

第一章 簡介

1.1 研究背景

行人計數統計主要應用於兩種環境，一種是應用於封閉區域內，一種是應用於開放空間中。封閉區域指的是有固定範圍、特定出入口，供行人進出的場所，例如：大樓、列車車廂、園區、博物館、商場等，計數器位置通常就設置在出入口的位置附近；開放區域則沒有固定出入口，如騎樓、行人穿越道、廣場等，行人行進的方向不受場景的限制，所以通常是透過計算特定位置的經過行人數，以了解該地點的人數統計。

在行人計數應用相當廣泛，以大眾運輸系統上的人數計數而言，可藉此判斷運輸載具的搭乘人數是否超過其運量，以便進行班次、車輛調配或是路線規劃，予以作即時或是更為精確的調整，以達到提升總體運量以及紓解人潮的目的。行人計數更在許多文獻中被指出其對於一些交通應用的重要性[鄭 84][陳 00]。

對於政府公共部門而言，行人計數可以提供精確衡量公共設施使用效率的功能，以供都市計畫的評估及參考。例如：在人行通道上計數計算行人通過數量，可應用在計算人行道、人行地下道、人行天橋的使用率，有助於相關單了解公共通道使用率，進而判斷其運量是否足夠。另外，若能同時監控多個出入地點，則可以統計出人員進出，以便進行行人通道負載量的預測。

在安全應用方面，在相關文獻[陳 03][沈 93]中提到，當大樓發生火警時，大樓內的人數往往影響著災難時疏散效率及方向，大樓內人員密度若是越高，則疏

散效率越低。因此，倘若能在大樓各出入口做行人計數，則可得知即時建築物內的人數，做適當的控管，對於消防安全，有相當重要的助益。以 2001 年在以色列耶路撒冷南部發生樓房倒塌事件為例，事發當時有 700 多人，聚集在一棟三層建築物內舉行婚禮，因為人數超過建築物的負載，導致建築物倒塌，造成 100 餘人傷亡的慘劇。如果利用人數計數系統，即時得知建築物內的人數，統計場內的人數是否超過其負荷量，進而判斷是否進行管制措施，有效控管公共場所的人數，則可避免災難的發生。

在商業用途應用方面，行人計數系統又可稱為客流量計數器。顧客流量對於依賴於以商業為主的企業來說意義重大，因為顧客流量與銷售額之間存在著高度正相關的關係，使得顧客流量已經成為商業企業投資決策、發展前景和經營效果評估的重要依據。例如：在商場設立據點前，先針對該地點過往的行人進行人數計數，其結果將有助於決定該據點是否設立。通過這一準確的量化的數據，不但可以獲得商場、購物中心、博物館或者飛機場完整的正在運行的狀況，甚至可以由此資訊規劃商店的最佳營業時間，而利用這些高精度的數據，進行有效的組織運營工作。

綜上所知，無論是在交通、公共、安全、或者商業上，行人計數系統確實為人們生活中所必須。事實上，在我們日常生活中也存在著各種形式的行人計數器，也由於其的高重要性，所以目前亟需發展出精確又能廣泛應用的行人計數系統。

1.2 研究目的

目前行人計數最常使用的方法，大致上可以分為三種，分別是人工計數、紅外線感應計數、閘門計數等。

第一種人工計數的方法，是在出入口處以人工的方式計算通過人數。此種方式不需要艱深的技術，就可以得到人數計數的結果。但是此種方法除了成本較高外，還有人員無法長時間工作的問題，因此無法用於長期或通過人數較多的計數應用上；另外，在列車、巴士等移動中的交通工具，也很難去安排人員在各個出入口予以計數。

第二種以紅外線感應的方法，主要是於進出入口處架設紅外線感應器，利用行人經過時，將紅外線遮斷的特性，得知是否有行人通過，也是目前較常見的自動計數系統。此方式雖然成本低廉，也不需要艱深的技術，而且沒有人員疲勞以及人工成本昂貴的問題，但是當行人並排進入時，容易發生誤差。因此對於尖峰時間所湧入的大量行人，無法精確的進行計數。圖 1.1 為設置於錢櫃 KTV 門口的人數統計系統，利用行人通過時，將紅外線遮斷的特性，得知是否有行人通過。架設於 SOGO 百貨公司忠孝復興店的行人計數感測器(圖 1.2)，雖然其使用兩個感測器，並架設於行人通道上方(圖 1.3)，減少了行人並排的誤差，但是當一個行人同時被兩個感測器偵測到時，或者行走於非感測器正下方時，計數都會發生誤差。



圖 1.1、錢櫃 KTV 人數統計系統 (a)紅外線感應器 (b)紅外線反射板



圖 1.2、SOGO 行人計數系統(a)行人計數看板 (b)兩個感測裝置



圖 1.3、SOGO 行人計數系統架設位置圖

第三種在出入口的地方設置閘門，利用行人通過閘門時，以推動橫桿的方

式，計算通過閘門的行人數量。此種計數方式，不需要艱深的技術以及大量的人力，即可精確得知各閘門通過的行人數量。但是閘門的設置會影響行人通行，降低用路人的行走速率，且人潮過於龐大時，則容易造成出入口壅塞，影響行人通行。

近年來開始有人嘗試使用數位攝影機拍攝行人通道，利用數位影像處理技術進行行人計數的工作。這種方式有下列幾項優點：第一、經適當的架設位置及角度後，不像紅外線遮斷計數器，容易受到遮蔽(occlusion)的影響。第二、不像閘門，會影響行人通行。第三、設置方法容易且成本合理。第四、蒐集豐富的資訊，可供行人計數以外的應用。因此本研究希望建構出以電腦視覺為基礎的行人計數系統，達到自動化雙向行人計數的功能，供各類型的行人管理及分析之用。

1.3 文獻探討

為了得到通道上行人的資訊，許多研究將攝影機架設於行人通過的路線旁 [Ket99]，拍攝得到行人的影像，再擷取出影像中的行人區塊進行統計。然而此種拍攝方式容易因行人區塊重疊(occlusion)，而發生計數錯誤的情況，因此在 [Yan03]的研究中，利用多支攝影機，架設在室內的環境中，監控室內的行人；再藉由每支攝影機相對的位置，來得知行人的個數以及行進的路徑，可以有效的解決行人重疊的問題。然而需架設多部攝影機，成本較高，且因資料量大、處理速度較慢，導致此方法在實際上應用較不方便。

當攝影機裝置於通道上方以俯視(top-view)監測時[Bes03]，可以大幅減少行人重疊的問題，但是由於受到場景及應用的影響，仍有攝影機架設空間的限制及通道高度不夠等問題，例如：在[Ter99]需將攝影機架設於離地面 410 公分之處，使場景能完全包含在監控範圍內，然而一般室內行人通道並不容許這樣的攝影機架設高度需求。在[Wan07]中，提出了使用了魚眼攝影機(fisheye camera)利用其廣角特性解決架設空間不足問題，雖然在廣角攝影機中取得的影像會有所扭曲變形，但是對於行人計數影響並不大。本研究即是根據[Wan07]系統架設方式，使用俯視方式對場景進行監控，並且提出通用於一般攝影機及魚眼攝影機的行人計數系統架構，以提高系統的應用範圍。

一個行人計數系統架構通常包含三個步驟：行人偵測(detecting)、追蹤(tracking)、及計數(counting)。在行人偵測階段，主要工作為偵測前景，而一般常見前景擷取的方法有：model matching [Bro00][Zha97][Xia95][Xia97], temporal differencing [Dai00], and background subtraction[Che03][Mas01][Zha95][Jae]。Background subtraction 是事先建立好該場景的背景影像[Lai98]，然後將目前輸入的影像，和背景影像比較以得到行人區塊；Temporal differencing 利用前後兩張 frames 的時間差，找出影像中有移動的物體；而 model matching 的方法，則是事先以形狀[Har99]或顏色[Dar98]建立好行人模組，接著在影像中尋找相似的部分，取出行人區塊。由於光線變化會影響行人區塊的擷取，因此，此方法主要應用於室內等較不易受到光線影響的環境中。在前景偵測階段所擷取出來的前景物

建和物體實際的形狀跟位置通常有所誤差，而在使用一般前景擷取的方法時，最容易影響擷取成果的因素主要為光影的影響以及行人的重疊，尤其在光線較強的場景，如：戶外，行人的陰影很容易跟前景一起被擷取出來，因此不少研究中提出了移除陰影的方式，在[Bes03]中，使用 DTC(discrete cosine transform)的方式，將亮度改變但材質沒改變的區塊視為陰影。在[Che05]中，利用 joint histogram 的分析，將每個像素作分類(classification)，藉此找出陰影部份，但是對於不同場景的應用需要重新分析。[Ter99]使用立體影像(stereo image)的方式，來獲得行人的位置，不過需要使用兩台攝影機來建立出立體影像，且無法進行即時計數。

行人區塊可能因為重疊的原因彼此相連結，導致一個區塊包含好幾個人，因此，[Che06]使用區塊面積的方式，判斷該區塊有多少行人，但是這樣的方式過於粗略，且對於不同的場景必須重新定義面積與人數之間的關係。[Har99]將側面監控的前景區塊作垂直統計圖分析，藉由尋找峰值個數來決定行人各數，但是此方法不適用於以 Top-View 拍攝的影片中。[Alb01]則以 morphological method 來將重疊的區塊拆開，但是其方法須考慮到時間的因素，導致不適用於即時系統。Level set Method[Jam99]被提出可以取得精確的人物區塊，在重疊遮蔽情況不嚴重的情況下，可將行人彼此之間分開來，也有一定的去除陰影效果，不過一般認為 Level set Method 的重複運算造成計算速度較慢，所以可實際應用度不高。還有一種可以判斷影像中區塊中人數的方法，是利用類神經網路(neural network)的方式來判斷影像中是否有行人、其狀態以及行進方向以及區塊中大約

人數[Hua02][Sch95]。首先，以類神經網路學習(training)各種人數狀況行人區塊的特性，待學習完成後，便可以目前影像，分類出不同人數的區塊，以計算影像中的行人個數。然而，此種方式，必須事先將可能遇到的情形加以分類，再花費時間讓類神經網路去學習分類的結果，最後才可到實際的環境中應用，因此在使用上比較耗時；而且若是換到其他的場景，或出現尚未分類的狀況，則可能造成錯誤的發生。

在行人計數的應用中，通常只需針對行人進行計數，非行人的物體通常會忽略計數。有關判定追蹤的物體是否為行人或者其他物體的相關研究中，大多數採用行人樣版的方式來進行人的比對。常見的方式為使用矩形或橢圓來表示出人物的形體(shape)[Mag07]，但是這樣的表示方式太過粗略，無法完全對應出行人的外型，特別是當行人之間彼此重疊的造成形狀改變的時候。而行走中的人類並非固定不變的形體，無法像在車輛辨識[Tan07]時，使用有限個特定的樣版。[Hsi07]提出一系統架設，可以學習場景中行人行走的方式，建立以格線為基礎(grid-based)的行人樣版。這樣的方式在不同應用場景中，需要重新建立行人樣版，加上學習時所擷取的行人區塊如不夠完整或帶有雜訊，將會影響統計人數的精確度。

在行人追蹤及行進方向判定上，為了得知行人的進行方向，[Alb01]利用兩條不同的觀測列，分別建立兩張 epipolar-plane images(EPIs)，在其上以 optical-flow 的方式計算行人速度，但是這種方法必須花費時間建立 EPIs，對於一些需要即時處理的應用並不適用。[Ter99]使用兩條掃描線建構出兩張時空影像(space-time

image)再將兩張影像作 template-matching 來判斷行人的移動方向，不過建立時空影像造成系統無法即時計數。近年來一些以往被用來作物體追蹤及預測的方法，如：卡曼濾波器(Kalman filter)[Mas01]及 HMM(Hidden Markov Model)，被應用到行人的追蹤當中，以找出行人的位置與運動方向。不過在[Isa96]中提到卡曼濾波器主要基於高斯分佈預測物體穩定的移動，對於像行人其速度跟方向可能隨時改變而不連續的情況，是很容易出現偵測錯誤的。而粒子濾波器與同樣熱門的 mean shift 及卡曼濾波器相比，卡曼濾波器有執行速度快優勢，mean shift 則具追蹤過程平順不抖動的優點，粒子濾波器對於移動速度快、運動方向變化大或物體有較大形變時，能展現出其強健性(robust)，達到準確的追蹤結果。

粒子濾波器(particle filter)[Isa96][Cho06]是一種 sequential Monte Carlo(MC)方法，而 sequential Monte Carlo 方法根據不同作者跟應用方式有些許差異，名稱也有不同形式的別稱，如：sequential importance sampling(SIS)、particle filter、bootstrap filter、the condensation algorithm、interacting particle approximations、及 survival of the fittest。粒子濾波器提供了一個非參數化(non-parametric)的模型，不像卡曼濾波器及 HMM，Particle filter 以模擬粒子隨機運動的方式，利用事前機率(prior)、事後機率(posterior)，預測物體可能的位置。在粒子濾波器應用中，前景提供了證據(evidence)，[張 05]使用前景的大小及位置來定義與實際物體的相似程度(likelihood)，而[鄭 05]則使用形狀、材質、及 color histogram 來定義之。

綜上所述，各種方法皆有其優缺點。在偵測階段，本研究則提出採用 temporal

differencing 找出移動物出現的區塊，並結合物件邊緣資訊，利用 Level set Method 收縮物件邊界將行人和場景分割出來，以取得較完整且較不受光影影響之行人區塊，以避免單純使用 temporal differencing 容易將移動物與部分背景一起擷取出來的缺點。本研究將嘗試以粒子濾波器追蹤行人區塊，結合區塊相對位置及大小形狀等定義追蹤目標相似程度。基於上述粒子濾波器特性，對於容易受到不同攝影機特性及架設環境影響的前景區塊，無論其產生的各種類型的區塊粒子濾波器皆能準確追蹤。最後利用粒子濾波器準確的追蹤結果進行行人進出場景的統計，以達到行人計數功能之系統。

1.4 論文架構

在第二章中，我們將對整個行人計數系統作一個描述，在這個系統中有兩個子系統，分別為：行人偵測系統、行人計數系統；我們將分別在第三章介紹行人偵測系統，第四章介紹行人計數中的追蹤及計數統計，第五章中展示實驗成果，第六章則為結論，並討論未來可能研究方向及工作。

第二章 系統架構

本章將簡介系統的整體架構，分為兩部分，第一部份介紹環境設定及系統設置，而第二部份則說明系統的運作流程。

2.1 系統設置

一般而言，以視覺為基礎的數位監控系統，會因為攝影機架設位置不同，而有不同的特點。比如將攝影機架設在行人通道的兩側，可以獲得行人的完整輪廓，在判斷行人行為的應用中，可提供較多資訊，但行人重疊的問題會較嚴重，不利於判斷人數。若將攝影機架設在行人通道的正上方(如天花板或門口上方)，以俯視(top-view)的方式，監測行人通道，則可以避免行人區塊重疊的問題，雖然無法用於判斷行人動作及行為，但卻可以有效進行行人計數。

本系統主要目的為行人計數，所以我們選用俯視方式為攝影機架設方式，取其可以避免行人區塊重疊的優點。然而在有些場景(如：室內場景)之中，因為沒有足夠高度，讓一般攝影機視角包含所有需監測的範圍，因此我們選用魚眼鏡頭的環場攝影機，取其超廣角的特性，以便增加可監控的範圍。本研究將魚眼攝影機架設於行人通道上，所拍攝畫面如圖 2.1 所示。

雖然架設將魚眼攝影機架設在行人通道上，可以減少影像中行人重疊的問題，但魚眼攝影機的影像特性仍然造成系統開發上的困難。第一種特性是在魚眼攝影機所拍攝到的影像中，當物體接近監控區域中心時，成像會較大，導致所觀

測到行人外型與周圍偵測到的明顯不同；再加上行人自身形狀的改變，會使得追蹤比對時產生困難。第二種特性是當行人位於監測區域較外圍時，所觀測到的影像會近似於側視影像(side-view)，在這種情況下，行人的成像較易產生重疊的現象，造成一開始計數行人的困難。因此，為了解決上述問題，針對第一種特性，我們在行人追蹤時，容許物體的大小形狀具有一定程度的變化，而第二種特性，我們則透過定義追蹤範圍，使得該範圍在接近監控區域中心來解決。

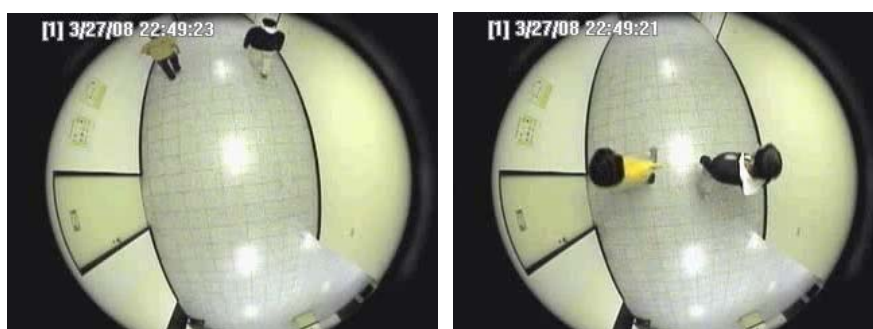


圖 2.1、魚眼攝影機取像結果

為了判斷人員的進出，我們在攝影機所監視到的影像中定義兩條基線，分別為內部基線(inside base-line)和外部基線(outside base-line)，如圖 2.2 所示。此兩條基線會將影像分為三個區域，分別稱為內部區域(inside area)、追蹤區域(tracking area)、外部區域(outside area)。內部區域指的位於出入口裡面的空間(如：房間、演講廳、建築物的內部)，換句話說就是我們需要統計人數的空間。追蹤區域指的是用來進行追蹤的通道空間，位於兩基線的中間。一般而言，我們將追蹤區域定義在接近鏡頭正下方的位置或監控區域中心的位置，因為在這些區域內影像受

到鏡頭扭曲的情況較輕微，其行人成像也比較不容易重疊，可以增加系統追蹤的準確度。外部區域指的是在出入口外面的空間(如：大廳、走廊、建築物外部)，這個區域會通往開放式的空間，所以行人數量是無法統計的。當行人從外部區域經過偵測區域進入到內部區域時，我們定義行人為進入，反之定義為出。

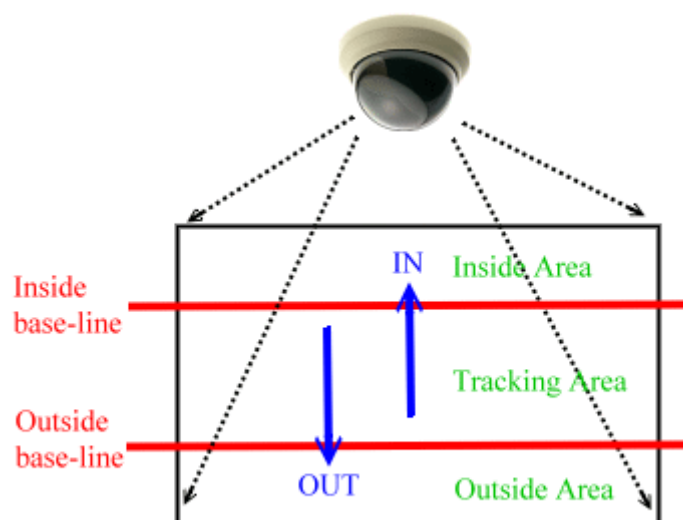


圖 2.2、攝影機與場景關係示意圖

2.2 系統運作

整個俯視型行人計數系統架構流程圖如下圖 2.3 所示，主要分成分別為行人偵測 (pedestrian detection)及行人計數(pedestrian counting) 兩個部分。

在行人偵測部份，主要是找出行人所在及其大小形狀。在影像中的人物分成兩類，一種是正在行走的行人，另一種是暫時停止不前進的行人。我們先從攝影機所得到的連續兩張影像比對後，找出影像中的變動區域，找出正在行走的行人可能的位置。在這之前所偵測到的行人區塊，則是暫時停滯的行人(temporal

stopping object)可能出現的地方，我們將這些區塊與變動區塊合併，稱為注意區域(attention region)，亦即行人可能出現的地方。接著我們在注意區域中找到行人區塊的邊緣(edge)，以擷取出行人區塊。行人區塊的邊緣主要是利用前處理(preprocessing)所得到的邊緣資訊，與背景邊緣影像比對後得到。最後，我們就得到了行人的前景區塊(foreground object)。

在行人計數部份，又分為行人追蹤(tracking)、統計計數(counting)等步驟。在行人追蹤中，主要是先將前面所得到的行人區塊轉換成狀態表示，包含行人位置、大小、形狀、及顏色等資訊。前後兩張影像中的行人區塊，藉由其狀態，進行比對，以得到行人的移動路徑。在區塊比對時，常常會發現行人區塊發生碎裂或者是形狀改變的問題，所以我們用粒子濾波器(particle filter)特徵進行追蹤，取其在較多雜訊或者觀測資訊不完整的狀況下仍能保持良好追蹤效果的優點。在統計計數方面，則是根據所追蹤的行人行進路徑跨越影像中偵測線的方式，將行人分成各種進出狀態，決定行人的進出方向，當行人穿越內部基線而進入到內部區域則為進(in)，穿越外部基線而進入到外部區域則為出(out)，以得到行人的進出數量。

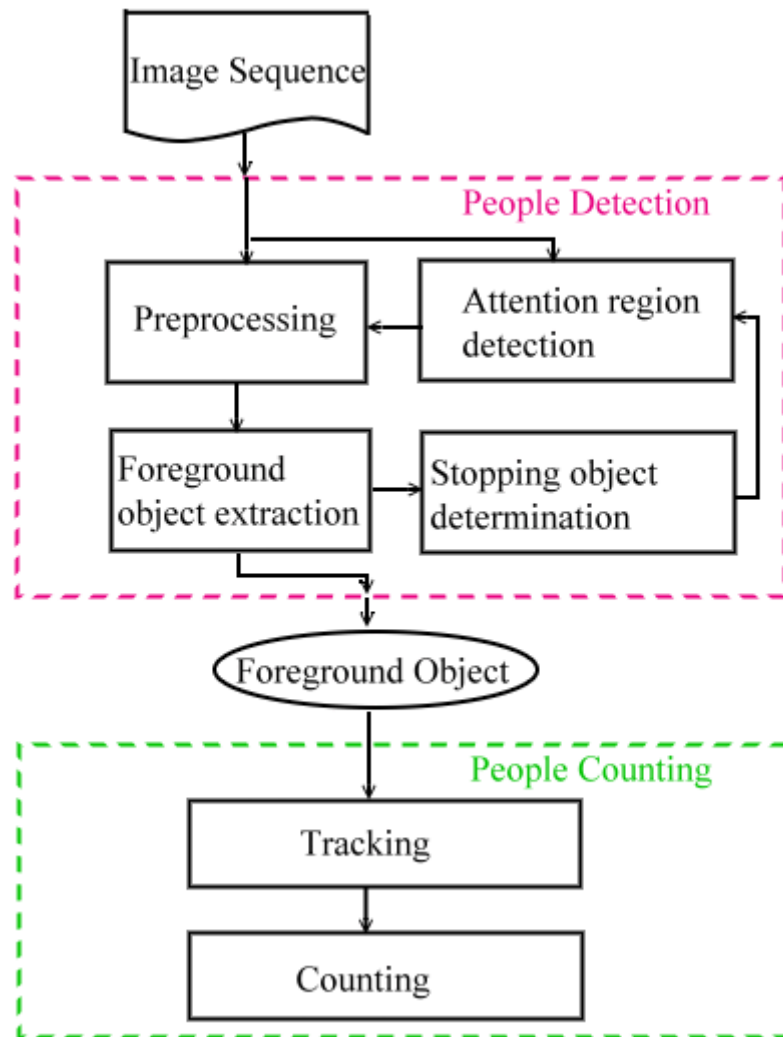


圖 2.3、系統架構圖

第三章 行人偵測

行人偵測包含四項主要工作：第一是前處理、第二是注意力區域偵測、第三為前景物擷取、第四則為暫時停滯物決定，其關係如圖 3.1 所示。

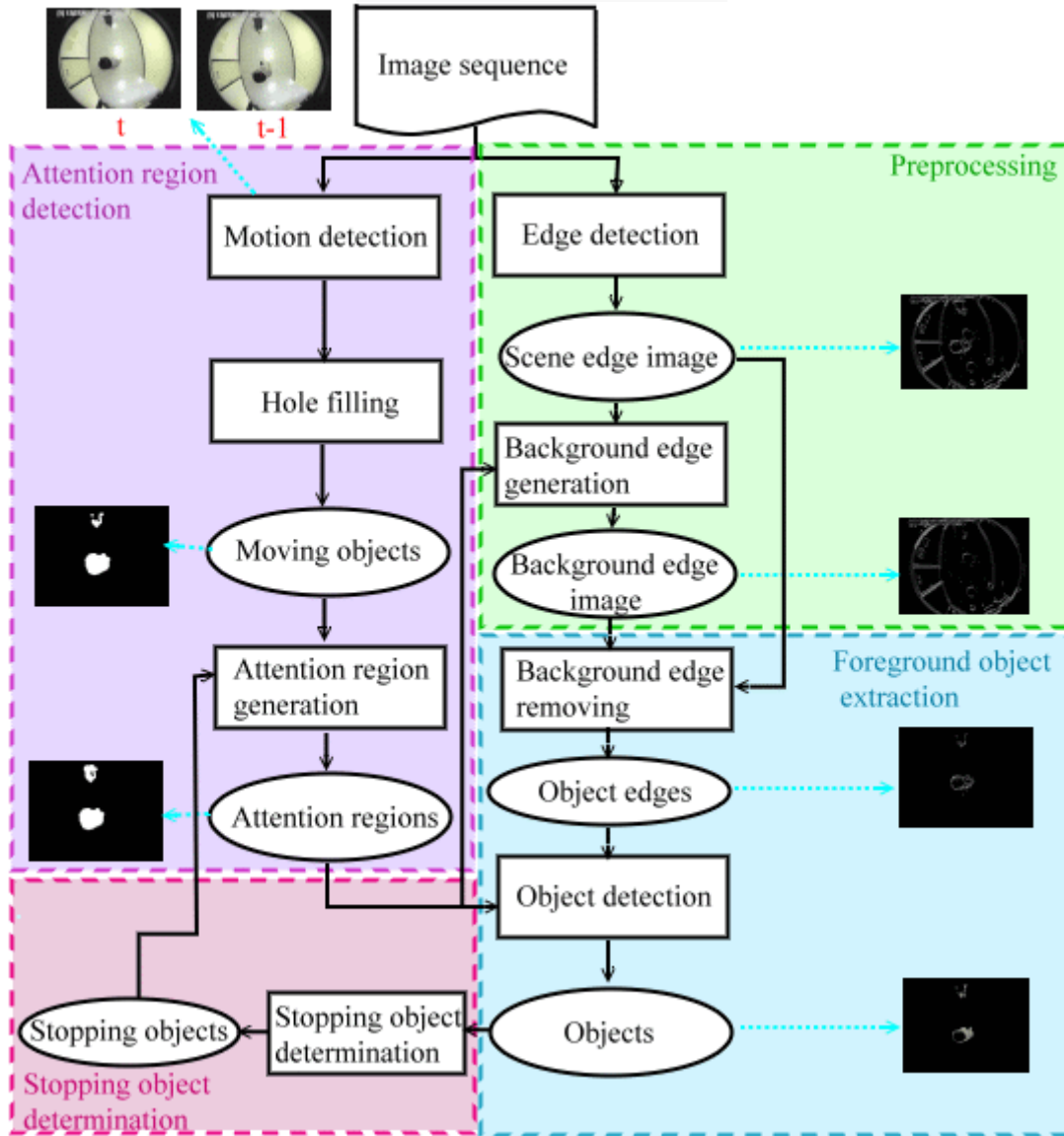


圖 3.1、行人偵測流程圖

在前處理中，我們將對攝影機所擷取到的影像作邊緣偵測(edge detection)，以建立出的邊緣影像(edge image)。再使用注意力區塊偵測中所得到的注意力區

域，忽略前景區塊以建立背景邊緣(background edge)出來。在前景物的擷取中，便是使在注意力區域中，經由背景邊緣分析後，找出前景邊緣(foreground edge)，進行得到完整的行人前景區塊。所擷取出的前景物資訊則會被儲存起來，當其暫時停滯時，與下個時間的變動區塊合併，以成為注意力區域。

本論文在 3.1 節將介紹前處理工作詳細步驟，在 3.2 節中描述尋找注意力區域的做法，3.3 節則說明前景物擷取，最後於 3.4 節介紹暫時停滯物決定與儲存方法。

3.1 前處理

前處理主要包含兩項工作，第一是將影像中的邊緣偵測出來，主要因為行人與場景之間通常會形成強烈的邊緣，透過邊緣偵測的技術，可以得到這些邊緣。第二項工作為建立背景邊緣影像(background edge image)，主要是為了幫助我們在後面的步驟中區分出物件邊緣或者背景邊緣。以下將分別在 3.1.1 介紹我們所使用的 Canny 邊緣偵測法，以及在 3.1.2 介紹如何建立背景邊緣影像。

3.1.1 Canny 邊緣偵測

在本研究中，為了避免一般邊緣(如：Sobel)偵測法容易產生的邊緣輪廓不完整、邊緣有斷口、有較弱邊緣(weak edge)及雜訊影響的缺點，採用一種稱為 Canny 邊緣偵測的最佳化邊緣偵測演算法，將行人的輪廓完整定位出來。Canny 邊緣偵

測的包含四個步驟：

Step1：利用 Gaussian 濾波器將影像平滑(smooth)去雜訊。

Step2：計算影像梯度(gradient)的強度(magnitude)與方向(orientation)。

Step3：使用 nonmaxima suppression，將強度非最大值的邊緣抑制。

Step4：利用雙門檻值演算法，偵測並且連接邊緣線。

各步驟說明詳細說明如下：

(1)高斯平滑(Gaussian smooth)法去雜訊

先將輸入的原始影格用高斯平滑法來做迴旋積(convolution)對影像達到平滑效果。假設 $I(x, y)$ 代表影像，而 $G[x, y, \sigma]$ 是一個高斯濾波器，而 σ 是用來控制平滑程度。較大的 σ 可以獲得較好的雜訊濾波，但是容易喪失重要的邊緣資訊，或造成兩邊緣線接在一起，通常 σ 的大小根據物件的尺寸與位置而被選擇。所以我們選用的是 5X5 且 $\sigma=2$ 的高斯濾波器遮罩。影像 $I(x, y)$ 使用 $G[x, y, \sigma]$ 作迴旋後的結果平滑影像應為：

$$S(x, y) = G[x, y; \sigma] * I(x, y) \quad (3.1)$$

(2)計算影像中的梯度強度及梯度方向

我們使用有限差分近似法(finite-difference approximation method)計算影像梯度，將被平滑後的影像 $S(x, y)$ 中可能是邊緣的區域尋找出來。首先我們先分別取得影像中的水平跟垂直方向的邊緣強度：

$$P(i, j) \approx (S(i+1, j) - S(i, j) + S(i+1, j+1) - S(i, j+1)) / 2 \quad (3.2)$$

$$Q(i, j) \approx (S(i, j+1) - S(i, j) + S(i+1, j+1) - S(i+1, j))/2 \quad , \quad (3.3)$$

再將兩個結果合併以得到完整的影像梯度：

$$M(i, j) = \sqrt{P(i, j)^2 + Q(i, j)^2} \quad , \quad (3.4)$$

角度的計算方式為：

$$\theta(i, j) = \tan^{-1}\left(\frac{Q(i, j)}{P(i, j)}\right) \quad , \quad (3.5)$$

再將 $\theta(i, j)$ 依照圖 3.1 分成四種分段： 0° (水平方向)、 45° (向著正相的對角方向)、 90° (垂直方向)、 135° (向著負相的對角方向)。任何角度介於 $0^\circ \sim 22.5^\circ$ 或者 $157.5^\circ \sim 180^\circ$ 將會被分配到 0 的區段，角度落入 $22.5^\circ \sim 67.5^\circ$ 被分配到區段 1，介於 $67.5^\circ \sim 112.5^\circ$ 則被分配到區段 2， $112.5^\circ \sim 157.5^\circ$ 分配給區段 3，如圖 3.2。

以數學式表示為：

$$\xi(i, j) = \text{Sector}(\theta(i, j)) \quad . \quad (3.6)$$

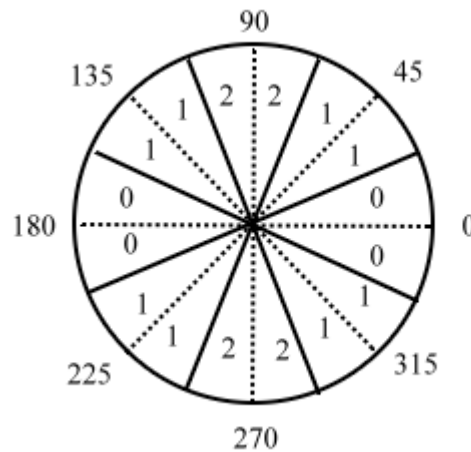


圖 3.2、角度分段示意圖

(3)使用 nonmaxima suppression

一般邊緣偵測法使用訂定門檻值的原則，為將梯度強度位於高處的像素留下來作為邊緣，Canny 則使用更精確方法 nonmaxima suppression，將位於區域最高值(local maxima)的像素留下，並讓其餘像素去除，使得過粗(thick)的邊緣變細(thin)，而所保留下來的邊緣寬度為一個像素寬。作法是檢查 $M(i, j)$ ，若 $M(i, j)$ 比其沿著梯度方向區段的鄰近點都較小，則設 $M(i, j)$ 為 0。在經過 nonmaxima suppression 程序後，所產生的影像以數學式表示為：

$$N(i, j) = nms(M(i, j), \xi(i, j)) \quad (3.7)$$

(4)利用雙門檻值演算法連接邊緣線

最後確保不必要的雜訊邊緣不會干擾到邊緣偵測的結果，所以 Canny 使用兩個門檻值做為過濾的標準。之後開始沿著這些區域追蹤每一個像素的強度，並在追蹤過程中依據兩個門檻值做為過濾，如果此像素強度低於第一個 T_L 門檻值的時候，便將之設為 0，即判定此像素不算邊緣，如果強度大於第二個 T_H 門檻值時，該像素即為我們要找的邊，若強度介於 T_L 和 T_H 之間時，除非與此像素相鄰的像素有任一個大於或等於 T_H ，否則將此像素設為 0。

通常前景物件和背景物件之間會產生較清晰的邊界，然而當物件顏色與背景相似的時候，邊界將會不明顯。也就是說，為了要抓取出完整的前景邊緣，我們必須要保留一些較弱的邊緣。實際作法上，我們在偵測邊緣時，使用了較小的強度門檻值。到此為止，我們完成了 Canny 邊緣偵測的所有步驟，並取得了邊緣影像，令時間 t 的邊緣影像為 E^t (如圖 3.3)。

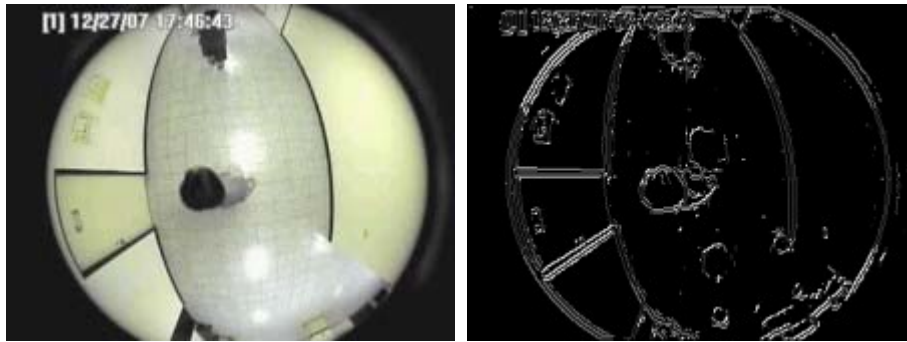


圖 3.3、邊緣偵測影像(a)原始影像(b)邊緣影像 E^t

3.1.2 背景邊緣影像的建立

為了要把影像中的行人抓出來，我們必須找出哪些邊緣是屬於行人，哪些是屬於背景部份。欲取得行人的邊緣，則必須先建立出背景邊緣，而背景邊緣則可以藉由前一個時間的邊緣影像及注意力區域建構出來。

所謂的注意力區域(attention region)就是在影像中行人可能出現的區域，再這樣的區域我們將會提高注意力來尋找影像中的行人，如此一來可以減少搜尋行人範圍，更可提高擷取出的前景之準確度。符合這樣定義的區域包含兩種，第一種是影像中的變動區域，是指行人經過畫面時所造成的影像變動；第二種是先前已被偵測出的前景區塊，主要是指暫時停滯行人區塊（注意力區域尋找方式詳見 3.2 節）。

由於行人包含在注意力區域中，所以在非注意力區域的邊緣皆可視為背景邊緣，而在注意力區域的背景邊緣則可以用之前的背景邊緣來填補。綜合上述，可以利用前一個時間的邊緣影像 E^{t-1} 及注意力區域 A^t 建構出背景邊緣 B^t ：

$$B^t(x, y) = \begin{cases} B^{t-1}(x, y) & \text{if } (x, y) \in A^t \\ E^{t-1}(x, y) & \text{otherwise} \end{cases}, \quad (3.8)$$

而在系統初始 $t=0$ 時候的，則使用 $t=0$ 的邊緣影像當作背景邊緣即可。背景邊緣結果，如圖 3.4。

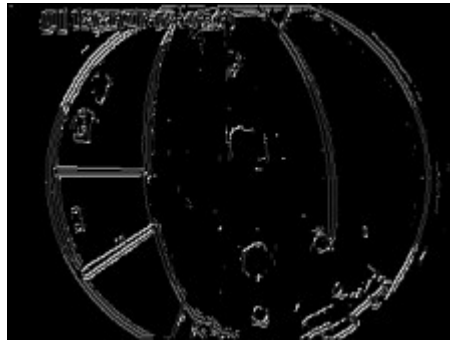


圖 3.4、背景邊緣影像

和一般背景影像不同的，我們的背景邊緣影像有下列幾項特點：

- (1) 影像邊緣比較不會受到環境中光影變化影響，背景邊緣建置可以不需將光影變化因素考慮在內。
- (2) 背景邊緣與上一個影格的背景邊緣影像有高度關聯性，所以背景邊緣可以參考之前的背景邊緣影像來建置。
- (3) 背景邊緣可以立即建置，不需要訓練時間(training time)。

到此為止，我們得到了背景邊緣 B^t 。

3.2 注意力區域偵測

注意力區域偵測方面，本研究使用連續影像變異(image difference)方法，比

較兩張連續影像 I^t 及 I^{t-1} (如圖 3.5)，以找出變動的像素，並令這些有變動像素集合為 D^t ，

$$D^t = \left\{ (x, y) \mid \left| I^t(x, y) - I^{t-1}(x, y) \times \alpha \right| \geq th \right\} \quad , \quad (3.9)$$

其中 th 為一根據影像品質而事先定義的門檻值， α 為增益率(gain rate)，主要是用於修正影像之間的亮度變化(計算方式說明於後)。 D^t 顯示出兩張影像發生變化的部分，顯示出移動物所出現的大概位置。

雖然兩張影像的差異時間很短，但可能會因為照度(illumination)快速改變，或者是攝影機的自動光圈變化，導致影像的整體亮度值(即 global intensity)發生變化。為了修正這種差異，我們藉由計算在時間 t 和 $t-1$ 連續兩張連續影像 I^t 及 I^{t-1} 的增益值 α 作補償的動作。 α 值的計算方式如下：首先，將一張影像分割成 k 個區塊(block)(每個區塊為 $n \times n$)，並計算每個區塊的平均亮度值。再將這些平均值標示成 $I_0^t \dots I_k^t$ 和 $I_0^{t-1} \dots I_k^{t-1}$ ，分別代表影像 I^t 及 I^{t-1} 。而 α 值可以藉由最小亮度值變化的區塊 p 計算出來：

$$p = \arg \min \left\{ \left| I_0^t - I_0^{t-1} \right| \dots \left| I_k^t - I_k^{t-1} \right| \right\} \quad , \quad (3.10)$$

$$\alpha = \frac{I_p^t}{I_p^{t-1}} \quad . \quad (3.11)$$

這樣，在計算公式 3.9 時，由於兩張影像的整體亮度獲得了補償，所獲得的影像差異可將光影變化的影響降至更低。

當我們將 D^t 顯示為一張影像時，會發現 D^t 的區域會存在著些許小洞(holes)，使得這些區塊並不完整。為了填補這些小洞，我們對 D^t 影像取互補

(complement)影像得到影像 $\overline{D^t}$ ，其中 $\overline{D^t}$ 可表示成：

$$\overline{D^t} = \left\{ (x, y) \mid \left| I^t(x, y) - I^{t-1}(x, y) \times \alpha \right| < th \right\}, \quad (3.12)$$

再對 $\overline{D^t}$ 作一次連通區域偵測，找出與影像邊界沒有連結的團塊(blobs)，也就是前面所提到的小洞。再將由 $\overline{D^t}$ 對應到 D^t 上把這些孤立(isolates)的團塊移除，到此可以取得完整的變動區域，如圖 3.6。

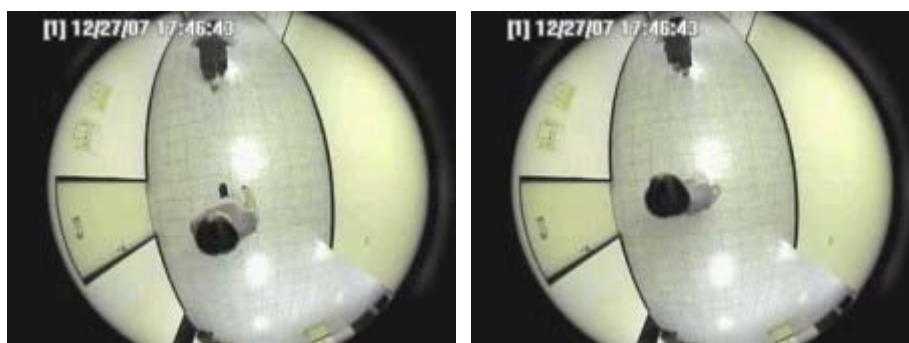


圖 3.5、原始影像(a) $t-1$ 時間的原始影像(b) t 時間的原始影像

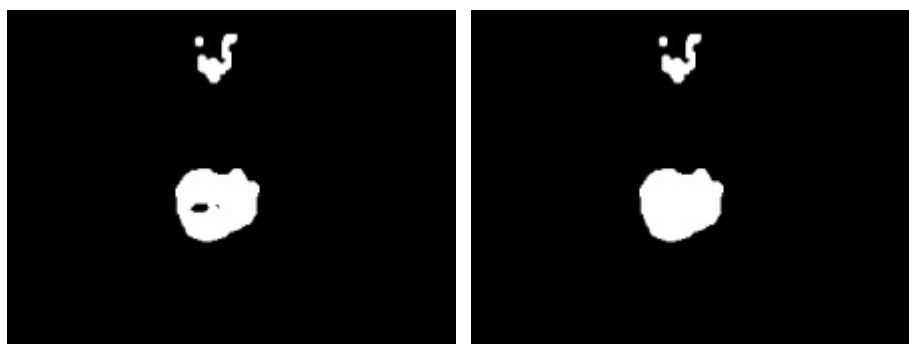


圖 3.6、影像中變動區域(a)有變動區域(b)缺洞填補後的變動區域

最後我們將影像中有移動物出現區域 D^t 及上個時間的暫時停滯物 M_i ,

$i=1\dots m$ 在結合一起成為新的區域 A^t ：

$$A^t = \left\{ (x, y) \mid D^t \cup M_i, i=1\dots m \right\}, \quad (3.13)$$

其中 m 為暫時停滯物的個數。所得到的區域 A^t 即為我們所尋找的注意力區域。

3.3 前景物擷取

產生了注意力區域後，我們就可以利用注意力區域找出行人所在的位置。既然影像中的行人包含於注意力區域，而前景物的邊緣也就是物件(或行人)的邊界，我們可以藉由收縮(shrink)注意力區域的邊界(boundary)至前景物件的邊緣，以找出這些行人。

然而在作收縮程序時，我們會面臨兩大問題：首先，我們沒有事先知識(prior knowledge)區分物件的邊緣以及背景邊緣；再來，可能因為物件與背景的邊界不明顯，所以導致物件的邊緣不全為強的邊緣(strong edge)，使得物件邊緣不完整。為了解決第一個問題，我們在 3.1 節中使用前一個時間的邊緣影像 E^{t-1} 作為事先知識，建立起背景邊緣，以便取出物件的邊緣。而第二個問題中，我們需要在物件邊緣上，對於邊緣較弱，或邊緣不連續的狀況下，對物件的邊界提出猜測，再這邊本研究提出了使用 level set 的方式將物件的邊界擷取出來。我們將在 3.3.1 節中介紹物件邊緣偵測的方法，而在 3.3.2 介紹整個收縮邊界的流程。

3.3.1 物件邊緣偵測

為了求得物件的邊緣，我們將現有的邊緣圖上，藉由比較邊緣的方向(orientation)的方法將屬於背景邊緣的部份移除掉。為了適應目前影像與背景影像

可能的偏移(shift)，邊緣的比較是由下列的距離函數 d 來決定：

$$d(X, Y) = V[(X - Y)] \quad (3.14)$$

其中 X 和 Y 為像素近鄰的邊緣方向分佈， V 為此分佈的變化程度。作法是檢查具有場景邊緣的像素周圍 5×5 的範圍內的背景邊緣及場景邊緣垂直及水平方向的邊緣的差異程度。如果距離小於一個門檻值，則此邊緣像素被歸類於背景邊緣，然後從 E 移除。

在實作中，為了提昇系統效能，且行人不會存在於非注意力區域中，所以物件邊緣只存在於注意力區域中，我們將現有邊緣圖 E 中不在注意力區域的邊緣，從 E 中去除。但是須注意到注意力區域的範圍中，仍有可能有屬於背景的邊緣，所以再將 E 將 B 的部分去除，將所得的邊緣圖表示為 $E_{obj} = (E - B)$ 。所得結果見圖 3.7。

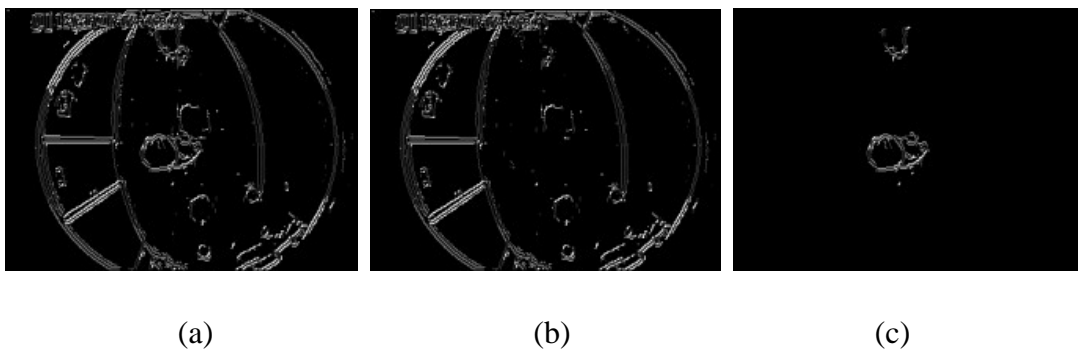


圖 3.7、邊緣影像(a)場景邊緣影像(b)背景邊緣影像(c)前景邊緣影像

3.3.2 收縮邊界

在處理收縮邊界的時候，可能因為物件邊緣不連續或有缺洞，使我們必須對不完整的物件邊緣作收縮，在此我們使用 level set 方法來表示以及收縮邊界，以便利用收縮後的邊界取出前景物件。首先，將物件的邊界表示成一個在影像中的封閉曲線(curve) $\Gamma : [0,1] \rightarrow (x, y)$ 。Level set 方法的主要觀念建立在 level set 函數 ϕ 上，使得：

$$\phi(x, y) \begin{cases} 0 & \text{if } (x, y) \in \Gamma \\ + & \text{if } (x, y) \in \Gamma_{in} \\ - & \text{if } (x, y) \in \Gamma_{out} \end{cases}, \quad (3.15)$$

其中 Γ_{in} 指的是在曲線內的區域， Γ_{out} 為在曲線外的區域。當 $\phi(x, y) = 0$ 時，我們稱為 zero level set，整個演算法就是藉由不斷更新曲線，使得曲線達到 zero level set 的狀態，這時邊界也收縮到物件的邊緣處。

為了說明曲線的更新的過程，這邊使用 Chain rule 來推導 ϕ 的偏微分方程式 (partial differential equation)：

$$\frac{\partial \phi(\Gamma)}{\partial \tau} = \frac{\partial \phi(\Gamma)}{\partial \Gamma} \frac{\partial \Gamma}{\partial \tau} + \frac{\partial \phi}{\partial \tau} = \nabla \phi \cdot \Gamma_{\tau} + \phi_{\tau} = 0 \quad (3.16)$$

其中 τ 為更新間隔時間，而 Γ_{τ} 可視為曲線的傳遞速度(propagation speed)。傳遞方向(propagation direction)則應該指向曲線內部的方向，且應該與 $-\nabla \phi$ 的方向是

相同的。接著我們計算向內的法向量(normal) $-\frac{\nabla \phi}{|\nabla \phi|}$ ，並用法向量及速度方程式

F 取代 Γ_{τ} ，令 $\Gamma_{\tau} = F \cdot -\frac{\nabla \phi}{|\nabla \phi|}$ ，代入(3.16)式中得到：

$$\nabla \phi \cdot \Gamma_{\tau} + \phi_{\tau} = 0 \quad (3.17)$$

$$\rightarrow \nabla \phi \cdot F \cdot \left(-\frac{\nabla \phi}{|\nabla \phi|}\right) + \phi_\tau = 0 \quad (3.18)$$

$$\rightarrow \phi_\tau = F \cdot |\nabla \phi| \quad \circ \quad (3.19)$$

我們可以根據物件的邊界來假設速度方程式，我們將 F 定義為：

$$F = F_{ext} + F_{int} + F_{img} \quad , \quad (3.20)$$

使得 Γ 滿足下列定義：

(1) 外部力(external force) F_{ext} ：由於物件一定位於注意力區域內， F_{ext} 被定義為一個用來將曲線往內縮的常數。通常，它是一個很小的數，在物件邊界上時，更可以將其設為零。

(2) 內部力(internal force) F_{int} ： F_{int} 可以經由曲線的曲度(curvature)來計算，令 $F_{int} = \gamma |\Gamma''|$ ，其中 Γ'' 指的是 Γ 的二階導數(second derivative)，而 γ 為一權重因子，如果 Γ 像外凸出， γ 則為正值；若 Γ 向內凹入，則 γ 則為負值，如圖 3.8，圖中藍色箭頭處 γ 則為正值，綠色箭頭處 γ 則為負值。

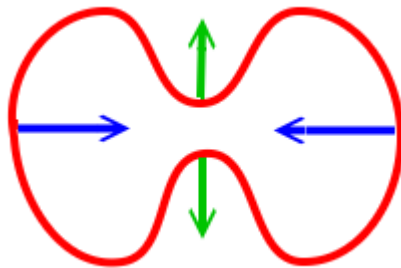


圖 3.8、內部力示意圖

(3) 圖像力(image force) F_{img} ：為了讓曲線能被物件邊界吸引 F_{img} 被定義為

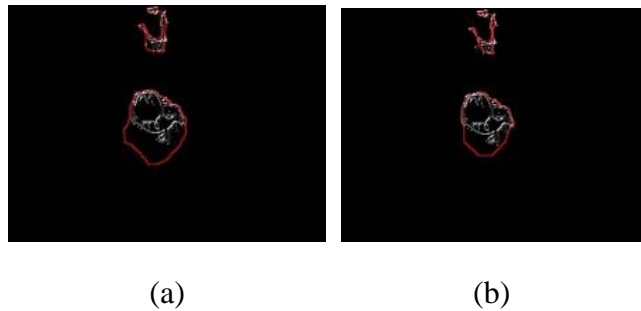
$F_{img} = \beta |E_{obj}|$ ，其中 β 為一權重因子。如果 $F_{ext} + F_{int}$ 為負值，則 β 為正值；如果 $F_{ext} + F_{int}$ 為正值， β 設為負值。而 E_{obj} 為物件邊緣強度。

實際上，我們將像點分成兩個集合 Γ_{in} 與 Γ_{out} ，對於像點 $(x, y) \in \Gamma_{in}$ 連接到其他像素 $(x, y) \in \Gamma_{out}$ 計算 $F(x, y)$ 。如果 $F(x, y) > 0$ ，這個像點將會被移動到 Γ_{out} ，也就是說，位於曲線內的像點數量變少，讓曲線會往內收縮。而可以藉由計數在鄰近像點範圍 $(2n+1) \times (2n+1)$ 內屬於 Γ_{in} 的像點個數來計算曲線的曲度。而曲度被定義成：

$$\Gamma'' = n_s - n \times (2n+1) \quad , \quad (3.21)$$

其中 n_s 為計數結果。為了避免曲線的收縮時發生毀壞，我們可以固定 β 值為負值，並根據 n 來調整 α ，使得曲線有較好的形狀。

最後，我們檢查每次曲線更新的結果，如圖 3.9，如果曲線的改變不大，即可視為曲線已收縮完畢，此時曲線內的像點集合 Γ_{in} 即是我們的前景物，也就是行人偵測步驟的結果，所得結果如圖 3.10。



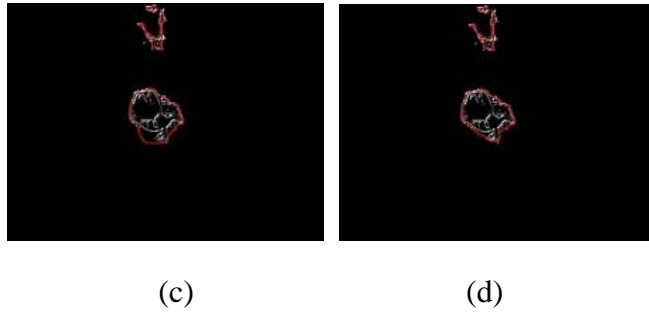


圖 3.9、曲線收縮過程圖(a)第一次曲線更新(b)第二次更新
(c)第三次更新(d)第四次更新



圖 3.10、前景物擷取結果

3.4 暫時停滯物決定

暫時停滯物決定的目的是將決定出何者為暫時停滯物並將其儲存在系統中，以便下一張影格進來時，能夠我們幫助找出注意力區域。這邊我們將介紹如何將暫時停滯物儲存以及找出暫時停滯物的方法。

首先，我們藉由儲存暫時停滯物件邊界內的所有像素集合來保存物件：

$$M_i = \{(x, y) | (x, y) \in i\text{-th object's region}\}, i = 1 \dots m \quad (3.22)$$

對任意一個擷取出來的前景物 O^j ，利用物件的特徵將它與 M_i 比較差異度，以了解 O^j 是否為已儲存的暫時停滯物。比較結果會發生兩種狀況：

(1) 若差異很小，代表可能為同一物體發生了暫時停滯。對於一個發生暫時停滯的物體，我們於每過一個時間後，增加 M_i 的停留時間(stay time)。且為了避免錯誤儲存到曾經移動過的背景物(如：行李、手推車、光影變化等)，故在時間超過一定門檻值(本研究定為 12)後，我們必須將其移除。

(2) 如果差異過大，則代表 O^j 為一個移動物體，我們可以直接將它儲存起來，留做下一個時間判斷是否為一個暫時停滯物。

我們也可以說，被擷取的物件 O^j 如果有一個小的差異值 $S(O_j)$ 則會被儲存起來：

$$S(O_j) = \min \{s_i(O_j), i = 1 \dots m\} \quad (3.23)$$

$$s_i(O_j) = \frac{|M_i - O_j| \times C + \sum_{(x,y) \in M_i \cap O_j} |I^t(x,y) - I^{t-1} \times \alpha|}{|M_i|} \quad (3.24)$$

其中 C 為一個常數值(如：256)，而 $|M_i - O_j|$ 以及 $|M_i|$ 指的是區域大小。在(3.24)

式中，一個沒有改變位置的物件(即 $|M_i - O_j| = 0$)且顏色改變

($\sum |I^t(x,y) - I^{t-1} \times \alpha|$) 很小，會有一個較小的差異值。如果物件過去已被儲存

過，我們就將增加 M_i 的停留時間。比對所有場景中的移動物後，那些沒有增加停留時間的 M_i 便會從記憶體中移除。

在上述的相似程度條件下，還有一些問題可能發生，例如：移動物跟暫時停滯物之間發生相連的時候，或者一個暫時停滯物分裂為多個的時候。然而這些會導致物件形狀改變而產生的問題，都可以考慮為新的物體產生而不需處理之。最

後需要注意到，在 3.2 節中所提到，由於注意力區域跟有移動物區域皆為二值化影像，在將影像中有移動物出現區域 D' 及上個時間的暫時停滯物 M_i^{t-1} 在結合一起成為注意力區域 A^t 的時候，必須先將已儲存的暫時停滯物轉換為二值化影像後，才能進行合併。

第四章 行人計數

在行人計數階段中，為了要統計場景中的進出人數，必須藉由追蹤連續影像中的前景區塊，得知行人的行進路徑，再利用追蹤的結果定義出行人進入或離開場景以進行數量統計。本研究將行人計數分為兩個步驟：行人追蹤及行人數量統計。由於行人追蹤步驟較為繁瑣，為了方便說明，本論文將先對所使用的追蹤技術粒子濾波器作簡介，再針對行人追蹤技術的定義及流程做描述。接下來先說明粒子濾波器追蹤單一目標的實作情況與步驟，才探討同時追蹤多人的實作方法及注意事項。

本論文的4.1節為粒子濾波器簡介，4.2節為行人追蹤，在4.3節為實作粒子濾波器追蹤單一行人的步驟，4.4節是相似程度的定義，而4.5節介紹粒子濾波器用於多個行人追蹤。最後於4.6節中定義行人的進出狀態以進行數量統計。

4.1 粒子濾波器簡介

粒子濾波器運用了數值方法上非常成功的 Monte Carlo(MC)技術，能有效且準確的算出因傳統方法無法逼近多維度或變因複雜的真實物理現象而不容易計算出的數學方程式。Monte Carlo 法是由 Stanislaw Marcin Ulam, Enrico Fermi, John von Neumann 和 Nicholas Metropolis 等四位學者所提出，主要是由於 Los Alamos(美國國家實驗室中子散射研究中心)為應曼哈頓計劃的需求而發展出來的，當時這方法是用來研究模擬可引起核分裂的材料中中子的擴散運動，也就是

說為了發展原子彈而研究出的方法。因為這方法的機率、亂數的產生和賭博有關，加上 Ulam 為了紀念他的賭徒舅舅，因此才以最著名的賭場 Monte Carlo 來命名。

Monte Carlo 法主要理念為，當所要求解的問題是某種事物出現的機率或者某隨機變量的期望值時，可以透過實驗的方法，求得該事物出現的概率或者該變量的平均值，做為該問題的解。而 Monte Carlo 法來解決實際問題時，即是利用尋找事物的運動過程或特徵，利用數學方法加以模擬實驗。Monte Carlo 方法的應用有兩個核心觀念：仿真和取樣(sample)，仿真是指提供近似於實際現象的隨機程序並以數學模仿之的方法；取樣是指通過研究少量的子集來演繹大量母體集合特性的方法，所以 Monte Carlo 法也是隨機過程和概率統計這兩大重要理論的合併。當取樣的次數越多，也就是進行的模擬實驗次數越多，其模擬結果越接近真實值。所以 Monte Carlo 法說明白了就是一種暴力解題的方法，但往往這種暴力方法在處理複雜問題的時候是非常有效的。

粒子濾波器基於 Monte Carlo 法，藉由粒子模擬中子在空間中運動的方式，對於物體的運動進行仿真。由於中子之間可能發生碰撞或漂移，其運動軌跡是類似鋸齒形的隨機路徑，所以粒子的移動為模仿中子隨機運動方式。而粒子濾波器使用一定數量的粒子來進行物體狀態的估測，則是基於 Monte Carlo 法中取樣的動作。

粒子濾波器亦是傳統卡曼濾波器(Kalman filtering)的一般化技術，因為卡曼

濾波器是假設物體的運動模型是呈線性、高斯分佈的，而粒子濾波器是可以被運用於非線性、非高斯模型的濾波器(filter)，其特點是可以得到近似任意函數的期望值，就是粒子濾波器的原始理念。粒子濾波器主要是利用粒子的質量(particle mass)來表示機率的密度，利用一定數量隨機樣本粒子來表示系統狀態的事後機率(posterior)分佈，並用粒子的期望值當作的估測結果。

將粒子濾波器應用於物體追蹤上時，由於所追蹤的物體通常為非線性的運動方式，所以使用非線性的粒子濾波器追蹤效果會比較好。粒子濾波器之所以準確度高而不容易將追蹤物追丟的原因，主要在於它對追蹤物的狀態一次提出了多個隨機的假設，並同時保留多種可能性較高的狀態假設，並非只留下一種預測狀態。所以當物體狀態驟變導致前一個時間的預測出現錯誤時，可以靠其他可能性高的狀態也將錯誤狀態修正，也就是說系統具有能夠自動修正偵測結果的能力。由於以上粒子濾波器特性，對於容易受到不同攝影機特性及架設環境影響的前景區塊，無論其產生的各種類型的變化，粒子濾波器皆能準確追蹤。

在早期粒子濾波器剛提出來的時候，因為此方法在使用時往往需要大量且重覆的產生亂數才能得到正確結果，受到計算時間影響，可以使用的應用受到限制。所幸隨著電腦設備的進步，電腦運算速度越來越快，粒子濾波器也重新被學者們重視，此技術應用也愈來愈廣，如物件偵測、物件追蹤等。

4.2 行人追蹤

完成行人偵測步驟後，我們所取得的行人區塊可能會隨著行人之間的移動發生分裂及合併的問題，例如：多個行人可能被擷取成同一區塊，並於之後才分裂成數個區塊；也可能單一行人區塊因為背景或邊緣不明顯被分成數個小區塊，諸如此類不同時間下區塊間彼此合併的問題。為了解決或容忍這些行人偵測階段無法解決的缺失，本研究藉由粒子濾波器所提供隨機模型來估測下一時間點行人出現位置、形狀大小，以進行行人區塊的追蹤。

粒子濾波器主要基於貝氏定理，對於一個追蹤的問題，我們定義以下表示法：

$$\mathbf{x}_k = f_k(\mathbf{x}_{k-1}, \mathbf{v}_{k-1}) \quad , \quad (4.1)$$

\mathbf{x}_k 為 k 時間的狀態，其中 f_k 為一個非線性系統使得時間 $k-1$ 狀態由轉變成時間 k 的狀態， \mathbf{v} 則為一個非高斯的雜訊(noise)。然而實際的情況下，我們無法直接求得想要估測的狀態，而是經由一個觀察(observation)：

$$\mathbf{z}_k = h_k(\mathbf{x}_k, \mathbf{n}_k) \quad , \quad (4.2)$$

其中 h 為一個非線性觀察轉換函式，將目前的狀態經由觀察，得到目前的觀測值 \mathbf{z} ，而在觀察的時候也常有誤差與干擾 \mathbf{n} 的影響。若以貝式(Bayesian)機率的觀點來看，假設時間點 $k-1$ 時所需要的 pdf (probability density function) $p(\mathbf{x}_{k-1} | \mathbf{z}_{1:k-1})$ 為已知，則在時間點 k 時，該系統事前機率(prior)的表示法如下：

$$p(\mathbf{x}_k | \mathbf{z}_{1:k-1}) = \int p(\mathbf{x}_k | \mathbf{x}_{k-1}) p(\mathbf{x}_{k-1} | \mathbf{z}_{1:k-1}) d\mathbf{x}_{k-1} \quad . \quad (4.3)$$

其中 $p(x_k | x_{k-1})$ 稱為轉移模型(transition model)或運動模型(motion model)。

當時間點 k 時， z_k 為已知，根據貝式定理，此時事後機率(posterior)可由事前機率和觀察模型(observation model) $p(z_k | x_k)$ 組成：

$$p(x_k | z_{1:k}) = \frac{p(z_k | x_k) p(x_k | z_{1:k-1})}{p(z_k | z_{1:k-1})} \quad (4.4)$$

其中 $p(z_k | x_k)$ 可以根據(4.2)式定義，且 $p(z_k | z_{1:k-1})$ 為一個正規化常數(normalizing constant)定義如下：

$$p(z_k | z_{1:k-1}) = \int p(z_k | x_k) p(x_k | z_{1:k-1}) dx_k \quad (4.5)$$

觀察(4.3)式中 $p(x_{k-1} | z_{1:k-1})$ 和(4.4)式的中 $p(x_k | z_{1:k})$ ，我們可以知道此系統是經由遞迴(recursive)運算，不斷求更新現在事後機率的狀態。

本研究根據 MC 方法，將要追蹤的非線性系統利用粒子密度來模擬時間 k 的事後機率密度(posterior density)，並以粒子集合(particle set)代表此系統，則該系統可以被近似(approximate)為：

$$p(x_k | z_{1:k}) \approx \sum_{i=1}^{N_s} w_k^i \delta(x_k - x_k^i) \quad (4.6)$$

其中 $z_{1:k}$ 為時間 1 到時間 k 的系統觀測值，(4.6)式利用 N_s 個粒子 x_k^i 描述了在連續觀測 k 個時間下的觀測值，目前的狀態 x_k 的機率分佈情形，並且每一個粒

子 x_k^i 會有一個質量權值 w_k^i ，而 $\delta(g)$ 為 Dirac delta function。當 $N_s \rightarrow \infty$ 時後，(4.6)式會逼近真實的事後機率密度 $p(x_k | z_{1:k})$ 。

而質量則可以運用前一個時刻的質量與系統觀測值的機率模型來表示，如下：

$$w_k^i \propto w_{k-1}^i \frac{p(z_k | x_k^i) p(x_k^i | x_{k-1}^i)}{q(x_k^i | x_{k-1}^i, z_k)}, \quad (4.7)$$

其中 q 函式為重要性密度(importance density) 函式，也就是取樣粒子分佈函數， $p(x_k^i | x_{k-1}^i)$ 為運動模型(motion model)。在實際(practical)情況中，當重要性密度的選擇為事前機率時，使得：

$$q(x_k^i | x_{k-1}^i, z_k) = p(x_k^i | x_{k-1}^i)。 \quad (4.8)$$

將(4.8)代入(4.7)式經過化簡，之後可以得到，粒子的質量只與粒子的觀測函數及前一個時刻的質量有關：

$$w_k^i \propto w_{k-1}^i p(z_k | x_k^i), \quad (4.9)$$

其中觀測模型 $p(z_k | x_k^i)$ 則可由粒子與追蹤物體的相似程度(likelihood)來定義，且依據不同應用情況，(4.9)式亦可令粒子的質量與前一個時刻的質量無關。運用這個方法，可以估測到每一個時刻狀態 x 的機率，並且利用期望值得到目前可能的狀態值 x 。

由以上定義，可規劃粒子濾波器追蹤物體的流程，其步驟如下：

step1. 初始粒子系統(Initialization)

定義所需要的粒子個數 N ，並由事前機率(prior) $p(x_0)$ 初始化一個粒子集合

$\{x_0^i, w_0^i\}_{i=1}^N$, x_0^i 每個粒子初始狀態及初始權值 w_0^i , 令 $k=1$ 。

step2. 取樣(Sampling)

(a) 移動粒子

對於 $i=1, \dots, N$,

根據運動模型 $p(x_k^i | x_{k-1}^i)$ 取樣出 x_k^i 。

(b) 計算新的權重植

$$w_k^i = p(z_k | x_k^i), i=1, \dots, N \quad (4.10)$$

(c) 權重植正規化(Normalize)

$$w_k^i = \frac{w_k^i}{\sum_{j=1}^N w_k^j}, i=1, \dots, N \quad (4.11)$$

step3. 輸出(Output)

一個粒子集合的事後機率由 $p(x_k | z_{1:k}) \approx \sum_{i=1}^{N_s} w_k^i \delta(x_k - x_k^i)$ 來近似，並藉由期望

值狀態(expect state) 作為粒子集合的輸出結果：

$$E(f_k(x_k)) = \sum_{i=1}^N w_k^i f_k(x_k^i) \quad (4.12)$$

step4. 重新取樣(Resampling)

對每個 x_k^i 經由 w_k^i 的機率分佈重新取樣出新的 N 個獨立且隨機的粒子 x_k^j 使其

能符合事後機率 $p(x_k | z_{1:k})$ 。

step5. $k=k+1$, go to step2.

整個粒子濾波器步驟流程圖，如圖4.1，而粒子濾波器流程的示意圖，如圖4.2：

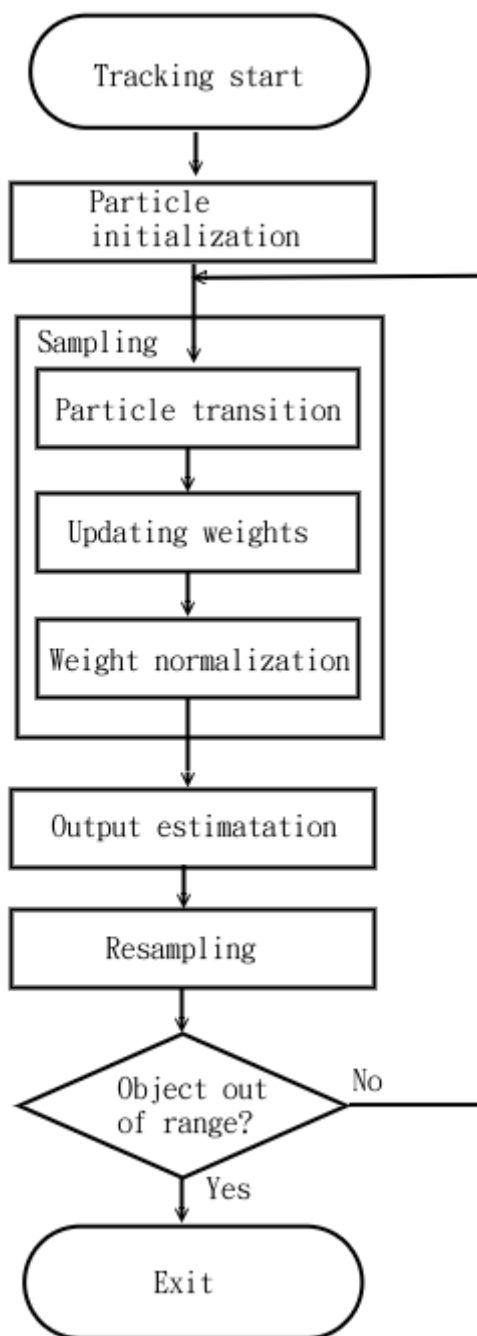


圖 4.1、粒子濾波器步驟流程圖

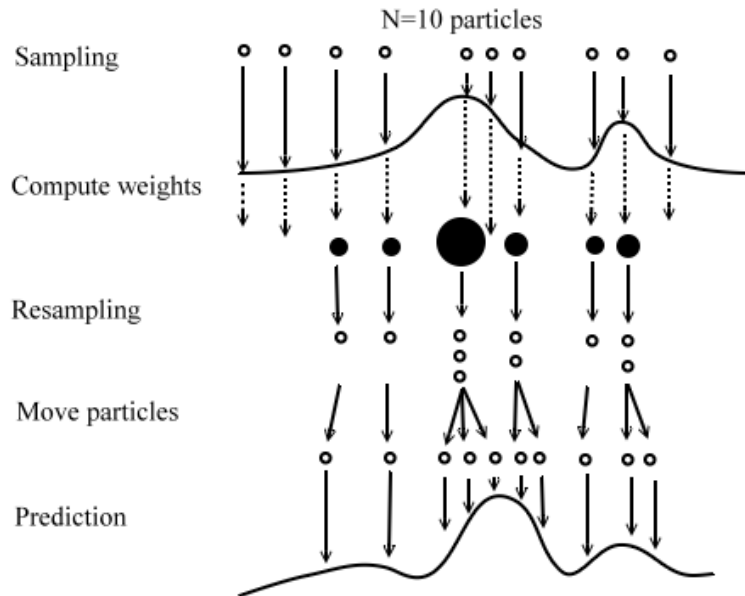


圖 4.2、粒子濾波器流程示意圖

4.3 實作粒子濾波器追蹤單一行人

由於多人追蹤需考慮更多細節，為了能說明整個粒子濾波器的追蹤流程，在這邊我們先假設影像中只有一個行人，而且擷取出該行人的區塊是完整的。多人追蹤的應用將留到 4.5 節再作說明。

由於粒子濾波器依賴大量的粒子來模擬追蹤物的事後機率，粒子個數多的話，各類特徵所形成的狀態假設組合也愈多種，對於形狀或特徵較多變的物體，能提出更多元的假設，而能更準確追蹤。然而根據相關研究顯示[Isa96]，當粒子個數大於一定個數的時候，系統的準確度無法獲得更多的提升，過多的粒子個數可能會產生重複的組合，導致執行速度過慢、降低系統效能的缺點。本研究定義粒子集合所需要的粒子個數 N 為 100，因為 $N=100$ 時，當使系統能正確追蹤

行人，卻又不會影響系統執行效能。而當 $N \leq 50$ 的時候，系統容易追蹤失誤。

當 $N \geq 200$ 時，則會有計算時間較久的問題。

在第二個影格(幀)(frame)進來後(第一個影格時，擷取前景資訊還不夠)，我們利用第三章前景擷取的方法，由前後兩張影像中得到的前景區塊，對於每個區塊，計算其質心(mass center)(X_{obj}, Y_{obj})、面積(area)($Area_{obj}$)、最小外接矩形(bounding box)的長跟寬 ($width, height$)、及色彩直方圖(color histogram)等資訊。利用這些資訊，計算出行人的狀態值，以 $S = [x, y, v_x, v_y, a, b, \theta, w, t]$ 。其中包括行人的形狀資訊，以橢圓形表示：中心位置(x, y)、長軸(a)、短軸(b)、旋轉角度(θ)，行進資訊：水平及垂直移動速度(v_x, v_y)，權重植(w)，及所屬粒子集合存在的時間(t)。

由於一開始追蹤的時候，我們不知道行人的行走方向，為了不失一般性，我們將粒子的初始位置散佈於該行人區塊周圍，並令其分佈狀況符合高斯分佈。所以對於每一個粒子其初始速度為：

$$v_x = gaussian_random * c_1$$

$$v_y = gaussian_random * c_2$$

其中 $gaussian_random$ 為介於-1 到 1 且產生機率符合高斯分佈的隨機亂數(附錄 1)， c_1 和 c_2 為一個常數，用來將亂數縮放(scale)到我們所需要的範圍。

粒子位置的計算方式為行人區塊質心座標加上初始速度：

$$x = X_{obj} + v_x$$

$$y = Y_{obj} + v_y$$

當中 X_{obj} 為行人區塊質心 X 座標、 Y_{obj} 為行人區塊質心 Y 座標。除此之外我們也要注意粒子的位置是否有超出影像範圍，若超出影像邊界則以該邊界座標為粒子座標。初始橢圓的長軸、短軸則為行人區塊最小外接矩形長與寬的一半：

$$a = \frac{width}{2}$$

$$b = \frac{height}{2}$$

至於旋轉角度、權重植、及粒子集合存在的時間定義如下：

$$\theta = 0$$

$$w = \frac{1}{N}$$

$$t = 0$$

除此之外，本研究以行人前景區塊的色彩直方圖作為粒子集合所追蹤行人的初始色彩直方圖。

在粒子濾波器第二個步驟取樣中，為了提升粒子濾波器追蹤效果，實作上將控制粒子集合整體移動方向的運動模型 $p(x_k^i | x_{k-1}^i)$ 設定為相隔一張影格下，行人在影像中移動的平均距離，其定義如下：

$$motion = \begin{cases} \frac{h}{t_a r_f} & , \text{ if moving direction is In.} \\ \frac{-h}{t_a r_f} & , \text{ if moving direction is Out.} \end{cases} \quad (4.13)$$

其中 h 為影像高度， t_a 為一個行人以正常行走穿越監測區域的平均時間， r_f 為幀率(frame rate)。而考慮到行人移動受出入口的影響，大致行進方向分為進(in)或

出(out)，並由行人的進與出決定運動模型的方向，至於如何判斷行人行進方向(moving direction)將於第 4.6 節行人統計交代細節。

為了節省運算，本研究將重新取樣步驟中，隨機移動粒子的程序併入取樣步驟中。對於所有集合中一個粒子，其資訊更新方式如下：

$$v_x = random * c_1$$

$$v_y = random * c_2$$

$$x = x + v_x$$

$$y = y + motion + v_y$$

$$a = a + random * c_3$$

$$b = b + random * c_4$$

$$\theta = \theta + random * c_5$$

其中 *random* 為隨機亂數， $c_1 \sim c_5$ 為常數。各粒子所產生的狀態橢圓結果，見圖

4.3。

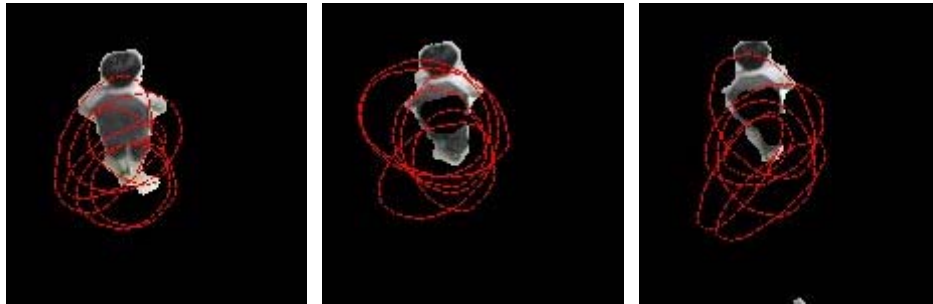


圖 4.3、(N=6 時)粒子狀態橢圓分佈圖(a)時間為 t-2 時(b)t-1 時(c)t 時

粒子系統的觀測模型 $p(z_k | x_k^i)$ 就是指粒子與追蹤物的相似程度，為了判斷一個粒子所代表橢圓狀態特徵跟所追蹤的行人是否相似，本研究使用了三種特徵

來定義：粒子所形成之橢圓與行人前景之間的形狀大小相似程度(詳見 4.4.1.)、
 粒子位置與行人前景之間的距離相似程度(詳見 4.4.2)以及粒子狀態橢圓與其所
 追蹤的行人色彩直方圖相似程度(詳見 4.4.3)，然後依據這三種相似程度更新權
 重值 w_i 。計算完每個粒子的權重之後再利用(4.10)式將權重值作正規化，使整個
 粒子集合的權重值加起來總和為 1。

再來本研究使用(4.12)式計算了整個粒子集合的期望值狀態，以顯示目前追蹤
 的成果。因為我們用橢圓來顯示輸出結果，所以我們僅需計算出橢圓中心位置、
 長軸、短軸、旋轉角度這幾個參數的期望值，以期望值橢圓中心位置 (E_x, E_y) 為
 例：

$$E_x = \sum_{i=1}^N w_i x_i \quad ,$$

$$E_y = \sum_{i=1}^N w_i y_i \quad ,$$

其中 w_i 為第 i 個粒子的權重值， (x_i, y_i) 為第 i 個粒子的中心座標。期望值的輸出

結果如圖 4.4，紅色橢圓為每個粒子的狀態橢圓，綠色橢圓為期望值狀態橢圓。

最後，考慮到行人的顏色可能隨著時間有所改變，且本研究將期望值狀態視為目

前追蹤行人的狀態，所以將追蹤目標的色彩直方圖更新為期望值狀態橢圓所包含

到之前景區塊的色彩直方圖，使追蹤目標的色彩直方圖也能隨著追蹤物體變化。

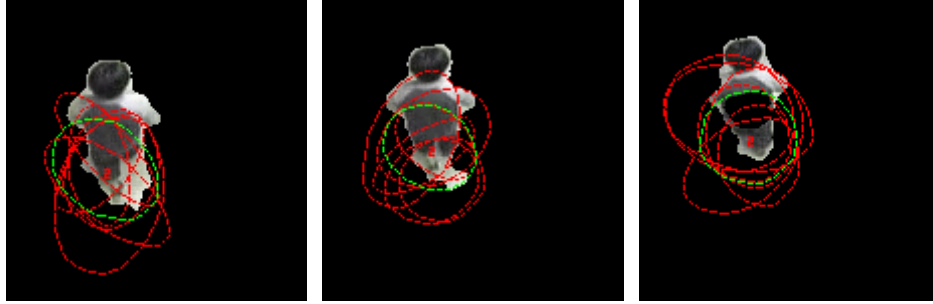


圖 4.4、($N=6$ 時)期望值輸出結果 (a)時間為 $t-2$ 時(b) $t-1$ 時(c) t 時

由於粒子濾波器是基於隨機程序來產生出各種假設的狀態，其中有些粒子狀態在計算權重後，會被給予極低的權重值，也就是說此粒子的狀態不是我們所期望的結果。為了避免這樣的變化被保留下來，影響到後續的追蹤結果，我們有重新取樣的動作，將不適合的變化去除(這也粒子濾波器別稱為 survival of the fittest 的原因)。

本研究規劃的重新取樣的方法為將權重值低於 $\frac{1}{N}$ 的粒子移除，再將權重值較大的粒子，按其權重的大小計算其應分為幾個粒子，完整步驟如下：

Step1. 找出 $w_k < \frac{1}{N}$ 的粒子 k ，將其 w_k 設為 0。

Step2. 對於每個 $w_j \geq \frac{1}{N}$ 的粒子 j ，計算照其權重大小，計算其應分為 n_j 個

particle，其中 $n_j = \left\lfloor \frac{N * w_j}{\sum_{i=1}^N w_i} \right\rfloor$ 。

Step3. 將粒子 j 分割成 n_j 個粒子，其 (x, y) 、 (v_x, v_y) 、 a 、 b 、 θ 、及 t 設定給新

產生的粒子，而 $w = \frac{1}{N}$ 。

然而要注意的是因為分數化為整數時取 floor 函數(即捨去正小數至最近整數)計算的關係，產生 $\sum n \leq N$ 的情況，所以當 $\sum n$ 小於 N 時，這 $N - \sum n$ 個粒子設定給權重最大的粒子。在重新取樣過後，我們完成了一次粒子濾波器預測。在下一個影格進來的時候，將時間 t 加 1，回到取樣步驟，完成一個迴圈(iteration)的計算。

4.4 相似程度及定義

前面提到本研究在判斷一個粒子所形成的橢圓跟擷取出來的前景物是否相似時，使用了三種特徵來定義相似程度：(1) 粒子位置與行人區塊之間距離相似程度、(2) 粒子狀態橢圓與行人區塊之間的形狀大小相似程度、以及(3)粒子與追蹤目標行人的色彩相似程度。其詳細方法分別詳述於 4.4.1、4.4.2、以及 4.4.3。

4.4.1 距離相似程度定義

粒子位置與行人區塊之間的距離定義方法有很多，如：Euclidean、Manhattan、Chessboard、Chamfer(圖4.5)，一般最常見的影像中物件之間的距離是經由Euclidean distance方法求得，而Manhattan及Chessboard則非常少見。但是Euclidean distance的計算需使用平方及開根號運算，若應用在粒子濾波器中，會受到粒子數量多寡而影響效能。再者粒子集合與前景區塊的對應情況，不一定是一個粒子集合對應到一個前景區塊，例如：當前景擷取出不完整或破碎的區塊

時，對於一個粒子集合而言，使用Euclidean distance並無法表現出這種多(破碎區塊)對一(particle set)的關係。

取 chamfer distance 運算快速的優點，本系統採用 chamfer distance 作為粒子位置與前景物體之間距離的描述，在每次影格輸入後，僅需計算一次 chamfer distance map 即可。

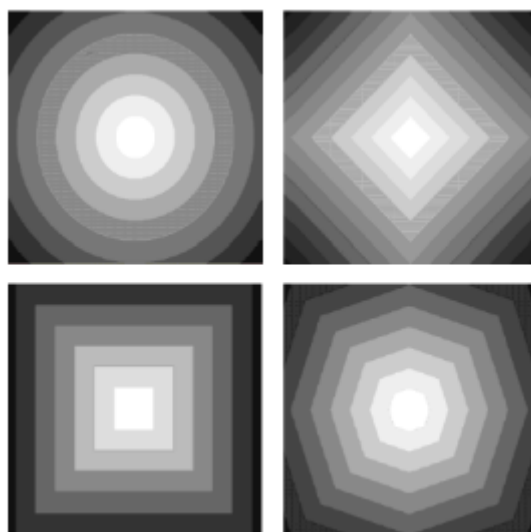


圖4.5、各種類型的距離定義(a) Euclidean(b) Manhattan

(c)Chessboard(d)Chamfer

用來計算距離的 chamfer distance map 可以採用 Rosenfeld 與 Pfaltz 提出演算法[Fou03]，這個演算法利用 chamfer mask 在”forward pass”與”backward pass”中，對影像做遮罩(mask)運算兩次，以快速計算出 chamfer distance map。當前景影像(圖 4.6 a)I輸入後，由於考慮到之後行人判定步驟所需的邊界比對，我們先針對

前景做邊界偵測(boundary detection)：對前景影像中每一個像素作檢查，如果不是前景，則像素值設為 0，如果是前景則進一步檢查其八近鄰中有任一個像素的像素值為 0，則像素值設為 255，否則設為 0，並令此影像為 I_B (圖 4.6 b)， $I_B(x, y)$ 為影像 I_B 於座標 (x, y) 的像素值。

再來將邊界偵測後的影像 I_B 轉換成 chamfer distance map I_C (圖 4.6 c)。整個 chamfer distance 演算法藉由二次遮罩運算達成，將 chamfer mask 分成兩個左上半跟右下半部份 (圖 4.7)，演算法步驟如下：

Step1. 令長為 width，寬為 height 的影像 I_B 中的為邊界的像素設為 0，否則設為無限大(∞)

Step2. forward pass：對影像 I_B 從左上到右下掃描計算：

for($x = 1, x < \text{width} - 1, x++$)

for($y = 1, y < \text{height} - 1, y++$)

$$I_B(x, y) = \min(I_B(x, y), I_B(x-1, y) + 2, I_B(x-1, y-1) + 3,$$

$$I_B(x, y-1) + 2, I_B(x+1, y-1) + 3) ;$$

Step3. backward pass：對影像 I_B 從右下到左上掃描計算：

for($x = \text{width} - 2, x > 0, x--$)

for($y = \text{height} - 2, y > 0, y--$)

$$I_B(x, y) = \min(I_B(x, y), I_B(x+1, y) + 2, I_B(x+1, y+1) + 3,$$

$$I_B(x, y+1) + 2, I_B(x-1, y+1) + 3) ;$$

此時 I_B 被轉換為 chamfer distance map I_C 。

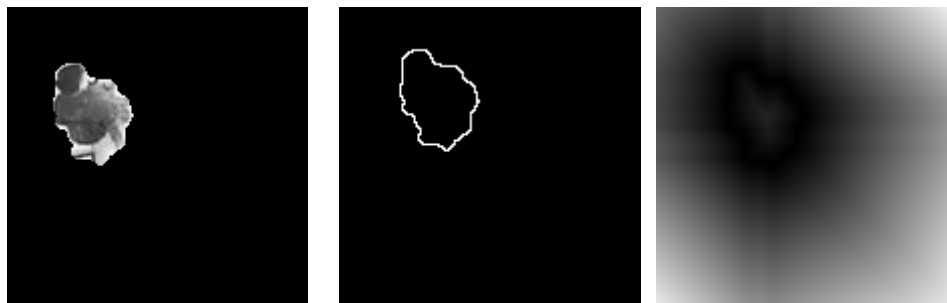


圖 4.6、chamfer 影像轉換圖(a)前景影像(b)邊界偵測後的影像(c) chamfer 轉換後

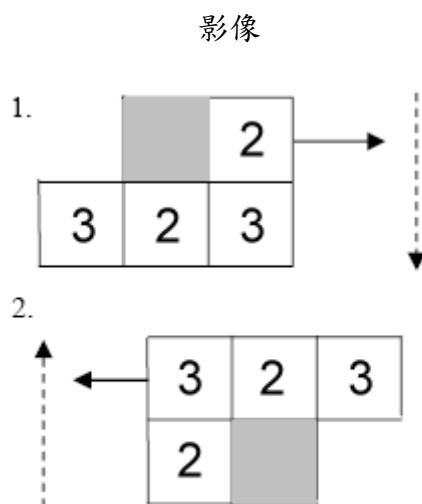


圖 4.7、chamfer mask 示意圖(2 passes)

但是在距離相似程度定義中，我們需要的是粒子中心到前景物體所需的距離，即是說粒子中心若在前景物上距離即應為 0。所以將 I_C 中像素位置其對應到前景影像 I 中相同像素位置且為前景的像素設為 0，並令此時的影像為 I_C' ，使 I_C' 位於前景邊界內部的距離值也為 0。所以當我們需要一個粒子中心 (x, y) 到前景物體所需的距離 d ，只需查詢像素值 $I_C'(x, y)$ 。

4.4.2 形狀大小相似程度定義

每一個粒子有都一個橢圓代表著它的狀態，而本研究則藉由該橢圓的大小及形狀定義與行人區塊相似程度，在粒子濾波器計算權重時，挑選出合適代表追蹤物的橢圓並給予較高的權重。這邊要注意的是相似程度注意到不只是橢圓的面積大小(長軸、短軸長度)影響，橢圓的形狀(旋轉角度、長短軸長度比)也扮演著很重要的腳色。圖 4.8 中，假設紅色的橢圓代表著一個粒子所形成的橢圓，其面積為 $A+B$ ，計算公式為 πab ，黑色的輪廓的為可能為行人的前景，面積為 $B+C$ ， B 部分代表兩者交集的面積，可藉由統計前景影像中前景區塊邊界矩形內符合公式(4.14)的像素數量求得。

$$0 \geq b^2(X \cos \theta - Y \sin \theta)^2 + a^2(X \sin \theta - Y \cos \theta)^2 - a^2b^2 \quad , \quad (4.14)$$

$$X = |I_x - x| \quad ,$$

$$Y = |I_y - y| \quad ,$$

其中 I_x 、 I_y 為像素座標， x 、 y 為粒子橢圓中心座標。

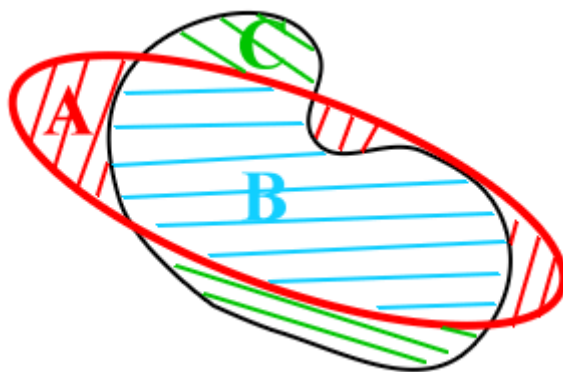


圖 4.8、粒子橢圓與前景區塊關係示意圖

一般來說，我們希望橢圓能包含越多交集區域越好，也就是說面積比： $\frac{B}{A}$ 要大，但是這樣的選出來的橢圓可能會完全包含在前景物中，導致橢圓過小的情況；然而若只考慮要橢圓將前景物都包住，即是面積比： $\frac{B}{C}$ 要大，則會造成所選擇的橢圓過大。所以我們期望 $\frac{B}{A}$ 要大且 $\frac{B}{C}$ 也同時要大，這時選出的橢圓才是大小及形狀比較合理的。若將這兩項改變分母成 $\frac{B}{A+B}$ 及 $\frac{B}{B+C}$ ，即可以改以概率(0~1)的觀點來看，也就可以視為觀測模型 $p(z_k | x_k^i)$ 。為了求得 $\frac{B}{A+B}$ 及 $\frac{B}{B+C}$ 兩項 $\frac{B}{C}$ 、 $\frac{B}{A}$ 同時都能找到較高的比值，所以本研究採用以下方法定義來定義形狀大小相似程度 f ：

$$f = \min\left(\frac{B}{A+B}, \frac{B}{B+C}\right) \quad (4.15)$$

這邊的 \min 帶有著取交集之意，並將其定義成相似程度，在粒子濾波器計算權重值時，即是找出粒子能使得 $\frac{B}{C}$ 、 $\frac{B}{A}$ 兩項同時都能找到較大，並給予較高的權重。

4.4.3 色彩相似程度定義

當兩個行人區塊非常靠近的時候，若只使用形狀大小或距離位置作為追蹤時的兩區塊的相似程度，常會導致追蹤錯誤或追錯人的情況，為了解決此一問題，本研究將顏色資訊考慮到相似程度中。色彩相似程度定義中，本研究使用粒子所包含的行人區塊的色彩直方圖模型與所追蹤行人的色彩直方圖模型作為比較，並在建立直方圖時將色彩由 256 種分為 m 段區間(bin)(本研究取 $m=64$)，如圖 4.9。

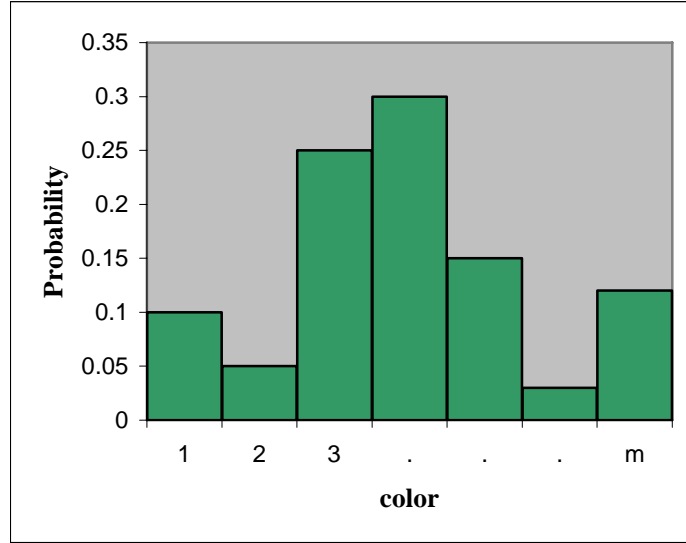


圖 4.9、色彩直方圖示意圖

首先將所追蹤的行人模型稱為目標物模型(target model)，而粒子所包含的行人區塊的模型稱為追蹤目標的候選物體(target candidate)模型。令 $\{x_i\}_{i=1\dots n}$ 為屬於所追蹤行人的像素，將目標物的模型定義為：

$$q_u = C \sum_{i=1}^n \delta[h(x_i) - u] \quad , \quad (4.16)$$

其中 δ 為 Kronecker delta function， C 為一正規化常數，其值為 $\frac{1}{n}$ ，使得

$\sum_{u=1}^m q_u = 1$ ， $h(x_i)$ 為 x_i 落於色彩直方圖的區間編號， u 為所有區間的編號索引 (如：1、2、...)。

再令 $\{x_i^k\}_{i=1\dots n'}$ 為與第 k 個粒子的狀態橢圓有交集之行人區塊的像素，而候選物模型可表示成：

$$p_u^k = C' \sum_{i=1}^{n'} \delta[h(x_i^k) - u] \quad , \quad (4.17)$$

C' 為一正規化常數，其值為 $\frac{1}{n'}$ ，使得 $\sum_{u=1}^m p_u^k = 1$ ， $h(x_i^k)$ 為 x_i^k 落於色彩直方圖的區間編號。

為了找出候選物模型與目標物模型之間的相似程度，我們採用統計上常用的 Bhattacharyya 係數作判斷：

$$\rho[p^k, q] = \sum_{u=1}^m \sqrt{p_u^k q_u} \quad , \quad (4.18)$$

由於 p_u^k 及 q_u 皆經過正規化，所計算出的 Bhattacharyya 係數也是介於 0~1 之間的概率分佈。而 Bhattacharyya distance 的分佈可描述為：

$$d_B = \sqrt{1 - \rho[p^k, q]} \quad 。 \quad (4.19)$$

如果兩個愈相似，則 Bhattacharyya 係數愈大，Bhattacharyya distance 則愈小。由於根據我們的相似程度定義，粒子所包含的行人區塊的色彩直方圖模型與所追蹤行人的色彩直方圖模型相似時，給予該粒子較高的權重值，所以我們可以使用 Bhattacharyya 係數作為顏色相似程度定義，並考慮到 RGB 三種色彩頻道的 Bhattacharyya 係數，並將其結合：

$$c = \frac{\rho_R[p^k, q] + \rho_G[p^k, q] + \rho_B[p^k, q]}{3} \quad 。 \quad (4.20)$$

其中 ρ_R 代表紅色部分的 Bhattacharyya 係數， ρ_G 代表綠色部分， ρ_B 代表藍色部份。

最後，我們將三種相似定義結合起來，由於這三種相似程度皆為介於 0~1 之間的概率，所以將這三種相乘起來作為粒子權重值的定義：

$$w = f^k * \frac{1}{d^2} * c \quad , \quad (4.21)$$

其中 f 為粒子所形成之橢圓與物體之間的相似程度值， k 為調整 f 重要性的常數， d 為粒子位置與前景物體之間 chamfer distance，並且權重值與距離平方成反比， c 為色彩相似程度。此外，對於當相似程度值為 0 時候的例外處理也是必需注意的，例如： d 為 0 的時候，將 $\frac{1}{d^2}$ 這項設為 1，以表現出高度相關。

4.5 粒子濾波器用於多個行人追蹤

在 4.2 節中，本研究提到以粒子濾波器追蹤單一物體的演算法，然而面對多個物體需同時追蹤的時候，在進行前景擷取時，可能因為行人衣服顏色與背景相似或邊緣不明顯，產生出多個破碎的區塊。亦可能由於行人彼此之間的接觸，如：牽手、搭肩等親密動作，導致產生較大的區塊其中包含多個行人。區塊之間也可能在人群行走時，產生合併或分裂的情況。所以在追蹤多人時，無法假設每個行人剛好對應到一個完整的前景區塊，我們遭遇到何時需初始新的粒子集合以及如何定義區塊合併或系統追丟後粒子集合消滅的問題，並且定義出解決方法。本節中將詳述粒子濾波器於多人追蹤應用中的定義與方法，圖 4.10 為多人追蹤流程圖。

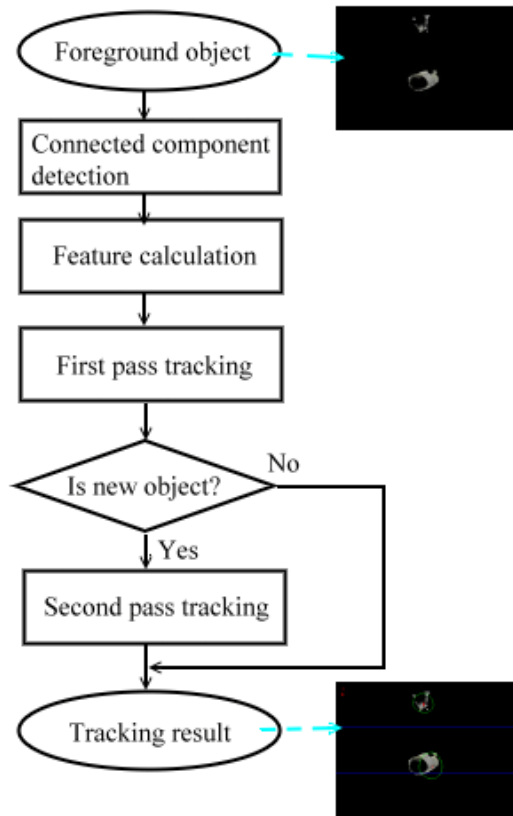


圖 4.10、行人追蹤流程圖

由於在新的前景影像進來時，無法直接了解哪塊前景區塊是由新進入的行人產生的，或者是已被粒子集合追蹤的行人所產生的前景，所以我們不能決定要初始幾個新的粒子集合或者判定是否有粒子集合失去追蹤目標。本研究將整個多人追蹤系統的流程分為兩個階段(two passes)：一、對於初始過的粒子集合進行追蹤，二、找出未有粒子集合追蹤的區塊，給予初始化一組粒子集合追蹤之。

當原始影像輸入後，首先本研究擷取出可能為行人的前景區塊，對這些區塊作一次連通區域(connected component)偵測，將各區塊作編號(label)，並計算各區塊的質心、面積、最小外接矩形的長、寬、及色彩直方圖。

在第一階段，對於所有已經過初始的粒子集合追蹤時，取樣、計算權重、計算期望值輸出都按照 4.3 節的單一物體追蹤方式即可，差異處在於對行人區塊與粒子集合之間對應的處理。在求得期望值狀態橢圓後，本研究使用(4.13)式檢查所有包含在該橢圓範圍之內的前景，若有前景區塊在橢圓範圍之內，將其記錄起來，設定此粒子集合追蹤該區塊；若沒有前景區塊在橢圓範圍內，則設定此粒子集合追蹤失敗，並消滅之。通常追蹤失敗原因可能為追丟、行人離開場景、或前景偵測錯誤(如：破碎的前景)。不過根據我們定義的方法，能夠容忍較為破碎的前景，這些前景只要有被狀態橢圓所包含，仍能有良好的追蹤效果。此外有時因為前景偵測不完整(如：破碎的前景)或雜訊影響時，出現一些存在時間短暫的區塊，而產生多餘的粒子集合。為了解決此問題，所以當破碎區塊合併時，本研究採用下列方法，使粒子濾波器對於這樣的情況具有一定的程度的容錯能力。對於兩個粒子集合之間中心距離過近且其中一個為剛初始不久的粒子集合，則將時間小的粒子集合消滅：

$$\begin{cases} \text{if } \sqrt{(E_x^i - E_x^j)^2 + (E_y^i - E_y^j)^2} \leq \lambda \text{ and } \tau < t^i \leq t^j & \text{delete particle set } i \\ \text{if } \sqrt{(E_x^i - E_x^j)^2 + (E_y^i - E_y^j)^2} \leq \lambda \text{ and } \tau < t^j < t^i & \text{delete particle set } j \end{cases} \quad (4.20)$$

其中 E_x^i 、 E_y^i 代表粒子集合 i 的中心座標期望值， λ 為面積門檻值， τ 為時間臨界值。之後再對沒有追丟的粒子集合作進行重新取樣的動作。

完成第一階段後，沒有被粒子集合追蹤的前景即可能為新的行人前景區塊。

在第二階段中，由於新的前景區塊的可能來自於雜訊、或前景不完整，而這些區

塊的共同特徵是面積略小。所以本研究對面積定訂門檻值，將每一塊尚未有粒子集合追蹤且面積大於門檻值的前景區塊，初始一個新的粒子集合，並在下一次影格輸入時，開始追蹤行人。粒子濾波器用於多人追蹤的結果，如下圖 4.11。

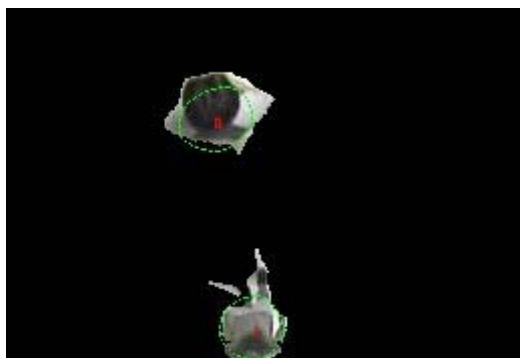


圖 4.11、多人追蹤結果

4.6 行人數量統計

本研究實作出一個雙向計數(bi-direction)的行人統計方法，將場景由上而下(或由左至右)分成三個部份：進入區域、觀測區域、離開區域。分隔這三塊區域的基線(base-line)分別為：內部基線(inside base-line)、外部基線(outside base-line)，如圖 2.2。

本研究建立五種不同狀態並使用有限狀態自動機(finite state automata)來表示行人目前的動態，以利計數。對於已初始化的粒子集合，將其視為一個行人，將每個行人以狀態的方式來表示其位置及移動方向，並分成五種狀態：開始離開、持續離開、離開場景、開始進入、持續進入。

當偵測到行人欲進入場景時，對於尚未計數且中心座標位於進入區域或觀測區域的粒子集合，將其行人狀態判斷為位於場景外部(outside)，並設定motion model為in，繼續進行追蹤，若穿越內部基線，則改變狀態為進入內部(entering inside)，當該粒子進入到外部區域時，將進入次數增加1，將粒子集合設定為已計數，並考慮到行人進入後可能會出轉向離開，行人狀態改為位於場景內部(inside)，之後行人若走出監測區域，則狀態設為離開範圍(out of range)，而行人轉身離開時，則將粒子集合設定為未計數，則當作離開監測狀態。

反之，當行人離開場景時，對於尚未計數且中心座標位於離開區域或觀測區域的粒子集合將其行人狀態判斷為位於場景內部，並設定motion model為out，若穿越外部基線，則改變狀態為進入外部(entering outside)繼續進行追蹤，當該粒子集合進入到外部區域時，我們將離開次數增加1，將粒子集合設定為已計數。並考慮到行人離開後可能會轉向再進入，行人狀態改為內部。行人走出監測區域，則狀態設為離開範圍。行人若轉身進入時，將粒子集合設定為未計數，則當作內部狀態監測。整個狀態變化若以狀態圖(state diagram)表示，如圖4.11。

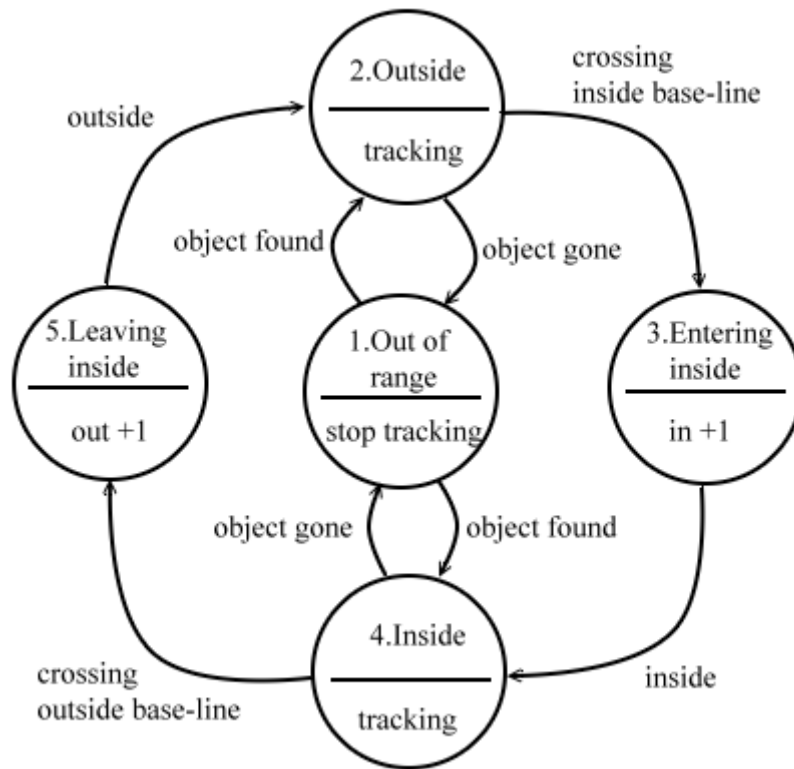


圖 4.11、行人計數狀態圖

第五章 實驗結果

實驗的影片分為兩種類型，一種是在室內場景，採用魚眼鏡頭攝影機所拍攝；另外一種是戶外於各種光影影響下(晴天、雨天或夜間)情況下使用一般攝影機拍攝。

在行人計數時，所發生的錯誤有兩種，一種是漏算，另一種是多算。漏算是指行人通過而系統沒有覺察，通常發生在行人彼此之間重疊，導致多個行人只產生一個行人區塊；或者是行人被追丟時，系統無法統計到而漏算。多算是指把不是行人的物體(比如將背包)，當成行人計數；或者將一個行人的分裂區塊，計算成兩個行人。

而一個行人計數器的準確度，通常用監測場景一段時間下，人數計算的正確率來衡量。本實驗令實際計數人數為 R ，系統計算的人數為 S ，則正確率定義為 $100\% - \frac{|R - S|}{R}$ ，這邊 $|R - S|$ 加上絕對值是因為考慮到行人可能被系統漏算或者多算的情況。

本章的實驗分為三種，第一種是當影像中前景物發生合併與分裂時，驗證我們的方法是否可以準確追蹤及計數。第二種是測試不同攝影機及不同拍攝環境下，我們的方法所受到的限制。第三種是當行人流量較高時，其計數結果的實用性。

5.1 行人區塊的合併與分裂

本實驗在這部分所使用的影像魚眼攝影機所拍攝，解析度為 320X240，每隔 0.25 秒取一張影像。擷取出影像後，使用 Pentium4-3.0GHZ 為中央處理器的桌上型電腦作影像處理，電腦的其他資料。

首先，我們將兩個行人在場景中行走所產生的區塊所可能發生的情況有幾大類，分別如圖 5.1 所示，圖(a)指的是兩個行人場景中行進，一開始走得很靠近，導致前景只擷取到一個前景物區塊，但稍後於偵測區域中兩人分開，分成兩個前景物，圖(b)則為圖(a)的相反情況。圖(c)與圖(d)為較圖(a)、圖(b)更複雜的情況，用來形容區塊之間持續發生的分裂及合併的情況。圖(e)則為當兩人行走方向相反時，可能彼此交會時過於接近，導致前景區塊發生合併或分裂的情況

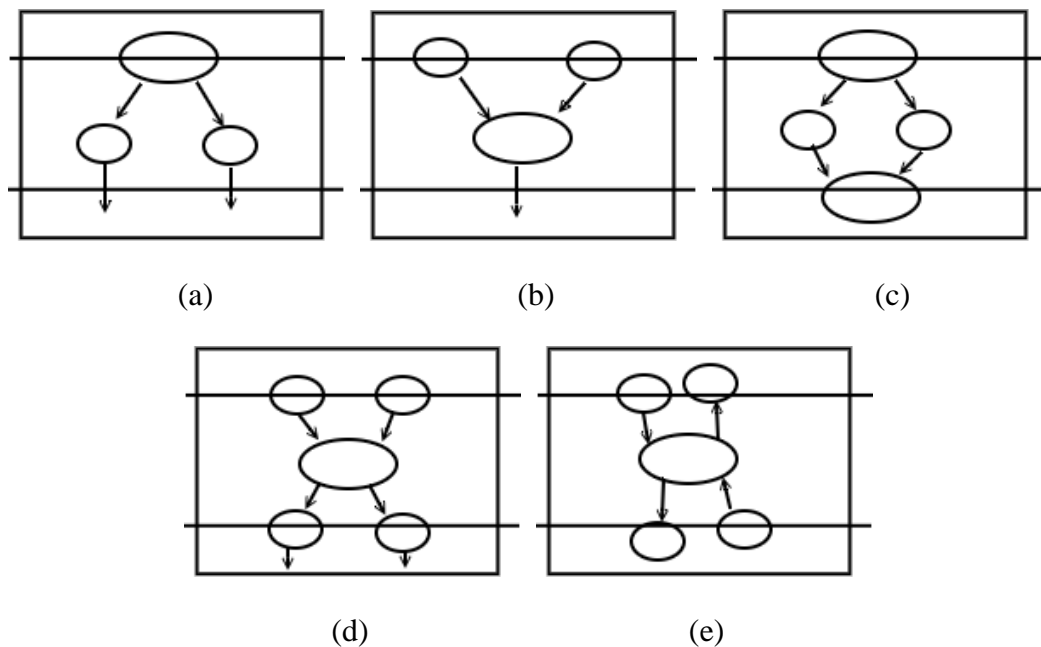


圖 5.1、合併與分裂情況圖(a) 1-to-2 split (b) 2-to-1 merge (c) 1-to-2-to-1 split-merge
(d) 2-to-1-to-2 merge-split (e) bi-directional inverse merge

在圖 5.1a 中 2-to-1 merge 情況下，系統偵測行人的追蹤和計數結果，如圖 5.2 所示，圖中狀態橢圓中心的紅色數字代表該粒子集合的編號。在圖 5.2a 時兩個人同方向進入場景時，因為兩行人之間尚有段距離，所以前景成功偵測為兩個前景區塊，系統對每位行人初始一個粒子集合追蹤之。而後在圖 5.2b、圖 5.2c 中可見，兩人在偵測區域中過於接近導致區塊合併為一個，兩個粒子集合一起追蹤該區塊，且由於兩粒子集合中心尚有一段距離，所以並未將時間較小的粒子集合刪除。最後，圖 5.2d 中兩個人通過基線，並且被正確的計數。

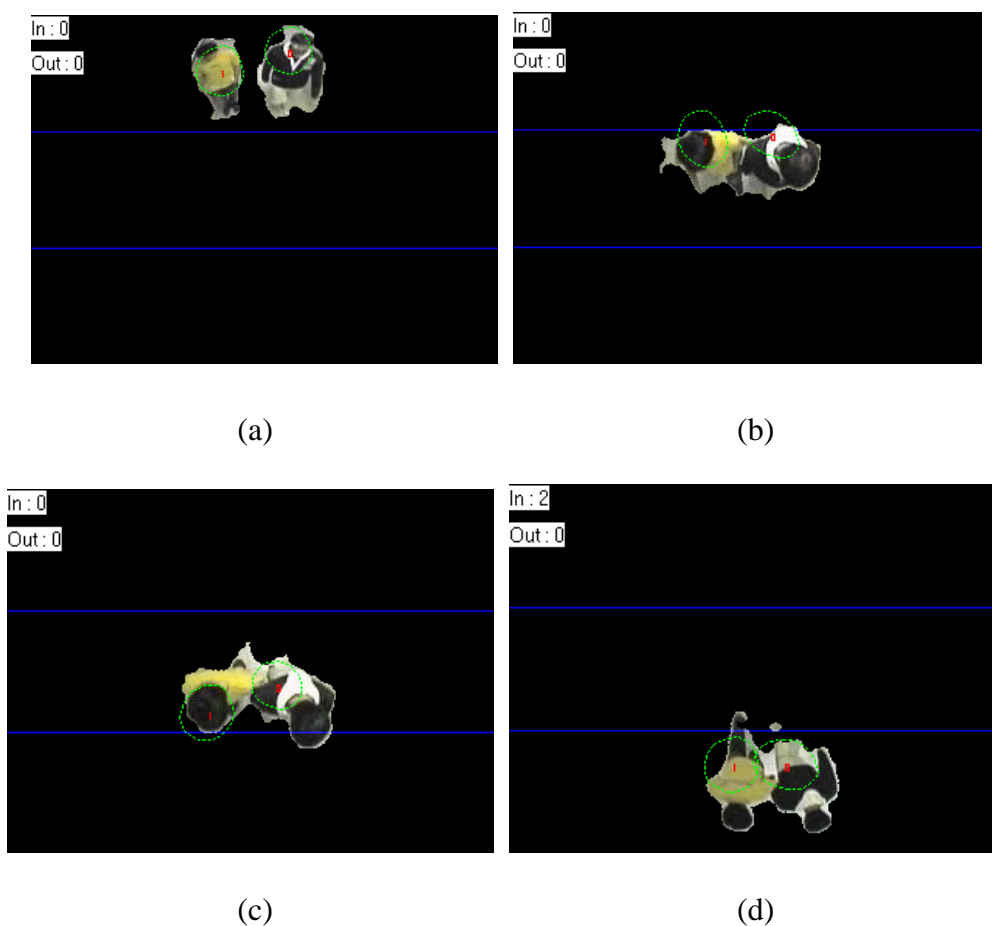


圖 5.2、2-to-1 merge 實驗結果(a)兩人分開進入場景(b)發生區塊合併(c)合併後持續追蹤(d)通過基線並完成計數

當行人行走情況類似圖 5.1b 1-to-2 split 的區塊分裂情況下，系統偵測行人的追蹤和計數結果，如圖 5.3 所示。首先，兩行人以同方向並行離開場景，如圖 5.3a。這邊要注意到，因為我們控制粒子集合狀態橢圓的大小變化範圍，所以橢圓大小無法立刻隨著兩人區塊成長般快速增大。行人進入偵測區域後，發生了分裂，圖 5.3b 中因為兩區塊仍被同一粒子集合包含，所以並未初始新的粒子，稍後在圖 5.3c 時，該粒子集合追蹤右邊的行人，左邊行人初始一個新的粒子集合，並使用紅色橢圓代表新的粒子集合。最後，圖 5.3d 正確計數。

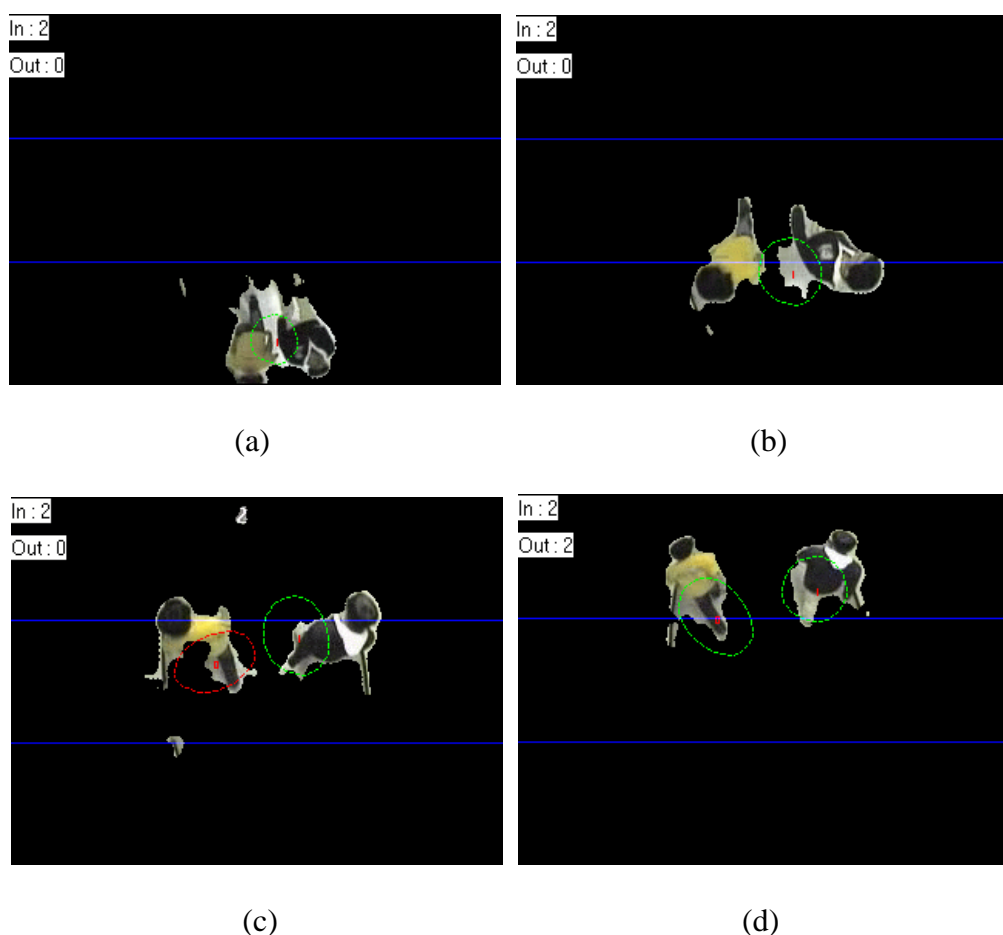


圖 5.3、1-to-2 split 實驗結果(a)兩人相鄰進入場景(b)發生區塊分裂(c)分裂後初始新的粒子集合(d)通過基線並完成計數

而行人行走區塊分裂合併情況如圖 5.1c 1-to-2-to-1 split-merge 時，系統偵測行人的追蹤和計數結果，如圖 5.4 所示。一開始，兩個行人同向並行進入場景(圖 5.4a)，在進入偵測區域後，兩人之間距離拉開，發生了區塊分裂，並且各有一個粒子集合追蹤之(圖 5.4b、圖 5.4c)，最後兩人並行離開了場景，而系統能夠正確的計數(圖 5.4d)。

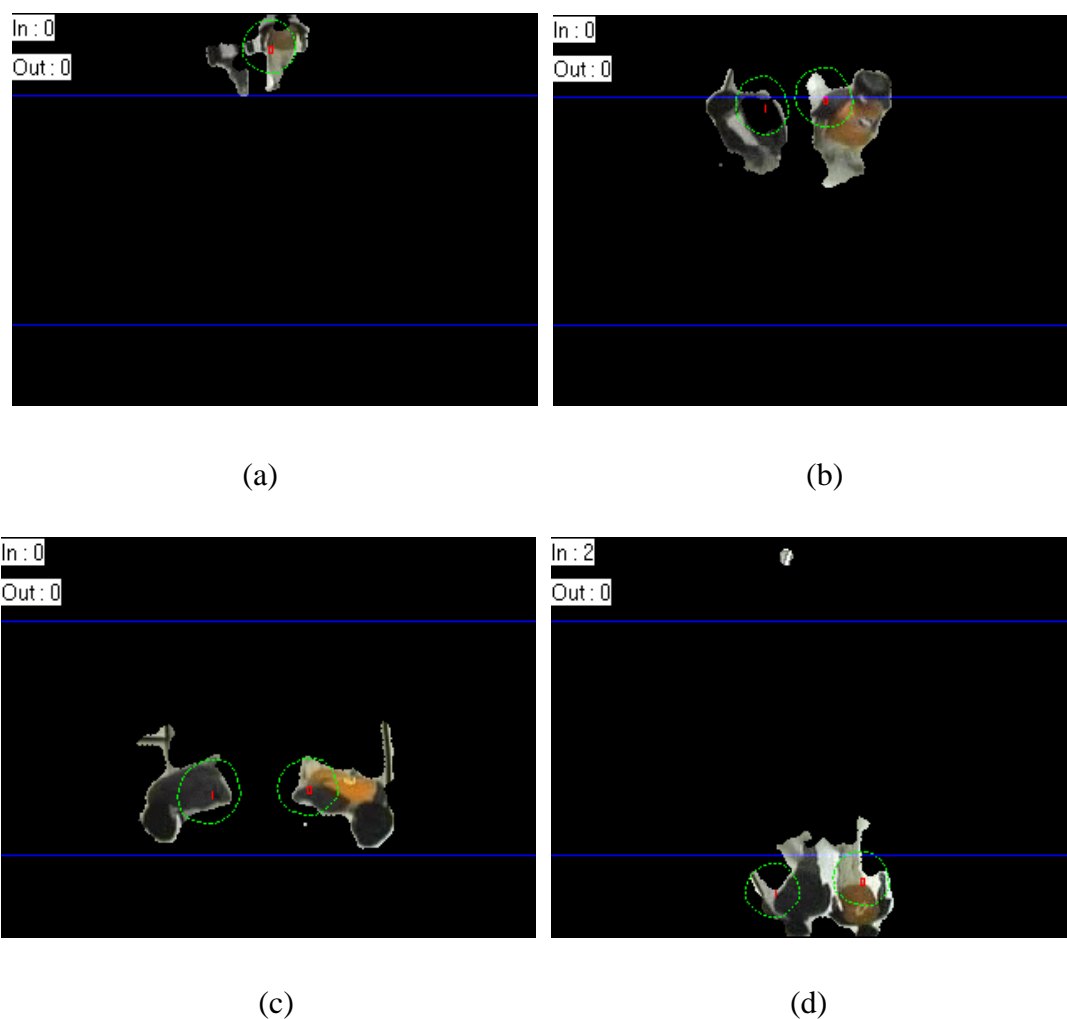


圖 5.4、1-to-2-to-1 split-merge 實驗結果(a)兩人並行進入場景(b)區塊發生分裂(c)持續追蹤(d)兩人並行離開造成區塊合併

在行人行走情況如同圖 5.1d 2-to-1-to-2 merge-split 的區塊合併分裂情況下，系統偵測行人的追蹤和計數結果，如圖 5.5 所示。兩行人同方向進入場景，並各有一個粒子系統追蹤(圖 5.5a)，在進入偵測區域後，兩人過於靠近，造成前景區塊發生合併，但是兩個粒子系統仍存在並且追蹤該區塊(圖 5.5b、圖 5.5c)，最後兩人距離拉開並且通過基線，系統正確的計數(圖 5.5d)。

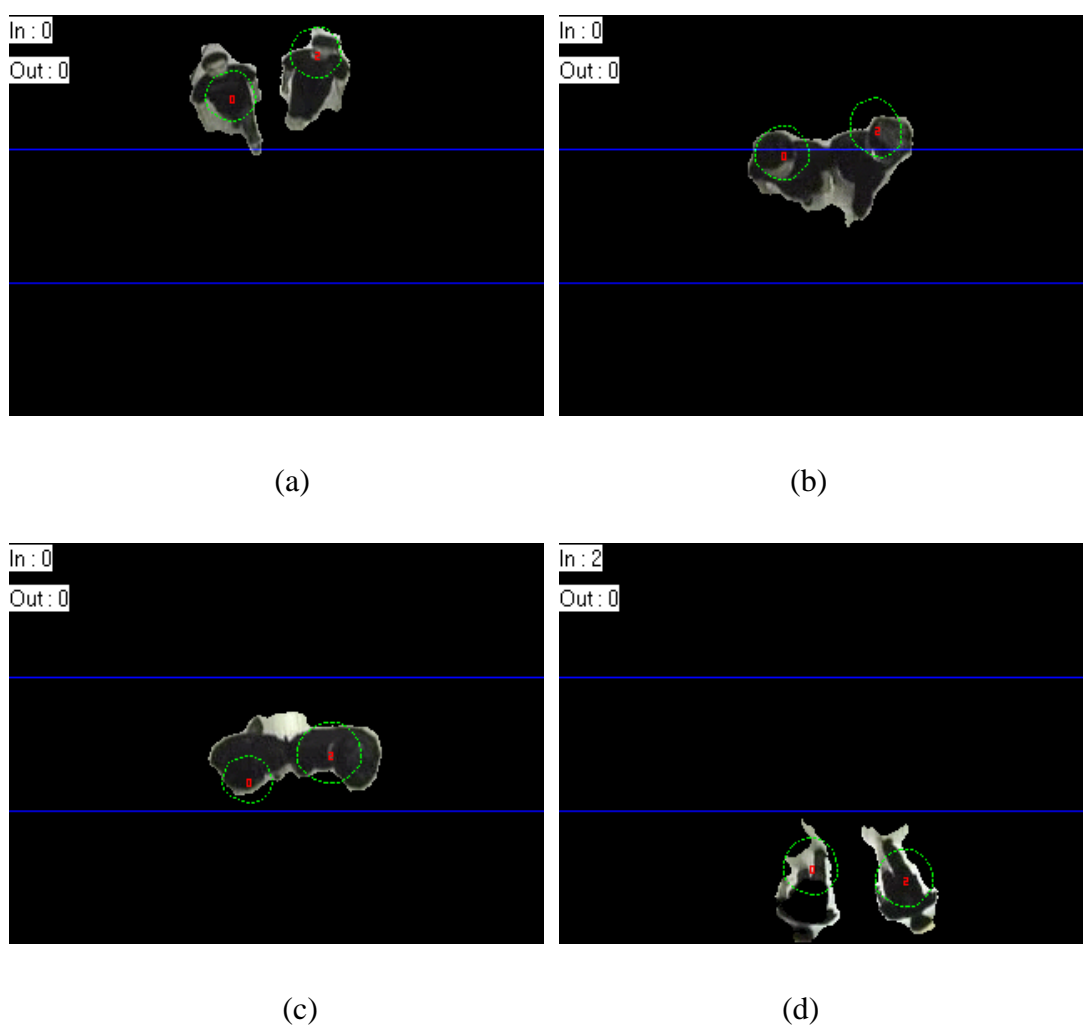


圖 5.5、2-to-1-to-2 merge-split 實驗結果(a)兩人分開進入場景(b)發生區塊合併(c)合併後持續追蹤(d)兩人分開離開場景

最後一種情況，當行人反方向行走，而類似圖 5.1e bi-directional inverse merge 的區塊合併情況下時，系統偵測行人的追蹤和計數結果，如圖 5.6 所示。兩行人以反方向進入場景(圖 5.6a)，而後在偵測區域交會，兩人接觸瞬間，兩個行人區塊合併為一個較大區塊(圖 5.6b、圖 5.6c)，但是稍後因為兩人行進方向不同，大區塊再度分裂為兩個區塊，並各自通過基線完成計數(圖 5.6d)。

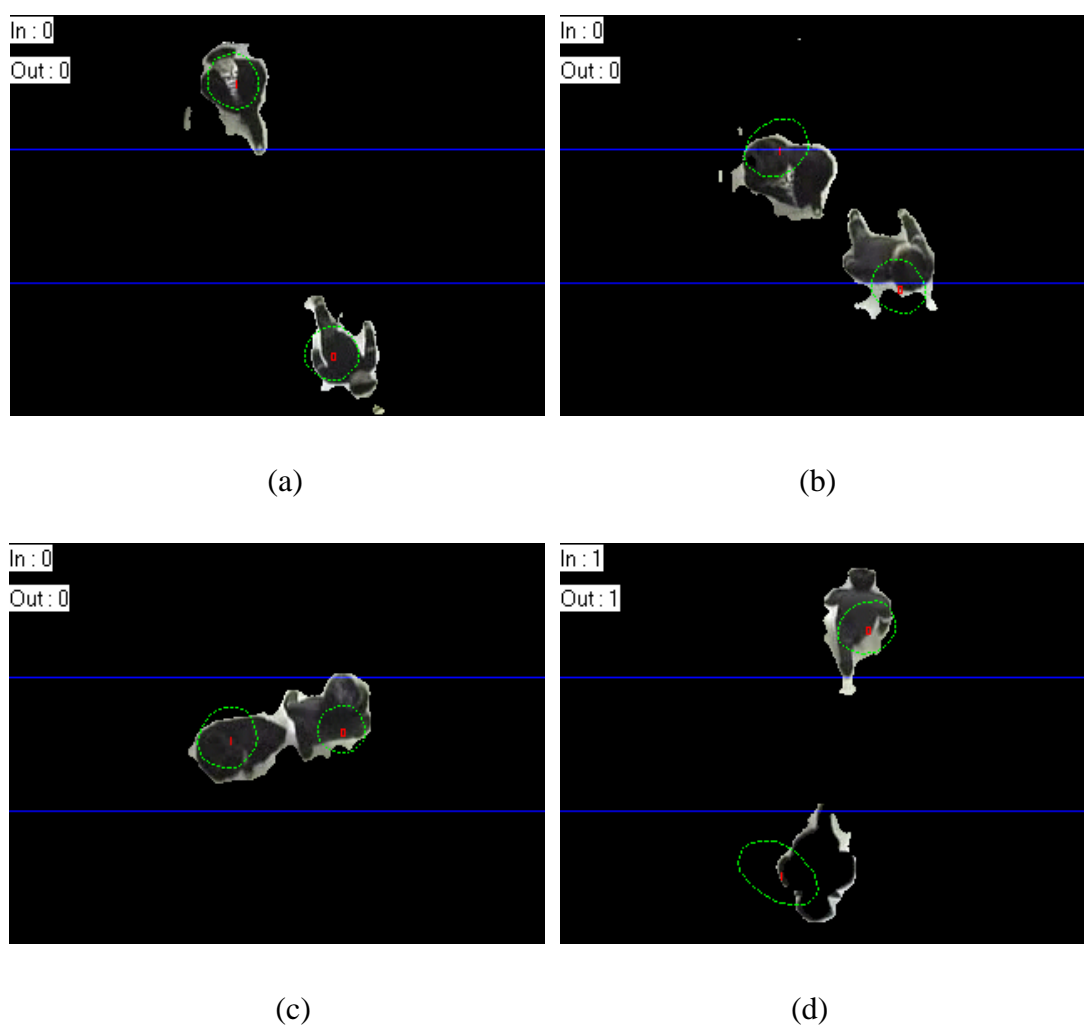


圖 5.6、bi-directional inverse merge 實驗結果(a)一人進入場景一人離開場景(b)兩人接近中(c)兩人交會，區塊發生合併(d)兩人背向而去

5.2 各種場景環境測試

在本實驗中測試了系統於戶外場景下使用一般攝影機所拍攝的影片，在這裡，我們總共測試了晴天、雨天、以及夜晚的不同場景。實驗原始影像的解析度為 640X480，在經過 sub-sampling 程序後，所處理的影像解析度為 323X215，並每隔 0.25 秒取一張影像。

在晴天場景的部份，在長約 5 分鐘的實驗影片中，由於晴天光影變化的影響較強烈(圖 5.7)，場景中的物體也陰影會較為明顯，更由於可有行人使用洋傘等工具(圖 5.8)，這些都會影響到計數結果。由於此戶外場景為開放式空間，行人行走方向較為不固定(可能往兩邊走)，行人計數結果有可能受到影響。



圖 5.7、晴天場景光影變化強烈(a)無雲經過上空(b)有雲經過上空



圖 5.8、晴天行人(a)正常行人(b)行人攜帶洋傘

圖 5.9 顯示出行人偵測及追蹤結果。圖 5.10 顯示出由於行人之間彼此相鄰太

近，產生一個包含多個行人的區塊，若此區塊從進入場景到離開場景皆未發生過分裂的情況，將會導致系統漏算。表 5.1 則顯示計數的實驗結果。



圖 5.9、晴天場景 (a)原始影像(b)偵測結果



圖 5.10、晴天場景偵測失誤情況 (a)原始影像(b)三人區塊只有兩個粒子集合

表 5.1、晴天場景之雙向計數結果

	進	出
實際人數	17	4
實驗結果	14	3

在雨天的場景中，雨水滴下的過程與滴落地面的漣漪都會造成影像的變動，此外積水中形成也會行人的倒影，影像中的雜訊較室內環境更加複雜(圖 5.11)。雖然雨天較少行人相鄰行走的問題，但是卻偶爾會見到兩個人共用一把雨傘(圖 5.13)，使得計數會發生困難。實驗影片總長度約為 5 分鐘，共有 10 個人次進出，圖 5.12 則顯示偵測與計數結果，表 5.2 顯示出系統在雨天仍能有良好的計數準確度。

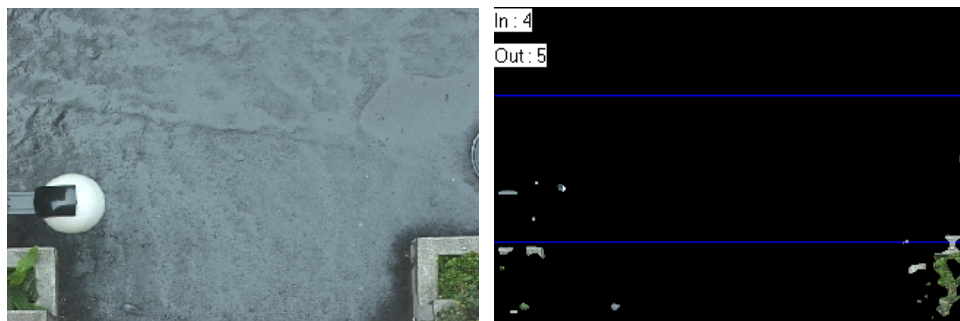


圖 5.11、雨天場景困難點 (a)場景影像(b)雜訊影響

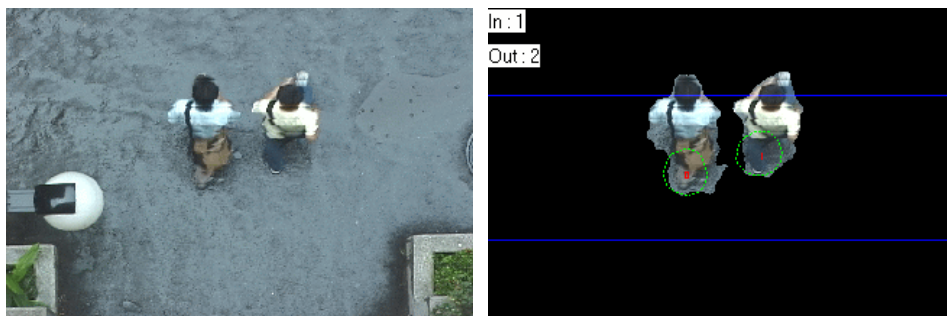


圖 5.12、雨天場景 (a)原始影像(b)偵測結果

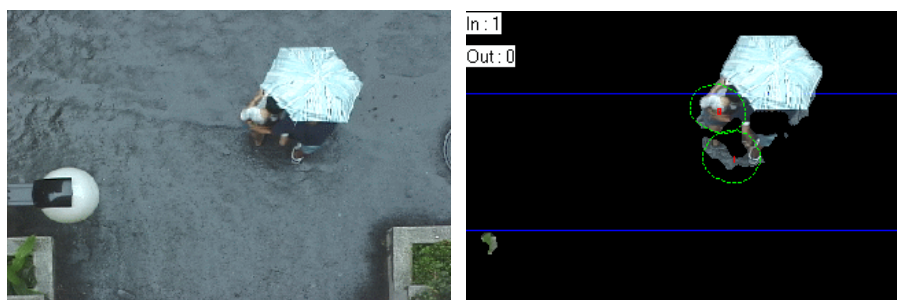


圖 5.13、兩人共用雨傘 (a)原始影像(b)偵測結果

表 5.2 、雨天場景之雙向計數結果

	進	出
實際人數	6	4
實驗結果	6	4

夜晚的場景中，場景中的光線來源為路燈，然而路燈所提供的光線並不充足(圖 5.14a)，所以這樣的特點使得陰影部份所產生的邊界較晴天場景模糊，對實際計數影響有限(圖 5.14b)。實驗影片總長度約為 20 分鐘，因為夜間行人較少，共只有 15 個人次進出場景，圖 5.15 顯示出偵測與計數結果，表 5.2 顯示出系統在夜間也能有較高計數準確度。

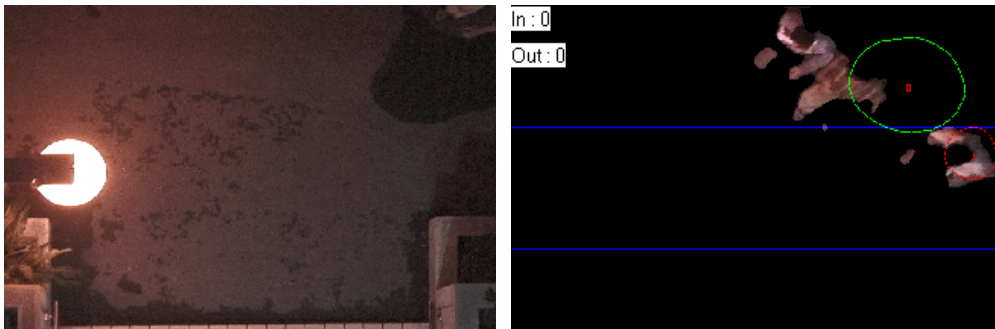


圖 5.14、夜間場景環境 (a)場景影像(b)陰影影響較少

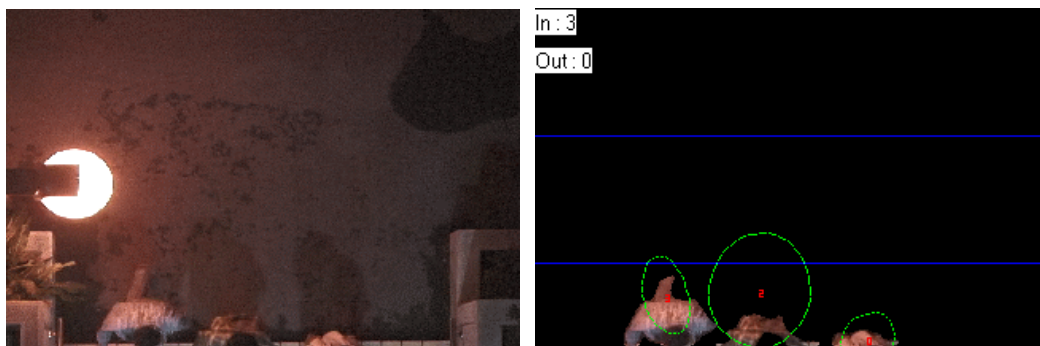


圖 5.15、夜間場景三人通過 (a)原始影像(b)偵測結果

表 5.3 、夜間場景之雙向計數結果

	進	出
實際人數	8	7
實驗結果	8	7

5.3 高行人流量環境測試

高行人流量的環境是行人計數最大的挑戰，一般的行人計數系統在人群密集和人群稀疏的情況下，準確率會有所不同。這是因為當場景中行進的人數過多時，影像中變動的區域也越大，行人彼此之間也越容易產生遮蔽或者行人區塊相連的情況。

為了測試高流量行人計數，本研究設計了三個實驗，其中兩個是晴天的戶外場景，另一個是室內通道。晴天戶外場景實驗的影片皆是由一般攝影機所拍攝的影片，第一段影片總長度為 30 分鐘，第二段長度為 13 分鐘。由於攝影機是架設於正上方並且為一般攝影機，如此可將行人之間互相遮蔽的情況縮減到最小，行人區塊會有較少相連的情況。在圖 5.16 中，有六個人經過行人通道，並且同時出現在影像中，在圖 5.17a 中位於影像右下方的兩人發生了合併，以及左上方的兩人一開始的時候相連為同一區塊。在圖 5.17c 中，原本左上方相連的兩人分開了，所以在之後兩人各有一個粒子系統追蹤。最後六個人皆通過，系統也正確計算了六個人。兩個晴天戶外場景實驗的計數結果顯示於表 5.4，分別以 seq.1、seq.2

代表之。

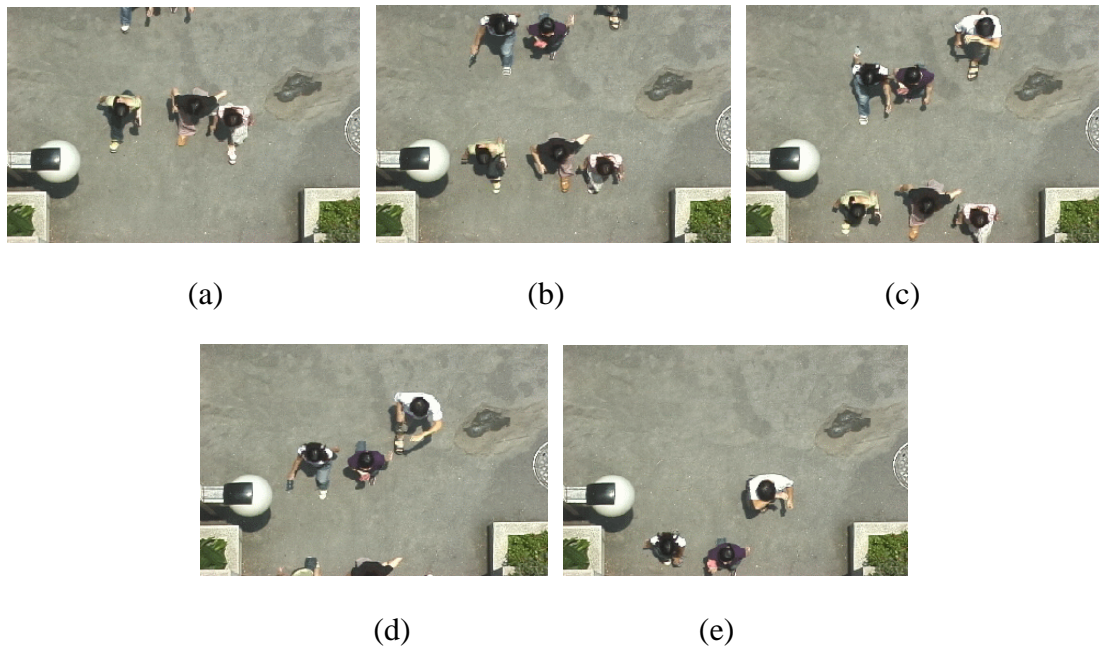


圖 5.16、晴天場景原始影像(a)第 20 影格(b)第 25 影格

(c)第 30 影格(d)第 35 影格(e)第 40 影格

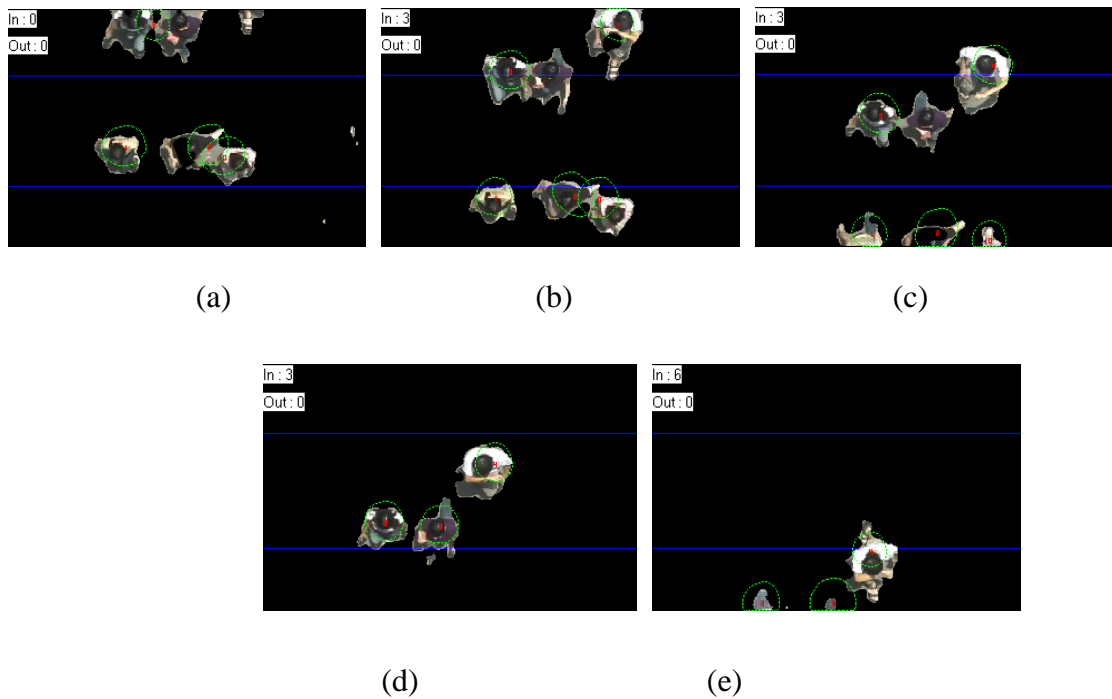


圖 5.17、晴天場景追蹤結果(a)第 20 影格(b)第 25 影格

(c)第 30 影格(d)第 35 影格(e)第 40 影格

室內場景高流量的實驗影片則為使用魚眼攝影機拍攝的影片，場景為演講聽出口的行人通道，共有兩段實驗影片，其中一段影片為 seq.3，監測時間總長度約一小時半；另一段影片為 seq.4，監測時間總長度為兩個半小時。

由於輸入的影像左上方有記錄時間，所以成為一個在固定間隔時間出現變動的雜訊，由於此雜訊不會移動而跨越基線，對計數沒有影響。此外本實驗的前景擷取法中邊緣為前景邊界的重要資訊，此場景中走道邊緣的強烈邊緣會影響到前景邊界的收縮結果，導致前景物件偵測錯誤，如圖 5.18。圖 5.19 顯示了影片中的一小段時間內的影格及追蹤結果，圖片上半段為原始影像，下半段為追蹤結果。

由圖片 5.19a 圖片、5.19b、圖片 5.19f 可發現當行人位於監測場景中心外圍時，受到魚眼鏡頭影響，行人之間發生遮蔽現象，造成多個行人擷取成一個前景區塊，當人群走到場景中心時候，遮蔽現象減少而能正確追蹤，但是仍有一些情形前景區塊在場景中心時並沒有分裂，造成人數漏算。

在編號為 seq.3 的影片實驗中，共有 61 個進入，6 個人離開，系統準確率顯示於表 5.4。由於離開人數較少，導致離開場景統計人數的誤差較為嚴重。此外，透過觀察影片可以發現位於通道左邊有一扇門正被關閉，但是並沒有影響到計數成果。在編號為 seq.4 的影片實驗中，共有 103 個人次進入場景，90 個人次離開場景，系統準確率顯示於表 5.4。由於進入場景的時候，人員進出的時間比較分散，所以準確率較高。而影片後半段中，大量行人出入時間集中在演講結束剛後的六分鐘內，行人之間重疊情況較嚴重，所以離開場景部分的人數統計準確率較

低。

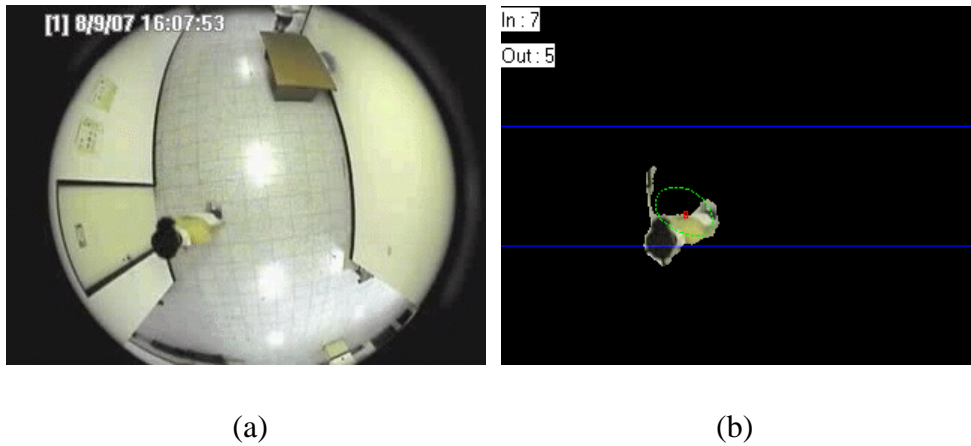
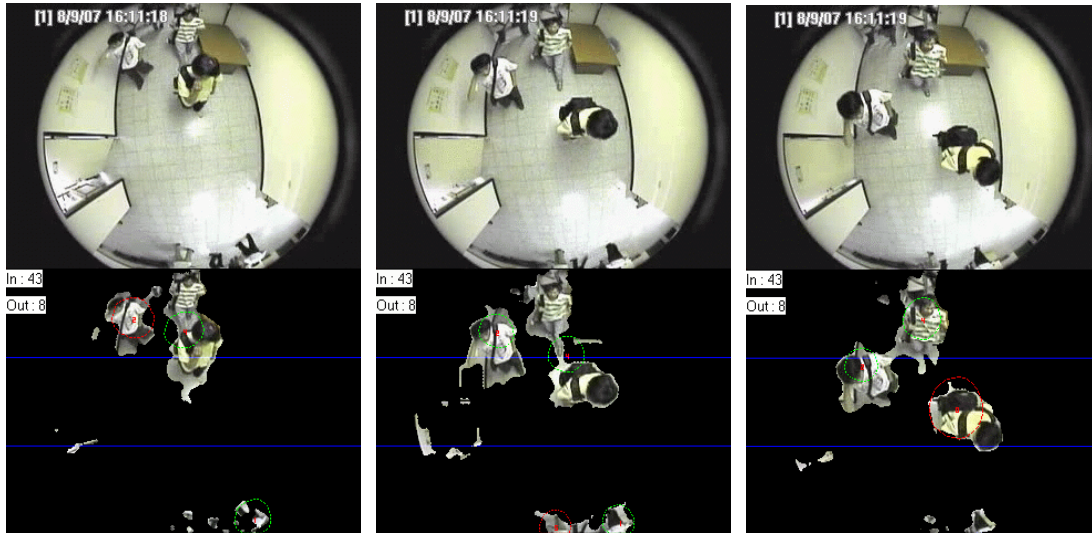


圖 5.18、前景偵測錯誤圖(a)原始影像(b)走道邊緣被截取出

表 5.4、行人計數結果統計表

影片	長度	方向	實際人數	實驗結果	正確率
Seq.1	30 min	In	130	105	80.7%
		Out	55	49	89.9%
Seq.2	13 min	In	86	71	82.5%
		Out	31	27	87.0%
Seq.3	1hr30min	In	61	55	90.2%
		Out	6	8	66.6%
Seq.4	2hr30min	In	103	102	99.0%
		Out	90	75	83.3%

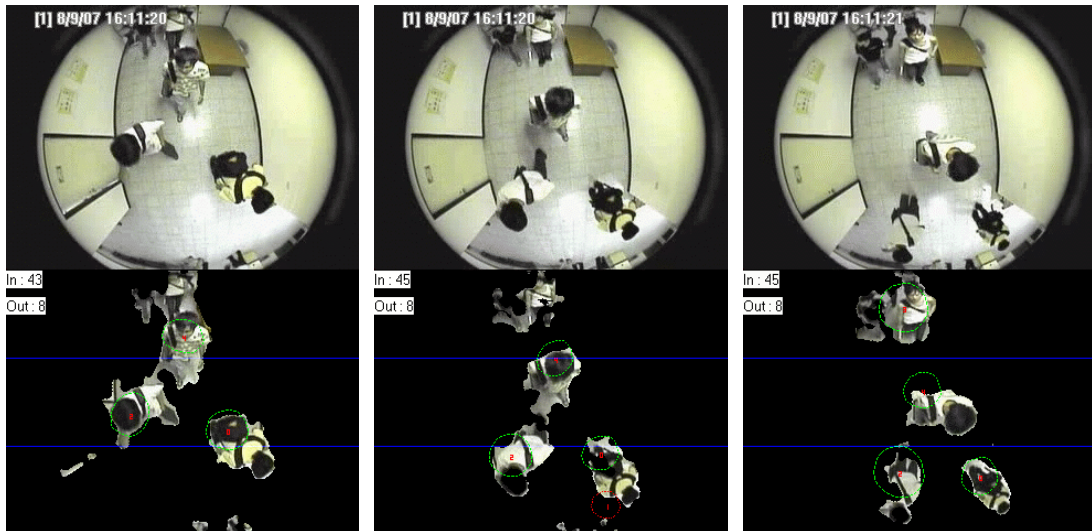
(Seq.1 與 Seq.2 為情天場景所拍攝的影片，Seq.3 與 Seq.4 為室內場景所拍攝)



(a)

(b)

(c)



(d)

(e)

(f)

圖 5.19、室內高流量場景追蹤圖(a)第 1149 影格(b)第 1152 影格

(c)第 1155 影格(d)第 1158 影格(e)第 1161 影格(f)第 114 影格

第六章 結論及未來方向

6.1 結論

近年來由於資訊科技蓬勃發展，加上數位科技產品的價格降低，各種先進的數位監控器材日益普及，已不再是富人的專利。這些數位攝影器材使用非常方便，且也很方便直接連接電腦做後續處理，可提供各種豐富的視訊影像資訊。然而目前影像資訊卻尚未被充分利用，大多數數位監測應用(如：行人計數)仍採用半人工或以全手動的方式來進行。因此，本研究提出一個以電腦視覺方法取代人工進行行人計數的系統，以便能取代傳統的非自動化或精確度不高的行人計數系統。

在行人計數的相關研究中，幾乎無可避免兩大問題，分別是前景區塊的擷取以及擷取後的行人區塊追蹤問題。本研究把俯視型行人計數系統架構主要分成行人偵測和行人計數兩個部分，行人偵測主要目的是找出影像中的行人，進行前景區塊的擷取，而行人計數為針對的擷取後的行人區塊進行追蹤，以便計算進出場景的人數。

行人偵測中，我們使用單一攝影機，藉由兩張連續影像來得到前景物。此方法避免了傳統前景擷取方法中前景區塊容易碎裂或不完整且容易受到光影影響的區塊，導致前景偵測不準確的缺點。既然此方法較不受光影變化的影響，它可以應用到各種不同的變化的場景，且適用於不同類型的攝影機。

在取得前景物件後，我們使用粒子濾波器來實作多個行人追蹤的部份。粒子

濾波器是一種以 Bayesian estimation model 為基礎的方法，可適用於追蹤非線性、非高斯模型的濾波器，其特點是利用一定數量隨機樣本粒子來表示近似任意函數的事後機率分佈，透過系統的期望值當作的估測結果。在連續影像中，單一行人產生的前景區塊，因為前景偵測錯誤而分裂為數個行人區塊，或者是場景中有雜訊影響時(如：戶外雨天場景)，粒子濾波器皆能有較強的容錯性。

在計數方面，本研究提出了一個雙向行人計數的方法。首先利用兩條偵測線將場景的影像分成三個區域，以定義場景的內部與外部，以及如何算進入或離開場景。依行人追蹤的結果，判斷行人是否跨越兩條偵測線及跨越偵測線時其所位於的場景區域，並利用行人進出狀態之間的轉換，判定何時需進行統計人數。

6.2 未來方向

在未來工作中，考慮到本系統為行人計數系統，行人才是系統偵測的焦點，希望未來階段可以增加行人判定的工作。所謂的行人判定即是使用已經建立好的人物模型在影像中進行比對，以判斷所追蹤的前景區塊是否為行人。另外藉由人物模型的部份輪廓，更可以將數個行人合併在一起的區塊，判斷該區塊包含多少個行人，以提升系統準確度。

我們可以用建立好的 3D 行人模型(圖 6.1)經由多重視角(multiple-view)(附錄 C)作 2D 投影(圖 6.2)，產生行人樣版影像(圖 6.3)，再利用分類演算法(如：K-mean)找出具有代表性的行人樣板，以去除多餘的樣版，加快比對時間。當進行行人追

蹤的時候，我們使用 Chamfer distance matching 針對行人區塊的邊界(boundary)與行人樣版的作比較，而 Chamfer distance matching 共需做兩次，一次是以行人區塊的邊界與行人樣版的 Chamfer distance 作比較，另一次是以行人樣版的邊界與行人區塊的 Chamfer distance 作比較，若兩次差異總和小於一定門檻值，則可判定該行人區塊為行人(圖 6.4)。不過這邊必須注意到如果使用魚眼型攝影機必須考慮到影像座標與場景座標之間的對應轉換才能樣版與行人影像進行比對。



圖 6.1、3D 人物模型

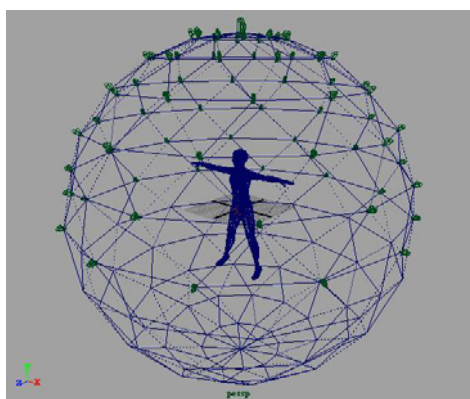


圖 6.2、多面體投影示意圖

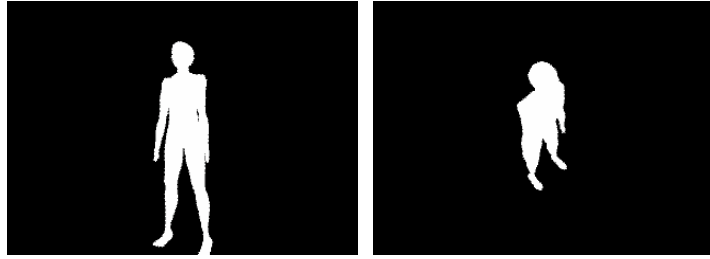


圖 6.3、所得到之二維投影(a)視角一(b)視角二

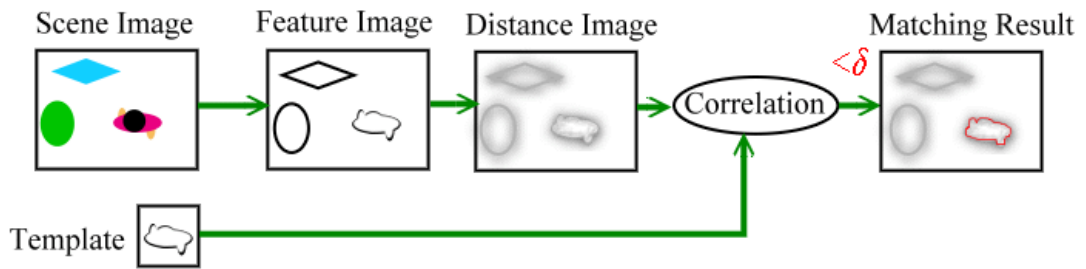


圖 6.4、Chamfer distance matching 示意圖

附錄 A：如何產生高斯亂數

如何取出具有高斯分佈亦或常態分佈(normal distribution)的亂數，一直是個值得探討的議題。其中 Dr. Everett F. Carter Jr.在 Taygeta 科學工業的網站提出了一個精確又快速的方法。Dr. Carter 提出的方法使用了 Box-Muller transform 的 polar form，以避免使用過多的數學函式(如：sine、cos)。其演算法的虛擬碼如下：

```
// Algorithm by Dr. Everett (Skip) Carter, Jr.  
float x1, x2, w, y1, y2;  
do {  
    x1 = 2.0 * randf() - 1.0; //-1 to +1  
    x2 = 2.0 * randf() - 1.0; //-1 to +1  
    w = x1 * x1 + x2 * x2;  
} while ( w >= 1.0 ); //see if it is in the unit circle  
w = sqrt( (-2.0 * ln( w )) / w );  
y1 = x1 * w;  
y2 = x2 * w;
```

其中 randf 為產生均勻分布(uniformly distribution)且數值為[0,1]的亂數的函式。這樣的演算法還有一個好處是會一次產生兩個 y1、y2 的期望值(mean)為 0、標準差(standard deviation)為 1 的常態分佈亂數，可大幅減少電腦運算時間。

附錄 B：Bhattacharyya 係數的定義

如圖 B.1 所示，令 \vec{q}' 為長度 1 的目標向量，而 $\vec{p}'(y)$ 為長度 1 的候選向量，則兩向量可分別表示為：

$$\vec{q}' = (\sqrt{q_1}, \dots, \sqrt{q_m})$$

$$\vec{p}'(y) = (\sqrt{p_1(y)}, \dots, \sqrt{p_m(y)}) ,$$

Bhattacharyya 係數則以兩向量之間的餘弦函數表示兩向量之間的相似程度：

$$\rho[p_u(y), q_u] = \cos \theta_y = \frac{p'(y)^T q'}{\|p'(y)\| \cdot \|q'\|} = \sum_{u=1}^m \sqrt{p_u(y) q_u} .$$

兩向量之間愈相似，則向量夾角 θ_y 愈接近零， $\cos \theta_y$ 則愈接近 1，Bhattacharyya 係數也就愈接近 1。反之，若兩向量之間愈不相似，則向量夾角 θ_y 愈接近 90 度， $\cos \theta_y$ 則愈接近 0，Bhattacharyya 係數也就愈接近 0。

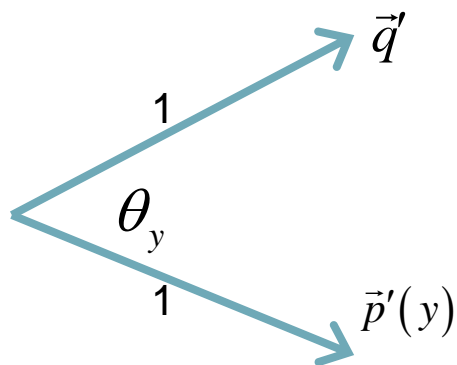


圖 B.1、Bhattacharyya 係數示意圖

附錄 C：多重視角投影環境建立

為了實作多重視角投影，我們使用正二十面體(Icosahedron)(圖 C.1)來作球面分割。首先我們建置了一個正二十面體，該正二十面體有 12 個頂點、20 個面、30 個邊，讓每個頂點到中心都是單位距離，每面都是正三角形。可定義正二十面體為：

$$t, \text{ the golden ratio} = \frac{1+\sqrt{5}}{2}$$

$$a = \frac{\sqrt{t}}{5^{1/4}}$$

$$b = \frac{1}{(\sqrt{t}5^{1/4})}$$

$$c = a + 2b = \frac{1}{b}$$

$$d = a + b = \frac{t^{3/4}}{5^{1/4}}$$

$$A = \text{angle subtended by edge at origin} = \arccos\left(\frac{\sqrt{5}}{5}\right)$$

然後這 12 個頂點的三維座標為：

$$(0, \pm a, \pm b)$$

$$(\pm b, 0, \pm a)$$

$$(\pm a, \pm b, 0)$$

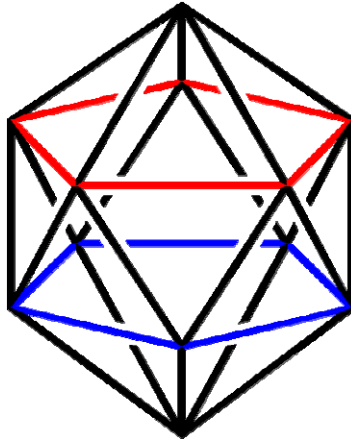


圖 C.1、正 20 面體

接下來我們針對正二十面體的正三角形面作子分割，以每個邊的中點為分割點，將邊分成兩個等長的邊，再使兩兩中點之間連線，如此一來，每個正三角面被分割成四個正三角面。重覆此步驟兩次(共三次)，此時的正二十面體變成了 320 面(即 $20*4*4*4$)、642 個頂點的多面體。之後，我們將每個新產生的頂點到中心距離作正規化，使其到中心距離為單位距離，正規下後結果如圖 C.2。雖然正規化後的每個面並不是正三角形，此方法但提供了一個頂點分佈較均勻的分割方式，這也是我們為何使用此方法作為球面分割的原因。

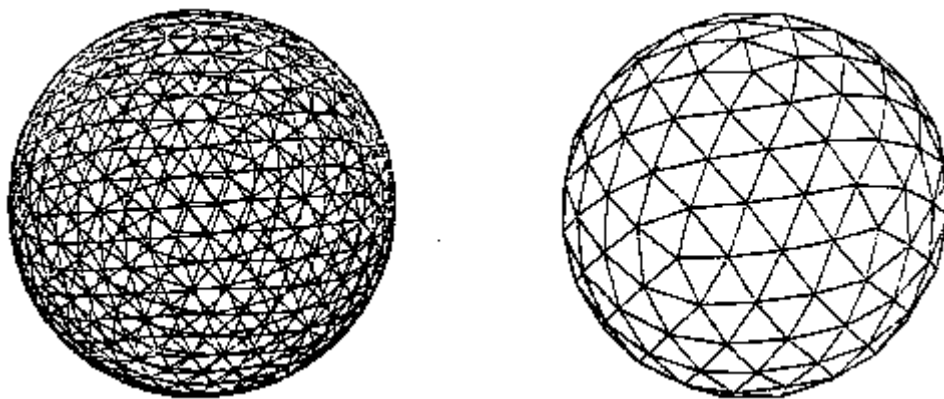


圖 C.2、球面分割後的多面體

我們將人物模型放入分割後的多面體中，並放在多面體上半部的頂點上放置

投影攝影機，鏡頭對準中間的人物模型，即可作二維投影。

參考文獻：

[Alb01] A. Albiol, I. Mora ,and V. Naranjo, “Real-time high density people counter using morphological tools,” IEEE Trans. on Intelligent Transportation Systems, Vol. 2 , No. 4 , pp.204 – 218, 2001.

[Bes03] J. Bescos, J.M. Menendez ,and N. Garcia, “DCT based segmentation applied to a scalable zenithal people counter,” Proc. International Conference on Image Processing, Vol. 3, pp. III - 1005-8 vol.2, 2003.

[Bro00] A. Broggi, M. Bertozzi, A. Fascioli ,and M. Sechi, "Shape-based pedestrian detection," Proc. the IEEE on Intelligent Vehicles Symposium IV, pp.215 - 220, 2000.

[Che03] Thou-Ho Chen, ” An Automatic Bi-Directional Passing-People Counting Method Based on Color Image Processing,” Proc. IEEE 2003 International Carnahan Conference on Security Technology , pp.200 – 207,2003.

[Che05] Maolin Chen, Gengyu Ma ,and Seokcheol Kee, ” Pixels Classification for Moving Object Extraction,” IEEE Workshop on Motion and Video Computing, WACV/MOTIONS '05, Vol. 2, pp.44 – 49 ,2005.

[Che06] Thou-Ho Chen, Tsong-Yi Chen ,and Zhi-Xian Chen, “An Intelligent People-Flow Counting Method for Passing Through a Gate,” IEEE Conference on Robotics, Automation and Mechatronics , pp.1 – 6, 2006.

[Che07]Ming-Hsu Cheng, Meng-Fen Ho ,and Chung-Lin Huang, “Gait Analysis for Human Identification through Manifold Learning and HMM,” IEEE International

Symposium on Circuits and Systems ,ISCAS 2007, pp. 969 – 972,2007.

[Cho06] Jung-Uk Cho, Seung-Hun Jin, Xuan-Dai Pham, Jae-Wook Jeon, Jong-Eun Byun ,and Hoon Kang, “A Real-Time Object Tracking System Using a Particle Filter,” 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems pp.2822 – 2827,2006.

[Dai00] D. J. Dailey, F. W. Cathey ,and S. Pumrin, "An Algorithm to Estimate Mean Traffic Speed Using Un-Calibrated Cameras," IEEE Trans. on Intelligent Transportation Systems, vol. 1, no. 2, pp. 98–107, 2000.

[Fou03] C.Fouard, "Automatic Calculation of Chamfer Mask Coefficients for Large Masks and Anisotropic Images," INRIA, pp.7-10 , 2003.

[Har99] I. Haritaoglu, D. Harwood ,and L.S. Davis, “Hydra: multiple people detection and tracking using silhouettes,” Proc. International Conference on Image Analysis and Processing, pp.280 – 285, 1999.

[Hua02] D. Huang, T.W.S. Chow ,and W.N. Chau, “Neural network based system for counting people,” IECON 02 [Industrial Electronics Society, IEEE 2002 28th Annual Conference] , Vol. 3 , pp. 2197 - 2201,2002.

[Hsi07]Jun-Wei Hsieh, Chuang-Yu Huang, Cheng-Shuang Peng ,and Kuo-Chin Fan “GRID-BASED TEMPLATE MATCHING FOR PEOPLE COUNTING,” Proc. of the 20th IPPR Conf. on CVGIP, Maui, Taiwan, 2007.

[Isa96] Michael Isard ,and Andrew Blake ”Contour tracking by stochastic propagation of conditional density,” Proc. European Conference on Computer Vision, vol. 1, pp. 343--356, 1996.

[Jae] Ae-Won Kim, Kang-Sun Choi, Byeong-Doo Choi ,and Sung-Jea Ko, “Real-time Vision-based People Counting System for the Security Door,” Department of Electronics Engineering, Korea University, Anam-dong, Sungbuk-ku, Seoul, pp. 136-701, Korea.

[Jam99] Sethian, and James “A Level Set Methods and Fast Marching Methods : Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science,” (2nd ed.). Cambridge University Press. ISBN 0-521-64557-3.4., 1999.

[Kom00] I. Kompatsiaris, G. Mantzaras ,and M.G. Strintzis, “Spatiotemporal segmentation and tracking of objects in color image sequences”, Proc. ISCAS 2000 Geneva. The 2000 IEEE International Symposium on Circuits and Systems, vol.5 , pp.29 - 32, 2000.

[Lai98] A.H.S. Lai ,and N.H.C Yung, “A fast and accurate scoreboard algorithm for estimating stationary backgrounds in an image sequence,” ISCAS '98. Proceedings of the 1998 IEEE International Symposium on Circuits and Systems, Vol. 4 , pp.241 - 244 , 1998.

[Lin01] G. Linda Shapiro ,and C. George Stockman, “Computer Vision” by Prentice Hall Chap.10.1.1 Clustering Methods ,pp.281-282,2001

[Mag07] E. Maggio, F. Smerladi, and A. Cavallaro, “Adaptive Multifeature Tracking in a Particle Filtering Framework,” *IEE Trans. On Circuits and Systems for Video Technology*, vol. 17, no. 10, pp. 1348-1359, 2007

[Mas01] Masoud, O., Papanikolopoulos, and N.P.”A novel method for tracking and counting pedestrians in real-time using a single camera,” *IEEE Transactions on Vehicular Technology*, vol. 50, no. 5, pp. 1267 – 1278, 2001.

[Ter99]K.Terada, ” A Method of Counting the Passing People by Using the Stereo Images,” *Proc. 1999 International Conference on Image Processing*, vol.2, pp.:338 – 342, 1999.

[Tan07] Yu-Gin Tang, Jiann-der Lee ,and Li-Chang Liu “AN EMBEDDED SYSTEM FOR VEHICLE TYPE RECOGNITION” *Proc. of the 20th IPPR Conf. on CVGIP*, Maui, Taiwan, 2007.

[Wan07] J. M. Wang, S. W. Chen, S. Cherng , and C. S. Fuh, “People Counting Using Fisheye Camera,” *Proceeding. of the 20th IPPR Conf. on CVGIP*, Maui, Taiwan, 2007.

[Xia95] Xiaowei Zhang ,and G. Sexton, “A new method for pedestrian counting,” *Fifth International Conference on Image Processing and its Applications* , pp. 208 – 212, 1995.

[Xia97] Xiaowei Zhang ,and G. Sexton, “Automatic human head location for pedestrian counting,” *Sixth International Conference on Image Processing and Its Applications*, vol.2, pp.535 – 540, 1997.

[Yan03] Yang, D.B., Gonzalez-Banos, H.H., Guibas, and L.J. “Counting people in crowds with a real-time network of simple image sensors,” Proc. Ninth IEEE International Conference on Computer Vision, vol.1, pp. 122 - 129 ,2003.

[Zha95] X. Zhang, and G. Sexton, “Automatic pedestrian counting using image processing techniques” Electronics Letters , vol. 31 , no. 11 , pp.863 – 865, 1995.

[陳 03] 陳重傑 “高層辦公建築火災避難安全評估之研究”中華大學 建築與都市計畫學系碩士班 國家圖書館全國博碩士論文 系統編號：92CHPI0224013,2003

[沈 93] 沈東石 “建築物內部人員交通系統緊急疏散效率之評估模型與應用”國立台灣大學 土木工程研究所 國家圖書館全國博碩士論文 系統編號：82NTU00015075,1993.

[張 05] 張厥煒、張傑閔 “運動視訊場景中動態物件搜尋與追蹤方法”國立台北科技大學資訊工程系 臺北科技大學學報第四十之一期,2005.

[鄭 05] 鄭皓盈 “使用Particle Filter之物體追蹤技術”工業技術研究院資訊與通訊研究所 第112期 電腦與通訊(CCL),2005.