

# Design and Implementation of a Portable People Counting System

*Chow-sing lin<sup>\*</sup>, you-hsuan li, ci-ruei jiang, ming-hsuan hu, and kai-min zeng*

*Department of Computer Science and Information Engineering*

*National University of Tainan, Taiwan*

*Abstract—People counting has been an important component in video surveillance systems. Video cameras are usually installed at fixed locations by professionals, and the recorded videos are processed to count the people flow either by human labors or image processing systems. As mobile technology advances, it is now possible to easily set up surveillance cameras at anywhere to get the information of people flow for a certain location by just using general mobile devices, which can be used to assess the feasibility of a business model, such as the opening of a chain store. In this paper we present the design of a portable people counting system which is implemented in an android-based mobile device. The system consists three subsystems which allow videos to be taken from three different angles: vertical, horizon, and inclination. Our experimental results show that the average accuracies of vertical, horizon, and inclination shots reach as high as 89%, 78%, and 72%, respectively.*

*Index Terms—People counting, Mobile Computing, Image Processing, Film Analysis, Android App*



# 移動式人流計數系統之開發與研究

林朝興\*、李侑軒、江啟睿、曾楷珉、胡銘軒

國立臺南大學資訊工程學系

## 摘要

以往在需要統計人流的時候，我們必須使用人工的方式，在一個定點統計所經過的人數，這樣不僅會因為人計算上的失誤，而導致誤差，更會耗費大量的人力成本。隨著科技的進步，出現許多自動化人流計數系統，並且發展出各自解決方案，但這些系統需要在固定定點架設，無法隨意的移動。且這些系統都需要由專業人員來進行架設與調整才能提供預期的計數效果。藉由現在行動裝置移動的方便性，提供利用行動裝置來進行人流計數，可以隨意的在任何地點使用，也不需專業人員來進行架設。我們的研究提出針對垂直、水平及傾斜角度所拍攝影片進行人流計數，並且針對行動裝置進行最佳化，使得人流計數在行動裝置上可以順利的被運用。最後經由實際測試我們在垂直拍攝方向影片上獲得約 89% 的準確率，水平拍攝角度則有 78% 的準確率，而傾斜拍攝角度則獲得 72% 的準確率。所以我們所提出的方法已經可以運行於行動裝置上，並且提供一定的準確率。

關鍵字—人流計數、行動計算、影像處理、影片分析、Android App

## 壹、簡介

在我們的生活中，人流計數系統 (People-counting systems) 被運用在許多地方，如：百貨公司的人流管制、商店的來客統計等等，涵蓋影片內容分析、建築物安全管理、商業的人潮分析等不同領域的應用。拜現代科技進步所賜，許多商業化的人流計數系統，皆可以提供良好的準確率，但在架設方式上卻需要專業人員與固定地點的限制，造成無法隨意的與任意地點進行計數。

在目前行動裝置可以說是人手一機的情況以及行動裝置運算速度的大幅提升下，利用行動裝置

來進行人流計數就變成了一個可行的方案。藉由行動裝置的移動性以及操作的便利性，可以在任何場所進行人流計數。

我們所提出的方法中，可以方便的在目前行動裝置上佈署，達到隨時隨地的使用。例如：商家在尋找店面時，可以利用智慧行手機拍攝人潮的影片，再利用此 APP 來計算出經過的人數，以減少傳統需要以人工方式計算。利用手機拍攝影片並且利用在手機上進行人流計算減少了人工計算的費時與誤差，以及提供優於傳統商業化計數系統的可移動性與便利性。

另外本實驗將採用 Android 做為開發平台，並且使用 Android Native Development Kit (NDK) 與 OpenCV 來開發計數演算法，其因使用 C++ 作為開發語言所以我們可以快速的移植到任何的平台也可以提升運算的速度。我們也會分別針對垂直、水平及傾斜拍攝角度分別提出各自的處理方法，並且使用 HTC New ONE 進行實際的測試。最後在垂直角度獲得 89% 準確率、水平角度獲得 78% 準確率及傾斜角度獲得 72% 準確率，證明我們所提出的方法是可以運行於行動裝置並且可以提供一定的準確率。

本篇論文編排如下，第二章節我們將針對目前現有的人流計數技術相關研究進行討論，分別提出各自的優點與缺點，並且與我們所提出的方法進行比較。第三章節則是會對我們提出的系統架構進行說明，分別對三種影片角度，垂直、水平及傾斜提出相對應的計數方法。第四章節是我們所進行的實際實驗結果。第五章節是我們目前系統的結論以及未來的展望。

## 貳、相關研究

在過去已經有許多關於人流計數方面的研究相繼的提出。並且在這些研究中，探討到人流計數所面臨的問題[1,2]。在[3]中，對現有的人流計數技術做出統整，並且根據使用不同感測器來做為分類

\* Corresponding author: mikelin@mail.nutn.edu.tw  
DOI: 10.6159/IJSE.2014.(4-3).03



依據。然而這些使用感測器的人流計數，如：光敏電阻(Photocells)、被動紅外線(Passive infrared, PIR)及微波(Microwave)，皆為透過阻斷的方式來計算人數，無法計算一群人走過的人數。

另一種進階的系統，需花費龐大的建置成本，如：Lee et al.[4]，利用兩組雷射(Laser)去計數人數。而另一種兼具效能以及成本的系統，虛擬視覺系統(Artificial vision systems)，降低人流計數的門檻，其根據所使用的攝影機數量而有所區別。由[5,6]提出使用兩個攝影機組成的立體視覺系統來進行人流計數，並且 Englebienne et al.[6]利用偵測感興趣區域(Region of interest, ROI)所出現的顏色以及概率模型來計算出人數。Fookes et al.[7]提出一個可以用在擁擠場景的計數系統，其並非真實的計算所有出現的人數，而是以一群人大概會包含幾個人來計算。在雙鏡頭系統中，在佈署上需要有更多的調整才能讓兩個鏡頭正常的運作，所以當建置完畢後，將無法隨意的變動，而沒辦法快速的運用在不同的場所。

Rizzon et al.[8]則是將攝影機架設高於行人頭部的地方，並且設定出兩個感興趣區域(ROI)，分別為影像的頂端與底端，計算兩個區域光流的直方圖，判斷是否超過所設定的門檻值來計算所經過的人數。

Barandiaran et al.[9]則是將攝影機架設於頂端由上而下的垂直拍攝方式，定義兩個水平的區域，並且計算通過這兩個區域的人數，並且依通過區域的順序來判斷人流方向。在此研究實驗中，得知陰影或行人穿越區域的速度都會造成計算的誤差。並且其無法解決當行人靜止時而被重複計算的產生。

Chen et al.[10]則是使用背景相減法(Background subtraction)，並且加入追蹤行人進入監控區域到離開，可以解決行人停止走動而被重複計算的問題。但是其無法適用於擁擠的場景。

Xu et al.[11]利用傾斜的攝影機，其系統須先辨識出行人的頭部，再進行監控區域內的追蹤。在行人頭部偵測上其利用 SVM 來做為分類器，在追蹤方面則是用 Kalman Filter，但是其並未在運作階段持續提供反饋訊息來修正預測追蹤頭部，所以無

法在行人發生碰撞重疊時，持續追蹤。

Garcia et al.[12]則是利用單一攝影機，並且拍攝角度設定為傾斜，此角度可以減少人與人之間互相遮蔽，或者人被一些物體遮蔽。其利用環行濾波器(Annular filter)快速的偵測行人頭部，並且避免因為空間或者碰撞所產生的遮蔽。接著將頭部偵測的結果交由 Kalman Filter 進行追蹤。

在本論文中我們將使用，單一攝影鏡頭，也就是行動裝置上內建的攝影鏡頭。並分別對三種可能的拍攝角度，垂直、水平及傾斜提出對應的計數方法，並且減少計算的複雜度以使其能於行動裝置上運行。

### 參、系統方法

在我們的方法中，我們將分別針對不同的影片拍攝角度，包含垂直拍攝角度、傾斜拍攝角度及水平拍攝角度，提出合適的演算法，以達到適用於多數場景與多人同時出現時的人流計數。並且考慮到行動裝置上的計算能力盡可能的簡化計算複雜度，使我們的方法可以順暢的運行於行動裝置上。

#### A. 垂直拍攝角度

圖 1 為垂直角度的拍攝示意圖，在此角度下所拍攝的影片，其人與人之間被遮蔽的機會相對較少，所以不需太多複雜的運算及可達到較高的準確率，並且頭部的樣板也較為簡單。

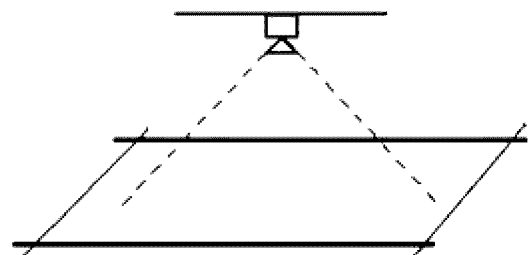


圖 1 垂直角度拍攝示意圖

圖 2 是我們在處理垂直拍攝角度影片的過程。我們將分別對每個處理程序進行說明。



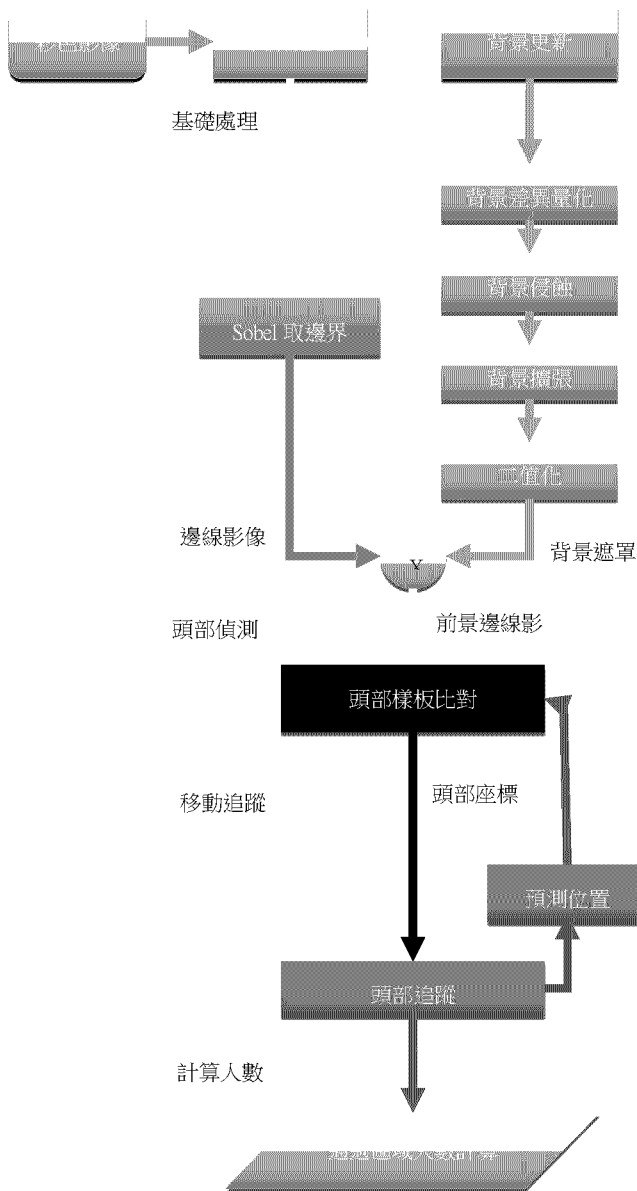


圖 2 垂直拍攝角度影片處理流程圖

我們將影片採用 Frame-by-Frame 的方式讀入。並且分別進行下列的處理：

### 1. 影像灰階處理

為了減少影像的處理量與不影響準確率的情況下，我們先將讀入的彩色影像進行灰階處理，以方便後續的處理。圖 3 將彩色影像進行灰階處理後所得到的結果。

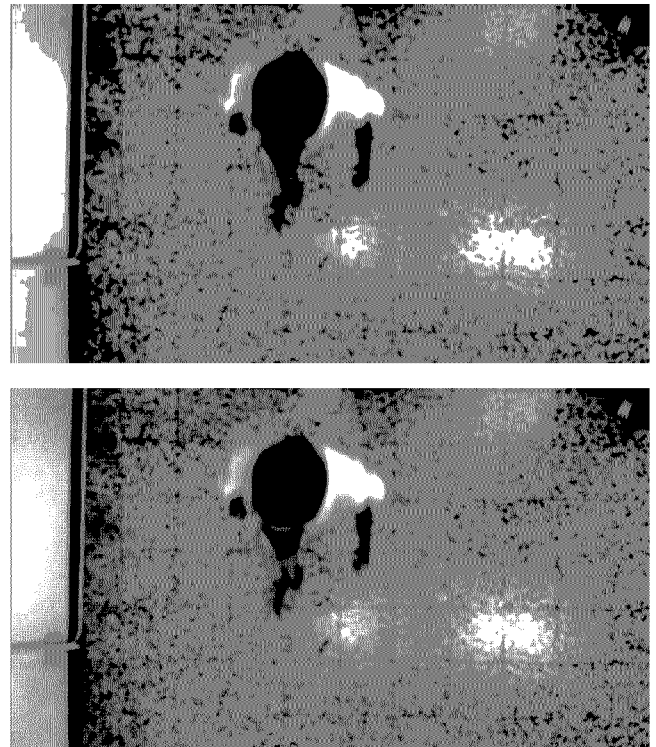


圖 3 彩色影像的灰階處理結果

### 2. 背景更新

我們將第一個影格的灰階影像作為整個影片的背景，並且以每 10 張影格作為一個更新單位。在一個更新單位內，我們會紀錄每個像素在此 10 張影格內的最大與最小值。並且計算同一個像素的最大與最小值差距，如果該差距小於門檻值(10)，就會將背景裡相同座標的像素值，而更新的值將根據下列方式更新

$$V_{m(x,y)} = \frac{[I_{\max(x,y)} + I_{\min(x,y)}]}{2} \quad (1)$$

$$\text{Back}_{\text{update}(x,y)} = \alpha \times \text{Back}(x,y) + (1 - \alpha) \times V_{m(x,y)}$$

- $I_{\max(x,y)}$  一個更新單位內，座標  $x, y$  的像素最大值
- $I_{\min(x,y)}$  一個更新單位內，座標  $x, y$  的像素最小值
- $V_{m(x,y)}$  一個更新單位內，座標  $x, y$  的像素平均值
- $\text{Back}_{\text{update}(x,y)}$  背景更新後座標  $x, y$  的像素值
- $\text{Back}(x, y)$  原本背景座標  $x, y$  的像素值

- f、 $I_n(x,y)$  值為原本背景像素值  $Back(x,y)$  於新背景所佔權重， $(1 - \alpha)$  值為更新單位內像素平均值  $V_m(x,y)$  於新背景所佔權重。預設值設為 0.3。

### 3. 背景差異量化

為了把當前灰階影像中的背景取出，我們需要將當前影像與背景影像進行差異量化的計算，其公式如下

$$I_{n(x,y)} = 255 \times \frac{(I_{diff(x,y)} - I_{diff_{min}})}{(I_{diff_{max}} - I_{diff_{min}})} \quad (2)$$

- a、 $I_{n(x,y)}$  表示座標  $x, y$  像素的差異量化值
- b、 $I_{diff(x,y)}$  代表當前灰階影像跟背景影像座標  $x, y$  像素的差異
- c、 $I_{diff_{max}}$  代表當前灰階影像跟背景影像座標  $x, y$  像素的差異最大值
- d、 $I_{diff_{min}}$  代表當前灰階影像跟背景影像座標  $x, y$  像素的差異最小值

### 4. 背景侵蝕

由於影片雜訊的問題，在進行背景差異量化時會產生不必要的雜訊，所以我們將進行侵蝕運算，來降低雜訊。

### 5. 背景擴張

在進行背景侵蝕後，可能會將一些完整的區域變成破碎，所以我們進行擴張運算來填補這些區域，使其得的背景影像可以更完整。

### 6. 二值化

為了降低整體背景影像的雜訊，我們設定一個門檻值(20)，來進行二值化。如果大於門檻值，像素變為 255，反之，則變成 0。圖 4 為最終背景遮罩的結果。



圖 4 背景遮罩二值化結果

### 7. Sobel 取邊界

另一方面，我們需要將灰階影像進行取邊界，我們採用 Sobel 的方法來進行。首先我們將灰階影像進行高斯模糊，以便後續取邊界時較為準確。接著針對水平與垂直方向進行 Sobel 取邊界，由於兩個不同方向所取得的邊界會有所不同，所以我們需要將兩方向邊界影像合併。圖 5 為進行 Sobel 取邊界處理後所得到的結果。

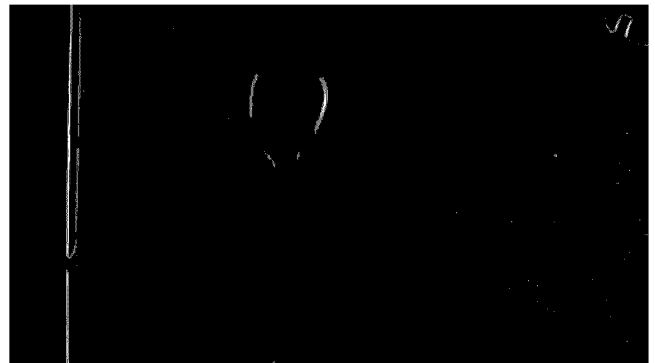


圖 5 Sobel 取邊線結果

### 8. 取得前景邊界圖

我們需要將目前影像的邊界圖，以及背景的遮罩，來取得前景的邊界圖。以方便後續頭部追蹤與人流計數。圖 6 為取得的邊界圖。

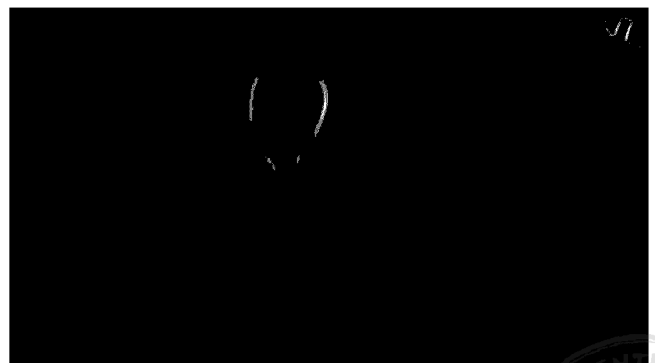


圖 6 前景邊界圖

## 9. 頭部樣板

在頭部樣板準備上，因為由上而下觀察人的頭部形狀，大都接近於圓形。但是在頭部與肩膀的分界有時可能會不夠清楚，所以我們將分為兩大類如圖 7，標準圓形，也就是最簡單的樣板。以及頭部加上肩膀的較複雜樣板。

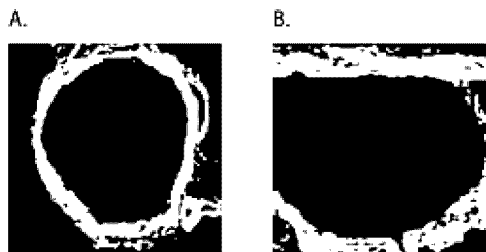


圖 7 (A)標準圓形 (B)頭部加上肩膀

## 10. 頭部追蹤

在這部分，我們在前景邊界影像裡，利用一系列的頭部樣板去做比對，找出其影像裡的頭部位置。由於使用樣板比對的方式有可能將不是頭部的影像也納入，所以我們將設定一個門檻值來過濾，若大於門檻值才確認為頭部，以減少誤判的情況。

## 11. 預測位置

為了加快頭部追蹤的速度，需要先進行下一個影格頭部可能的位置預測。我們以所偵測到的人頭座標，向右邊及下方各取 5 個點，共 25 個像素點。接著利用下方公式計算相似度來判別是否為同一點

$$r = \sqrt{\sum_1^{25} (c(x, y) - p(x, y))^2} \quad (3)$$

- a、  $c(x, y)$  目前所取的像素點
- b、  $p(x, y)$  前一張影格所取的像素點
- c、 當  $r$  越小時，則兩點相似度越高

## 12. 限制預測範圍

為了加快預測的速度，我們需要對預測的範圍進行限制。由於行人在移動時，兩張影格間的位置差異並不會太大。所以圖 3 是我們限制預測範圍的說明，我們將水平範圍僅擴大為樣板寬度的 1/2 大小，垂直範圍則增加樣板長度大小。圖 9 為預測範圍結果，紅色框線內為預測範圍。

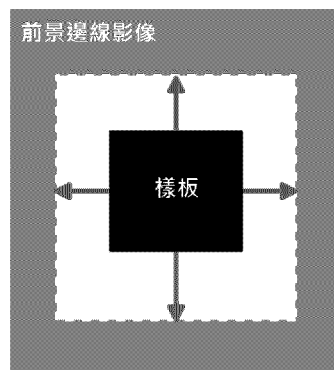


圖 8 限制預測範圍說明

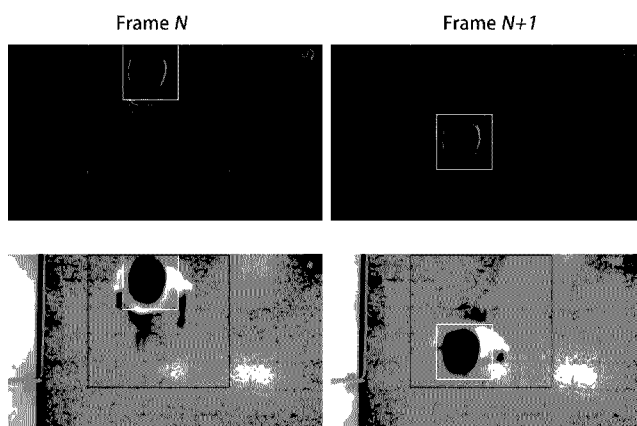


圖 9 預測範圍結果

## 13. 計算人數

當偵測行人頭部後，我們需要計算所出現在影像內的人數，其方法如下

- a、 設變數  $life$  為偵測到的點的生命週期，每一張圖片增加  $life$  次數 1 次
- b、 設變數  $detected\_life$  為偵測到的點又被追蹤次數，每追蹤一次  $detected\_life$  次數加 1 次
- c、 降低誤判
  - i. 當  $life - detected\_life > 門檻值(4)$  時，認定此點為誤判，清除此點資料
  - ii. 當座標過於接近時，以  $life$  大者為正確追蹤到的點， $life$  小者於以清除
  - iii. 設布林變數  $count$  為判別是否重複偵測
- d、 當  $count = false$  且 點通過中線時，再以前後座標判別方向，以計算人數



### B. 傾斜及水平拍攝角度

由於垂直角度在佈署上，需要架設高於行人的高度一定距離，造成在佈署上可能會有諸多的限制，如：要觀測的地方沒有地方可以架設垂直的攝影機，或者通道高度的限制，造成拍攝的影片沒有足夠的高度，而造成處理上的誤差。

所以我們另外提供了傾斜及水平拍攝角度的影片處理。圖 10 為水平角度拍攝示意圖，在水平角度上，可以更容易的在任一個場景佈署，不會受到場景的限制。但是在擁擠的人潮中，人與人互相遮蔽的問題就比其他角度來的嚴重。

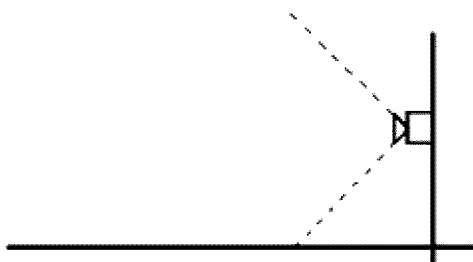


圖 10 水平角度拍攝示意圖

圖 11 為傾斜角度拍攝示意圖，所以在傾斜角度，就可以降低擁擠人潮所帶來的遮避問題，而最佳的傾斜拍攝角度位於 5 度[12]，在此拍攝角度內，人與人間的遮蔽較少。並且在佈署方面也沒有垂直角度來的麻煩。所以傾斜角度也會納入我們的方法中。

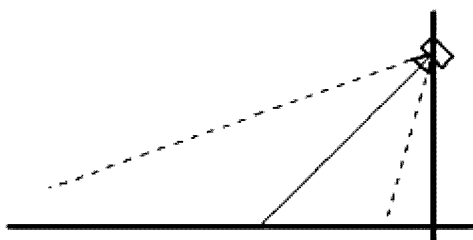


圖 11 傾斜角度拍攝示意圖

我們在處理傾斜角度及水平角度上所採用的流程相同，所以我們將一併在此說明。由於不論在水平及傾斜角度上，人與人之間的遮蔽會比垂直角度來的嚴重，並且在頭部的偵測上就會變得不容易。所以我們將不使用頭部偵測與追蹤，來進行計算，而是使用 Histogram of oriented gradients, HOG[13] 來做為主要的行人偵測，且 HOG 主要是以人體輪廓特徵作為比對，其偵測結果不受衣服紋理影響 [13]。

圖 12 是我們在處理水平及傾斜角度拍攝的影片時所採用之流程，我們將針對每個部分進行說明。

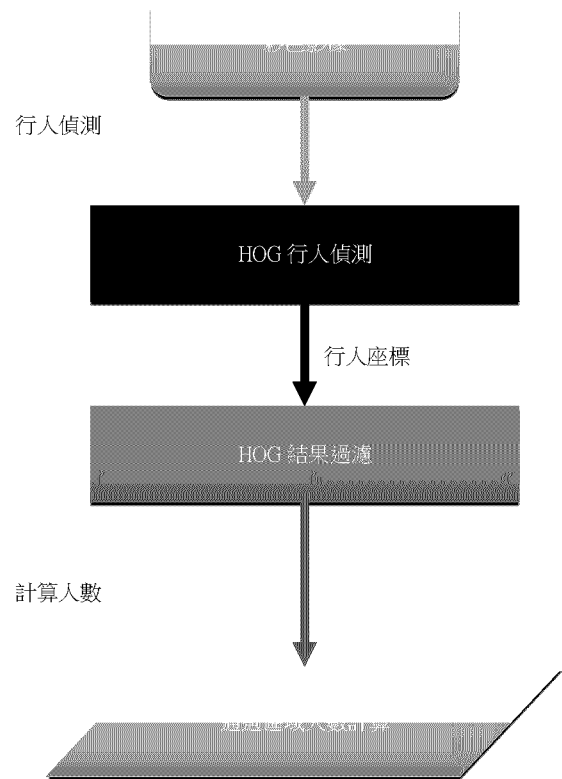


圖 12 水平及傾斜角度影片處理流程

#### 1. HOG 行人偵測

HOG 即是採用圖像的「局部梯度」當成圖像的特徵，並且利用 SVM 來學習圖片中人的特徵，來找到圖像中的人。

首先需要計算每個像素的梯度，利用  $[-1,0,1]$  及  $[-1,0,1]^T$ ，做摺積(convolution)。得到梯度後，便可以計算此像素的方向跟強度。

接著需要將影像進行單元(cell)的切割以方便後續直方圖(histogram)的計算。我們將圖像分成一格一格的單元，假設每個單元的大小為  $6 \times 6$  的像素，在將這 36 個像素依照他們的方向(alpha)利用統計得到一個值方圖。其 X 座標常見的是 4 個方向向量(4-orientation)或是 8 個方向向量(8-orientation)。

接著進行正規化結果，將單元分組在合併當成區塊(block)，而單元的直方圖也跟著合併，為代表此區塊的直方圖，並且正規化此直方圖。之後將所有已正規化的區塊直方圖串連成一個大的直方圖，此直方圖即為行人分析的特徵資料。最後利用已經



經由大量行人特徵資料訓練好的 SVM 來判斷，所取得的直方圖是否為行人，進而完成圖像中行人的偵測。

## 2. HOG 結果過濾

因為 HOG 會有誤判的情況，所以我們採用與垂直角度相同的前景取得方式，來得到當前影像的前景圖，判斷 HOG 所框出的範圍在前景圖中，是否有像素點(像素值大於 0)的存在。當其像素點數量大於門檻值(此門檻值為一經驗值，與影片解析度有關，以 630\*360 解析度影片為例，經驗值平均為 5000)，即代表其 HOG 所得到的物體是會移動的且大小與人接近，而其就有很大的機率為人。藉此改善 HOG 在行人偵測的誤判情況。

## 3. 人數統計

當有 HOG 所取得的行人結果後，還需要將取得的結果進行人數統計，我採用統計模型的方式來計進行。

假設 A 為記錄目前影像中出現過的總人數，B 為有偵測到行人的影格數，C 為每格出現行人之影格的平均人數，並且此三個變數皆初始化為 0。當 HOG 方法偵測到影像中出現人，則 A 加上影像中人的數量，B 則加 1。當 B 的數值達到行人通過畫面所需張數的門檻值(此門檻值為一經驗值，以 30 fps 鏡頭拍攝的影片為例，經驗值平均為 125 張圖，因 HOG 偵測上會有誤差，所以行人通過畫面的時間約為 4~6 秒)，則進行  $A/B$  並將值加入 C，並且將 A、B 歸零，C 即為目前的人流統計數量。當到達最後影格時，再做一次  $C = C + A / B$ ，C 即為整段影片的人流總數。

此方法在短時間的誤判上比較多，但在長時間的計算上，誤差隨著時間漸漸可被忽略掉。

### 肆、系統實驗

我們將實際開發一個 Android 平台上的 APP，並且提供三種不同角度影片的處理選擇：垂直、水平及傾斜。

我們將使用 OpenCV for Android 版本來進行圖像處理，並且在 APP 開發方式上採用 NDK 的方式，避免 JAVA 虛擬機的效率不彰，透過低層的程

式開發來加快整體效能。

圖 13 是本系統實際操作畫面，我們的測試平台為 HTC New ONE，其運算能力接近近期大部分智慧型手機的運算能力，所以我們以其為測試平台。

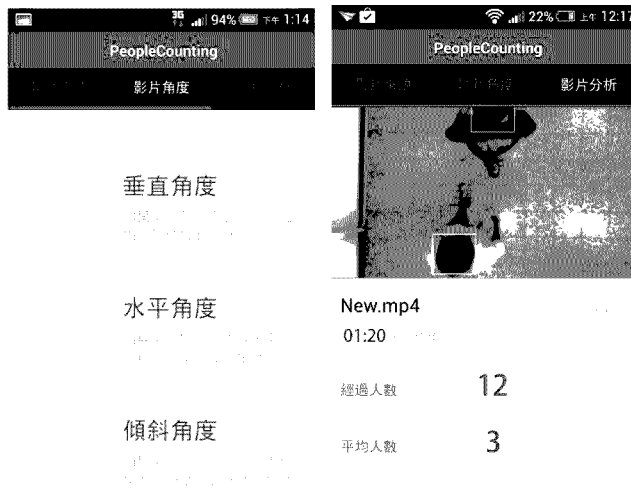


圖 13 APP 實際操作畫面

我們分別針對三種角度的影片進行測試。

### 1. 垂直角度影片

圖 14 是我們測試垂直角度的影片，在此影片中，我們使用一段 1 分鐘的影片，其拍攝幀數為 30fps，解析度為 640\*360，放置距離走道高度約 2 公尺的地方進行拍攝。

其總共經過人數為 54 人，測得人數為 48 人，準確率為 89%。其間同時經過的人數為 4，最少為 1 人。計算的平均速率為 2fps。

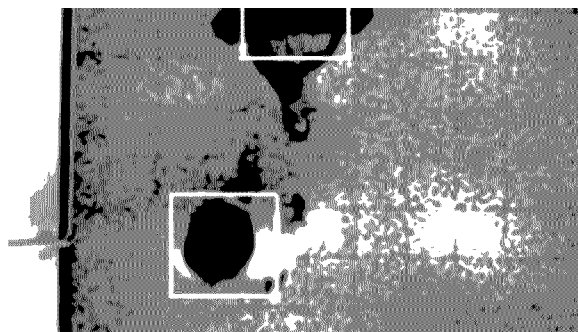


圖 14 垂直角度測試影片



## 2. 水平角度影片

圖 15 是我們測試水平角度的影片，在此影片中，我們使用一段 1 分鐘的影片，其拍攝幀數為 30fps，解析度為 640\*360，放置於高度為 160 公分的水平面上。

其總共經過人數為 15 人，測得人數為 19 人，準確率為 78%。其間同時經過的人數為 2，最少為 1 人。計算平均速率為 1fps。



圖 15 水平角度測試影片

## 3. 傾斜角度影片

圖 16 是我們測試傾斜角度的影片，在此影片中，我們使用一段 1 分鐘的影片，其拍攝幀數為 30fps，解析度為 480\*360，架設於天花板的攝影機，其攝影機拍攝角度為 30 度。

其總共經過人數為 25 人，實際測得 18 人，準確率 72%。其間同時經過人數為 3 人，最少為 1 人，並且出現停留的人次數為 1。計算的平均速率為 1.5fps。



圖 16 傾斜角度測試影片

表 1 各角度測試影片結果

項目	垂直	水平	傾斜
實際人數	54	15	25
偵測人數	48	19	18
準確率	89%	78%	72%

### 伍、結論

本人流計數的 APP 底層演算法是採用 C++ 來開發，所以未來可以快速地移植到各種平台，如：iOS、Windows Phone 等等行動平台，也可以移植到各種雲端平台來處理。由於 OpenCV 所提供的 HOG 行人辨識不夠準確，且其訓練資料大多以西方人為主，所以在亞洲人與西方人體型差異下，必須重新訓練，以提高行人辨識率。

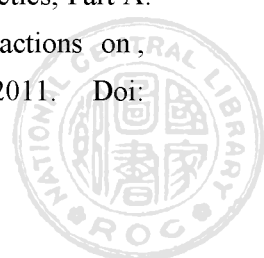
未來將持續改進人流計數演算法誤差，有效解決遮蔽問題，並且提升計算的效率使其能夠真正的達到即時的計數。並且將會利用雲端運算平行化行人偵測，加快人流計數計算的速度，以降低行動裝置的計算量以及減少耗電，盡而能提供更長時間的計數。

### 陸、致謝

感謝三商電腦股份有限公司 許顯達經理在本論文研究期間，給予諸多幫忙與意見，讓本研究可以順利進行並得到良好的實驗結果。

### 參考文獻

- [1] Chan, AB.; Vasconcelos, N., "Counting People With Low-Level Features and Bayesian Regression," Image Processing, IEEE Transactions on, vol.21, no.4, pp.2160,2177, April 2012. Doi: 10.1109/TIP.2011.2172800
- [2] Ya-Li Hou; Pang, G.K.-H., "People Counting and Human Detection in a Challenging Situation," Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on, vol.41, no.1, pp.24,33, Jan. 2011. Doi:



- 10.1109/TSMCA.2010.2064299
- [3] Velipasalar, S.; Ying-Li Tian; Hampapur, A., "Automatic Counting of Interacting People by using a Single Uncalibrated Camera," *Multimedia and Expo, 2006 IEEE International Conference on*, vol., no., pp.1265,1268, 9-12 July 2006. Doi: 10.1109/ICME.2006.262768
- [4] Jae Hoon Lee; Yong-Shik Kim; Bong Keun Kim; Ohba, K.; Kawata, H.; Ohya, A; Yuta, S., "Security Door System Using Human Tracking Method with Laser Range Finders," *Mechatronics and Automation, 2007. ICMA 2007. International Conference on*, vol., no., pp.2060,2065, 5-8 Aug. 2007.
- [5] Cottini, N.; De Nicola, M.; Gottardi, M.; Manduchi, R., "A low-power stereo vision system based on a custom CMOS imager with positional data coding," *Ph.D. Research in Microelectronics and Electronics (PRIME), 2011 7th Conference on*, vol., no., pp.161,164, 3-7 July 2011. Doi: 10.1109/PRIME.2011.5966242
- [6] Englebienne, G.; van Oosterhout, T.; Krose, B., "Tracking in sparse multi-camera setups using stereo vision," *Distributed Smart Cameras, 2009. ICDS 2009. Third ACM/IEEE International Conference on*, vol., no., pp.1,6, Aug. 30 2009-Sept. 2 2009. Doi : 10.1109/ICDS.2009.5289371
- [7] Fookes, C.; Denman, S.; Lakemond, R.; Ryan, D.; Sridharan, S.; Piccardi, M., "Semi-supervised intelligent surveillance system for secure environments," *Industrial Electronics (ISIE), 2010 IEEE International Symposium on*, vol., no., pp.2815,2820, 4-7 July 2010. Doi: 10.1109/ISIE.2010.5636922
- [8] Rizzon, L.; Massari, N.; Gottardi, M.; Gasparini, L., "A low-power people counting system based on a," *Circuits and Systems, 2009. ISCAS 2009. IEEE International Symposium on*, vol., no., pp.786,786, 24-27 May 2009. Doi: 10.1109/ISCAS.2009.5117869
- [9] Barandiaran, J.; Murguia, B.; Boto, F., "Real-Time People Counting Using Multiple Lines," *Image Analysis for Multimedia Interactive Services, 2008. WIAMIS '08. Ninth International Workshop on*, vol., no., pp.159,162, 7-9 May 2008. Doi: 10.1109/WIAMIS.2008.27
- [10] Thou-Ho Chen; Tsong-Yi Chen; Zhi-Xian Chen, "An Intelligent People-Flow Counting Method for Passing Through a Gate," *Robotics, Automation and Mechatronics, 2006 IEEE Conference on*, vol., no., pp.1,6, 1-3 June 2006. Doi: 10.1109/RAMECH.2006.252623
- [11] Huazhong Xu; Pei Lv; Lei Meng, "A people counting system based on head-shoulder detection and tracking in surveillance video," *Computer Design and Applications (ICDA), 2010 International Conference on*, vol.1, no., pp.V1-394,V1-398, 25-27 June 2010. Doi: 10.1109/ICDA.2010.5540833
- [12] Garcia, J.; Gardel, A; Bravo, I; Lazaro, J.L.; Martinez, M.; Rodriguez, D., "Directional People Counter Based on Head Tracking," *Industrial Electronics, IEEE Transactions on*, vol.60, no.9, pp.3991,4000, Sept. 2013. Doi: 10.1109/TIE.2012.2206330
- [13] Dalal, N.; Triggs, B., "Histograms of oriented gradients for human detection," *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol.1, no., pp.886,893 vol. 1, 25-25 June 2005. Doi: 10.1109/CVPR.2005.177

#### 作者簡介

林朝興 教授於西元2000年取得美國中佛羅里達州立大學電腦工程博士，目前任職於國立臺南大學資訊工程學系，研究興趣為網路媒體傳輸、同儕式媒體串流、分散式多媒體系統、行動計算、及媒體內容分析。

