramW16	0	1	100%	87.5%
HistogramW32	1	1	87.5%	87.5%

而置換成本論文的背景模型後在速度上也得到大幅度的提升，如下表 4-7。

表 4-7 鐵道違規偵測下使用原始 GMM 與本方法在速度上的比較。單位：FPS

背景模型	速度
OpenCV-GMM[14]	96.37
HistogramW16	264.12
HistogramW32	282.19

## 4.2.2 紅外線打光器夜視系統

在夜間影像監控系統中，適當的照明對於影像品質扮演相當重要的角色。就一般紅外線打光器而言，因受限於亮度固定，可能因亮度不足造成遠距離的物體難以辨識；而光源充足的情形下，過於接近攝影機的物體則會出現影像過曝的現象，以致前景影像品質低落。

該論文中，針對新型多重亮度紅外線打光器產生的不同亮度階層之影像進行影片摘要。該論文利用高斯混和模型對各個亮度階層的影像進行前景擷取，我們參考該論文多重亮度紅外線打光器之影片摘要的方法，並且將該方法原先使用的 GMM 背景模型，換成本論文的背景模型。

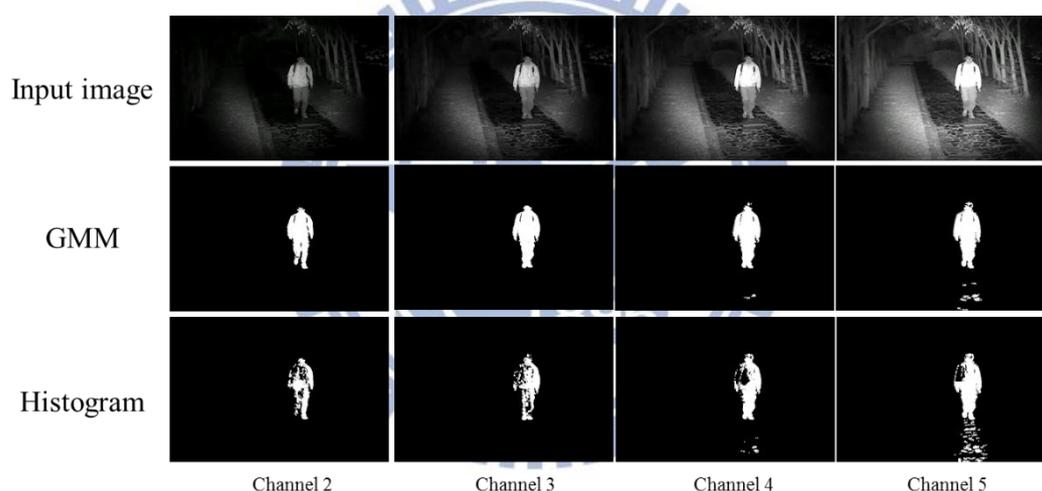


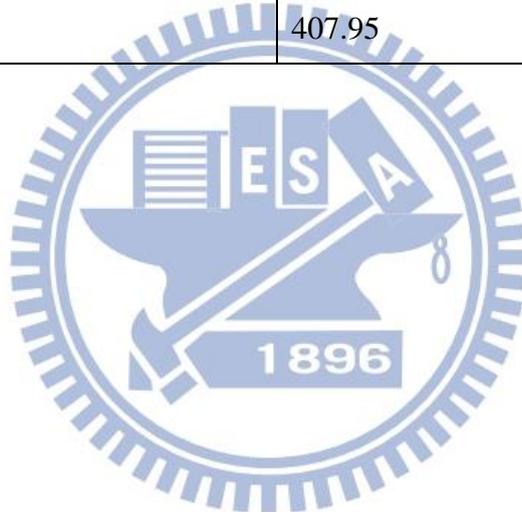
圖 4-4 紅外線打光器偵測較適當距離的前景，擷取前景 4 個頻道為紅外線打光器由暗至亮的四種亮度，橫排為輸入影像與兩模型的背景相減法之輸出影像

實驗結果如圖 4-4 呈現，由於多重紅外線打光器的特性，會在連續時間中打出不同的亮度，因此我們可在同一組圖中，找到時間接近的四種亮度進行比較。而在截取前景上，我們將結果分為三種狀況，依序為遠前景，中前景與近前景，以比較在不同的距離下兩種背景模型的執行結果，在此實驗中，本論文的方法所使用的參數設定為  $N = 32$ ，而四個頻道所使用到的學習率由 channel2-channel5 分別為  $\alpha_0 = \{0.016, 0.013, 0.010, 0.007\}$ ， $\alpha_1 = \alpha_0/2$  and  $\beta = 1 - \alpha_0 -$

$\alpha_1$ ；而高斯模型使用的參數為。  $K = 3$ ,  $T_\sigma = 2.5$ , and 學習率由 channel2-channel5 分別為  $\eta = \{0.016, 0.013, 0.010, 0.007\}$ ，由於在前景的部分該論文所使用到的 GMM [2]在前景判定上有，，能提取出比原作者之 GMM 更精細的結果。而計算速度呈現如表 4-8，我們在提取前景上的速度上也得到了提升。

表 4-8 紅外線打光器前景切割下使用 GMM 與本方法在速度上的比較。

背景模型	速度(FPS)
紅外線打光器之 GMM [2]	9.62
Histogram N = 32	407.95



## 第五章 結論與未來展望

本篇論文提出新穎的背景模型演算法，此演算法使用了無母數的直方圖，來逼近 GMM 的機率分布。此演算法會產生直方圖的背景模型，這樣的模型比 GMM 還更能全面性紀錄強度與機率分布情況。如同前面所述在 GMM 上有其計算上的限制性，與之相比的是直方圖背景模型會得到全面性的紀錄分布，因此背景景物的變化可以被記錄下來，而免於被誤判的情況。

此外，本論文所提出的模型計算複雜度比 GMM 來得低。以目前評估的最佳效能下，本論文所提出的模型對於一個像素的計算複雜度為 $\theta(M)$ ，已是多模式與無母數的背景模型方法中最低的計算複雜度。如同我們的實驗結果所述，本論文所提出的模型在處理 1920×1080 的 full-HD 影像在一般的電腦上可以達到約 18.81 的 fps。除此之外，如同 GMM 與本實驗的比較，前景切割的結果相似，但本實驗的處理速度比 GMM 快了 3.9 倍。



## 第六章 參考文獻

- [1] Stauffer, Chris, and W. Eric L. Grimson, "Adaptive background mixture models for real-time tracking," *IEEE Computer Vision and Pattern Recognition*, vol. 2, no. , pp., 1999.
- [2] H.H. Lin, J.H. Chuang, and T.L. Liu, "Regularized background adaptation: A novel learning rate control scheme for gaussian mixture modeling," *IEEE Trans. Image Process*, vol. 20, no. 3, pp. 822–836, 2011.
- [3] D.S. Lee, "Effective gaussian mixture learning for video background subtraction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 5, pp.827–832, 2005.
- [4] Z. Zivkovic and F. van der Heijden, "Efficient adaptive density estimation per image pixel for the task of background subtraction," *Patten Recognition Letters*, vol. 27, no. 7, pp. 773–780, 2006.
- [5] Z. Zivkovic, "Improved adaptive gaussian mixture model for background subtraction," in *Proc. Int'l Conf. Pattern Recognition*, vol. 2, pp.28– 31, 2004.
- [6] P. Gorur and B. Amrutur, "Speeded up gaussian mixture model algorithm for background subtraction," in *Proc. IEEE Conf. Advanced Video and Signal-Based Surveillance*, pp. 386–391, 2011.
- [7] H. J. Chang, H. Jeong, and J. Y. Choi, "Active attentional sampling for speed-up of background subtraction," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 2088–2095, 2012.
- [8] L. Li, W. Huang, I. Y.-H. Gu, and Q. Tian, "Statistical modeling of complex backgrounds for foreground object detection," *IEEE Trans. Image Process.*, vol. 13, no. 11, pp. 1459–1472, 2004.
- [9] S. Brutzer, B. H öferlin, and G. Heidemann, "Evaluation of background

- subtraction techniques for video surveillance,” in *Proc. IEEE Conf. Computer Vision and Pattern Recognition. IEEE*, pp. 1937–1944, 2011.
- [10] Y. A. Sheikh and M. Shah, “Bayesian modeling of dynamic scenes for object detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 11, pp. 1778–1792, 2005.
- [11] Change Detection Workshop. (2012). Change detection benchmark web site [Online]. Available: <http://wordpress-jodoin.dmi.usherb.ca/>
- [12] Y.C. Chang and J.H. Chuang, “Vision-based Detection for Railway Level Crossing Violations,” MS Thesis, National Chiao Tung Univ., 2013.
- [13] Y.C. Lu and J.H. Chuang, “Video Summarization for Multi-intensity Illuminated Infrared Videos,” MS Thesis, National Chiao Tung Univ., 2013.
- [14] KaewTraKulPong, Pakorn, and Richard Bowden, "An improved adaptive background mixture model for real-time tracking with shadow detection." *Video-Based Surveillance Systems*. pp. 135-144, 2002.
- [15] D. H. Parks and S. Fels, “Evaluation of background subtraction algorithms with post-processing,” *IEEE Conf. Advanced Video and Signal-Based Surveillance*, pp. 192–199, 2008.