

基於立體視覺之人流計數演算法

The Algorithm of Human Flow Counting Based on Stereo Vision

王韋捷¹、陳首次²

Wei-Jie Wang¹, Shou-Cih Chen²

¹ 國立台灣科技大學自動化及控制研究所

² 空軍航空技術學院管理系

¹ Graduate Institute of Automation and Control,

National Taiwan University of Science and Technology

² Department of Management, Air Force Institute of Technology

摘要

在人流計數上，以往的方式通常須以人臉辨識或者背景相減法來達成。但是，人臉辨識只能在人臉朝向攝影機時才能偵測出來；而背景相減法雖然可以解決前者的問題，但對於交疊的人群，則無法被個別區分開來。故本文提出基於立體視覺的人流計數演算法，透過雙鏡頭的架構來獲得視差資訊，不但可以不需透過偵測人臉完成計數，也能利用視差資訊來將交疊的行人個別偵測出來。此外，對於立體視覺計數人流而言，大部分的方法都是以獲得整張影像的視差資訊，即視差圖來完成。而本文則是只透過垂直邊緣的視差來達成，不僅節省在區塊比對上所花費的時間，實驗結果亦顯示出本文提出的人流計數演算法能夠有效計數出正確的人數。

關鍵詞：立體視覺、立體匹配、區塊比對、人流計數、垂直邊緣偵測

Abstract

In the past, the face recognition or background subtraction method is usually used to achieve the human flow counting. However, face recognition can only be detected when the face is facing the camera. Although the background subtraction method can solve the former problem, it cannot be distinguished individually for overlapping groups of people. Therefore, this paper proposes a human flow counting algorithm based on stereo vision to obtain disparity information through a dual-lens architecture. Not only can the count be completed without detecting faces, but also the disparity information can be used to individually detect overlapping people. In addition, for the use of stereo vision to count the flow of people, most of the methods are to obtain the disparity information of the entire image, that is, the disparity map. This article is only achieved through the disparity of the vertical edge, which not only saves the time spent on block comparison, but the experimental results also show that the proposed algorithm can effectively count the correct number of people.

Key Words: Stereo vision, block matching, human flow counting, vertical edge detection.

一、前言

隨著科技的快速發展，人工智慧的興起，越來越多的產品透過攝影鏡頭擷取目標物件的影像資訊後運用影像處理的技術來達成物件偵測及辨識的目的，如無人車自動駕駛功能，其藉由影像辨識技術來達成交通號誌或前方行人的偵測。除此之外，監控在影像處理中的應用也相當廣泛，且發展具有智慧的監控技術更是現在及未來的發展重點。監控主要目的在於安全管控及資料蒐集，因此，藉由監控系統來達成自動人流的計數及目標環境的監控顯然非常重要，像圖書館、大樓及無人商店等場合可透過自動計數人流的進出，來掌握每日進出的人數及場所環境的監控。另一方面，當應用於商場走道及廣告看板中，可以藉由觀察大量的人群進一步提高客群的分析作業。

在人流計數的方法中，透過人臉偵測目標數量是最直接的方式，因為，臉部是全身最具有特徵的部位。Chen 等人[1]即是針對人臉進行偵測，首先轉換 RGB 色域至其他色域之膚色偵測，再透過眼睛有較高的對比特性去定位出臉部位置。除此之外，特徵抽取是較為常見的方式，利用抽取不同臉部特徵的方式如 Local Binary Patterns (LBP)[2]或 Histogram of Oriented Gradient (HOG)[3]，接著搭配分類器如 Support Vector Machine (SVM)[4]來辨識人臉以達成計數人流的目的。由於行人於出入口通行時不會只以單一的方向進入或離開，故無論是正對或背對著攝影機，行人都屬於是正在移動中的物體，因此，透過移動中的物體進行偵測，也能夠達到人流計數的目的。所以 Q. Ye[5]提出運用背景相減法來執行人流計數，其利用前景物體出現時與背景不同顏色的概念，將有行人出現的前景影像減去無行人出現的背景影像，得到二值化影像後，接著進行型態學膨脹及收縮讓前景更加完整，再將該影像

以連通物件演算法標記所有前景物件，最終篩選出面積較大的物件後就可以得到物件的數量。然而背景相減法雖然能偵測出正面或背對的人，但是對於影像中交疊的人或物體，卻無法有效分出個體。

反之，立體視覺能夠以不同距離判斷出交疊的人群，而且不須先行透過偵測人臉即能完成人流計數之目的。而立體視覺是採用雙鏡頭的硬體架構取代單鏡頭，優點在於多了三維空間資訊，其主要是利用左右影像所獲得的視差圖及影像處理技術來偵測出行人的數量。如 N. Bernini[6]等人模擬在公車上計數乘客數量的方法，首先須先獲得整張視差圖後搭配其他邊緣與角點資訊偵測出影像中乘客的數量。而 T. Darrell 等人[7]則是結合整張視差圖、臉部特徵及膚色偵測來追蹤行人。上述所說的方法都是以獲得整張視差圖為主，且視差解析度不高，與只對垂直邊緣的視差圖而言，需耗費更多的比對時間。

常見的立體視覺比對方法分為兩種，一種為全域式比對方法，另一種則是區域式。全域式主要目的在於使能量函數最小化，能量函數分為資料項及平滑項，資料項即是加總每個像素比對所花費的代價，平滑項則是決定視差的平滑程度，鄰近像素其視差會呈現平滑的狀態，不會突然有很大的變化。因此在平滑項中會設定懲罰參數，若鄰近的視差變化愈大，則平滑項值愈大，更遠離使能量函數最小化的目標。但全域式所耗費的時間較多，常見的方法有 Graph Cuts[8]及 Dynamic Programming[9]。而較為被大家所應用的則是區域式的比對方法，即區塊比對。其目的是找出兩影像間相同的區塊，而做法係指用一固定大小的視窗在左右影像中計算區塊間的差異，稱為代價，最後在搜尋範圍中找出最小代價。除了最常見的 Sum of Absolute Difference (SAD)、Sum of Squared Difference (SSD)與 Normal Cross-Correlation

(NCC)外，Zabih and Woodfill 也提出 Census Transform[10]，其方法必須先透過鄰近像素與中心像素比較大小後轉換成二元碼，再計算兩字串間的漢明距離 (Hamming distance)，也就是加總兩字串間不同數值的個數作為代價，R.-K. Gupta 與 S.-Y. Cho[11]提出比對到不相似的區塊應該給予更重的懲罰，即提高代價讓相似與不相似的區塊拉開差距以確保有一個穩定的最小值，與 SSD 的概念有點類似，K. Zhang 等人[12]提出以十字為基礎，向外延展成不固定大小及形狀的視窗，以積分影像的方式加快比對速度，相對於較細的物體，能夠保有較精確的視差輪廓，但是目前大多數的區域式演算法，對於較大的平滑區域，無法得到很好的結果。而其他的方法像 H. Hirschmüller[13]則是提出以區域式的比對及兩個懲罰係數組成半全域的概念獲得視差圖，OpenCV 也將這個方法寫成函式提供使用[14]，C. C. Pham 及 J. W. Jeon[15]針對其中一張影像進行邊緣維持的平滑濾波，加上原始兩張影像的比對結果來決定視差，其視差圖在視差不連續的情況下會保存較完整的邊緣。

綜上所述，為了減少比對點數所花費的時間及降低在平滑區域容易比對錯誤的問題，在本研究中，我們提出只對垂直邊緣，也就是水平梯度變化較大的像素點，針對這些點去做區塊比對，這些像素點遠比平滑區域更能夠得到精確的視差值。

二、系統架構

本節首先進行本演算法的流程說明，接著說明如何運用軟、硬體設備擷取雙鏡頭所拍攝的影像及如何應用演算法來獲取三維資訊。

2-1 系統流程

演算法的流程圖如圖 2.1 所示，由於本文係以右影像的區塊對應左影像的區塊進行搜

尋，故須先行將擷取卡擷取出來的左右影像進行校正。接著，運用本文所提的改良 SAD 方法對左右影像的垂直邊緣點進行區塊比對以獲得視差資訊。在獲得視差資訊後，接著以兩階段方式來濾除雜訊，第一階段為濾除背景雜訊，即左右影像比對視差值過小視為背景雜訊濾除，第二階段利用連通物件方法對垂直邊緣點進行連通，並將長度較短的物件視為雜訊濾除。雜訊濾除後，將整張影像的邊緣點做直方圖統計並從中尋找波峰，當找到波峰後，運用最小平方法對每個波峰作高斯分佈函數的曲線擬合，並藉由所獲得波峰之標準差及平均值來自動決定視差範圍及分出不同的視差平面。最後利用投影的方式改善在視差平面對行人定位不合理的方框。

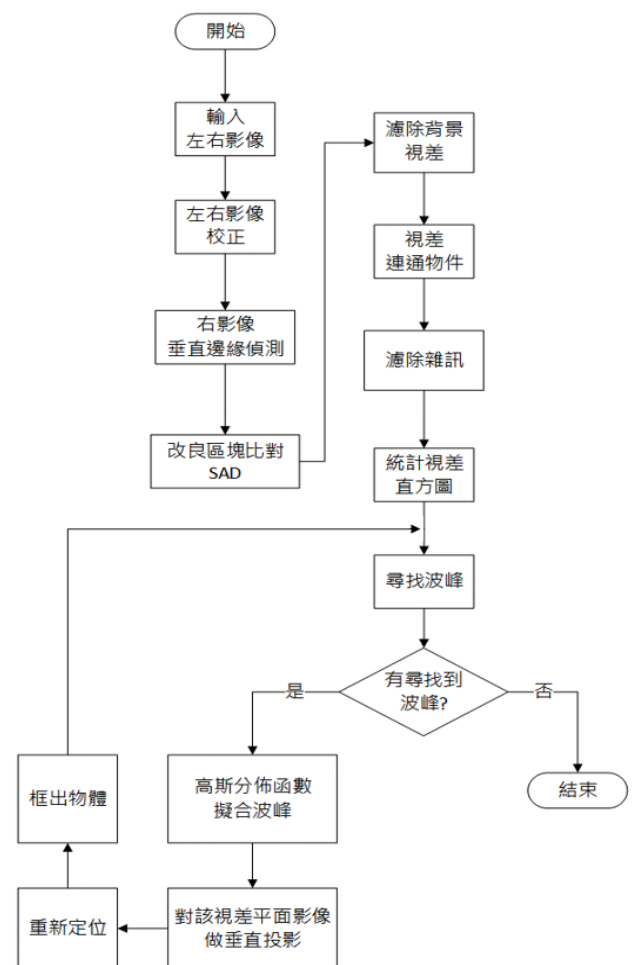


圖 2.1 完整演算法流程圖

2-2 軟硬體設備介紹

本文所需軟硬體說明如下，影像擷取卡採用凌華® PCIe-2602 3G-SDI[16]，如圖 2.2 所示，官網提供 SDK 範例程式，可改寫雙鏡頭影像擷取功能。而攝影機則是友旭科技®的產品，如圖 2.3 所示，形號為 15-CAH22，其攝影機規格如圖 2.4 所示，其提供兩種影像解析度，本研究左右影像的影像解析度皆為 1920x1080，鏡頭則搭配 6mm 的固定式焦距，如圖 2.5 所示。

軟體使用 Microsoft® Visual Studio 2015 裡的 MFC 視窗介面，搭配 C/C++ 程式語言及 OpenCV 函式庫完成演算法之撰寫。OpenCV 提供許多影像處理常用的演算法，如邊緣偵測、形態學膨脹及侵蝕、影像校正等等。



圖 2.2 影像擷取卡



圖 2.3 攝影機

SPECIFICATION

Model	15-CAH22
影像感知器	1/3" Panasonic CMOS
總畫素	2010(H) x 1108(V) = 2,227,080 (pixel)
有效畫素	1944(H) x 1092(V) = 2,122,848 (pixel)
鏡頭座	CS Mount
畫素	1920x1080p (1080p/30fps), 1280x720p (720p/60fps)
模擬水平掃描畫素	700TV Line
最低照度	彩色: 1.0 lux, 黑白: 0.5lux 彩色增累積: 0.002 lux, 黑白增累積: 0.001 lux
影像輸出	數位: HD-SDI / 類比: NTSC/ PAL (WDR 關閉)
S/N信噪比	50dB (關閉自動增益控制)
亮度	0 ~ 20階級
快門速度	自動/手動(1/30~1/60,000)
數位慢速電子快門	關/ x2/ x3/ x4
自動增益控制	關/開
寬動態/背光補償	關/寬動態/背光補償
夜間照明	自動/白天/夜間模式/ Ext

圖 2.4 攝影機規格表[17]



圖 2.5 固定焦距鏡頭

2-3 立體視覺比對

立體視覺是模擬人類雙眼造成的視差來獲得三維資訊，其概念像在看 3D 電影。距離的遠近與視差成反比，距離愈遠視差愈小；相反地，距離愈近視差愈大。舉個生活中的例子比較好理解，如果在火車上，從窗外看出去會發現較近的物體會快速移動過去，而較遠的物體則會移動較慢。圖 2.6 可以看到雙鏡頭拍攝物體 B 之示意圖，f 為焦距，Baseline 為兩攝影機間的距離。其中 OL 與 OR 為雙攝影機的位置，通過左右影像平面 b 與 b'到 B 點，可以在圖中看到立體視覺的特性，左影像中的 b 點會比右影像的 b'點還要偏右側，其之間的差距

就是視差值，即 $XL - XR$ 。圖 2.7 為左右影像比較示意圖，圖 2.7(a) 為左影像，圖 2.7(b) 為右影像，若以右影像為主，同一點在左影像其座標會往左位移，距離攝影機較近的物體其左影像的位移量會比距離攝影機較遠物體的位移量較大，也就是前面所提的較近的物體其視差值較大，反之則較小。

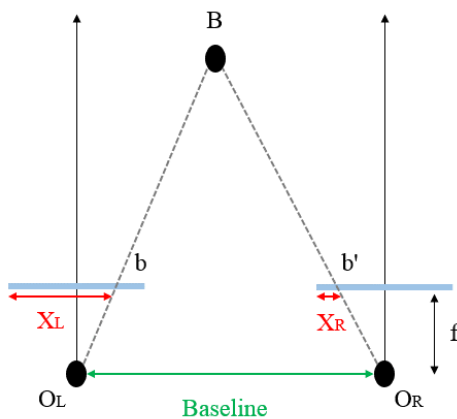


圖 2.6 立體視覺示意圖[16]



(a)



(b)

圖 2.7 左右影像之比較 (a)左影像 (b)右影像

而獲得視差圖主要有四個步驟，如圖 2.8

所示。首先，以 SAD 區塊比對為例，第一步就是將兩個區塊內每個像素強度值相減加上絕對值，再將所有像素的強度差異總和得到一個代價值，完成第二步聚合代價，公式如式 2.1 所示，其中 x 與 y 為影像座標， u 與 v 為水平及垂直搜尋範圍， W 為視窗大小的一半， I 則是灰階強度值，介於 0 到 255。

$$SAD(x, y, u, v) = \sum_{dx=-W}^W \sum_{dy=-W}^W \left| I_L(x+dx, y+dy) - I_R(x+u+dx, y+v+dy) \right| \quad (2.1)$$

而視差計算，通常以 Winner Take All (WTA) 的方式決定視差，也就是在搜尋範圍中，該點的區塊代價為最小值，就當該點當作視差。最後視差精細的部分，最常見的方式是以 Left-Right Consistency Check (LRC) 來找出遮蔽的像素點。當左影像到右影像作區塊比對時，找到的點同樣地到另一張影像重新作一次區塊比對，找到的點必須為同一個點，原則上才能確定該點視差是否正確，而被剔除的像素再以內插補點的方式讓視差圖更加準確。

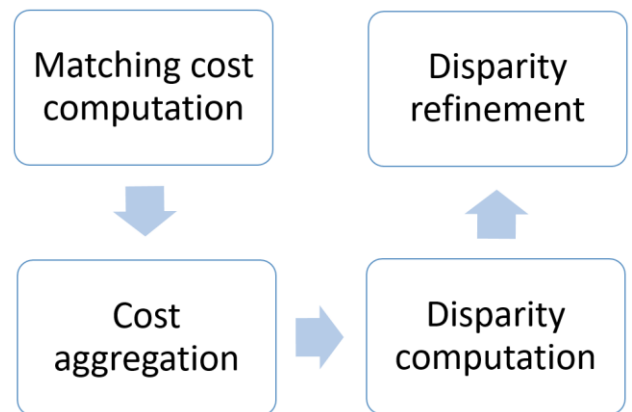


圖 2.8 立體視覺視差比對流程圖[17]

三、影像前處理

為使左右影像能偵測出較為精準的視差資訊，首先，須對影像進行前處理，本節針對兩部分進行解說，第一部分為影像校正，用以

改善影像邊界部分，第二部分詳細說明為何要以垂直邊緣進行視差比對，以及垂直邊緣偵測後的結果。

3-1 影像校正

影像形變分為徑向形變及切向形變[18]，徑向形變是由光進入到鏡頭時而產生的，切向形變則是因為相機組裝感光元件所導致的誤差，而徑向形變又分為枕狀形變及桶狀形變，如圖 3.1 所示，中間是沒有失真的情況，左邊是枕狀形變，右邊則是桶狀形變。

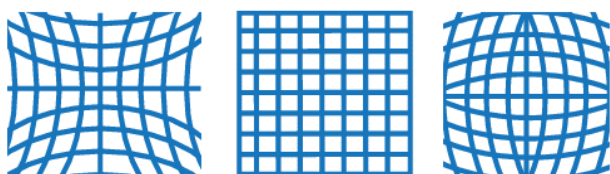


圖 3.1 形變示意圖[19]

我們利用 OpenCV 函式庫作影像校正，其方法來自於張友正所提出一種較為簡單但卻非常有效的方法[20]，而攝影機因為原廠組裝而導致內部參數的不一樣，因此，我們會各別對兩台攝影機進行校正。圖 3.2 為校正流程，首先，拍攝多張校正板，由於影像在邊界部分會出現較嚴重的失真，因此為了讓校正後的影像較無失真，校正板會多放置於影像的邊界，且需要不同角度，如圖 3.3 所示。接著程式會去偵測每一張校正板的角點，該校正板角點一共有 24 個，之後角點會進行連線，如圖 3.4 所示。

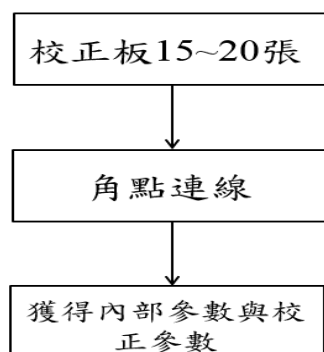


圖 3.2 校正流程

最後會得到相機的內部參數及校正參數。以這兩個參數對原始影像的每個像素做校正，圖 3.5 是實際拍攝的影像在校正前後的比較，圖 3.5(a)為校正前的影像，可以看到邊界有點桶狀形變，但在圖 3.5(b)的校正後，會發現影像邊界有明顯的改善。

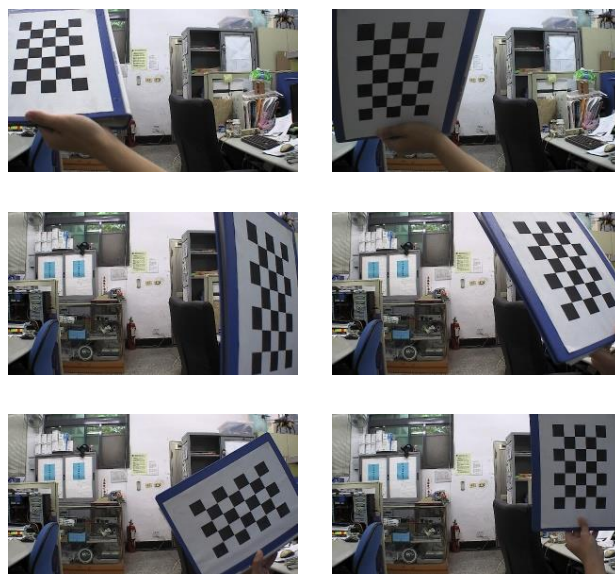


圖 3.3 多張角度校正板

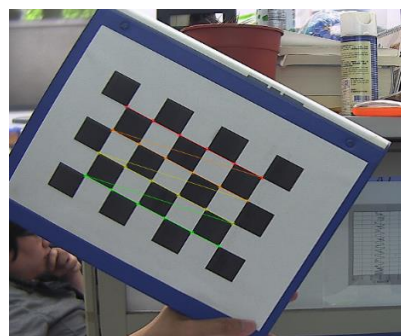


圖 3.4 校正板角點偵測結果



(a)



(b)

圖 3.5 校正前後比較 (a)校正前 (b)校正後

3-2 垂直邊緣偵測

邊緣在影像中是為高頻，與周圍的像素呈現出高對比，比起平滑區域的單一色彩，邊緣具有豐富的色彩資訊[21]，也因此會在區塊比對中獲得較好的結果，像 S. Birchfield 與 C. Tomasi[22]也提到邊緣點確實能夠獲得較正確的視差。然而，邊緣卻有分水平邊緣及垂直邊緣，由於雙鏡頭的擺放方式與人眼一樣，視差只有水平方向，故立體視覺主要以水平搜尋的方式獲得視差，也因此垂直邊緣的視差會比水平邊緣來得準確。圖 3.6 舉例水平及垂直邊緣在做 SAD 區塊比對的比較，最上方的小區塊表示在邊緣點的區塊，中間的區塊表示搜尋範圍，下方則是代價-搜尋範圍關係圖，可以看到左邊的曲線圖，完全呈現出一直線，表示無法有效找出相同點，與平滑區域所呈現的是一樣的曲線關係圖。

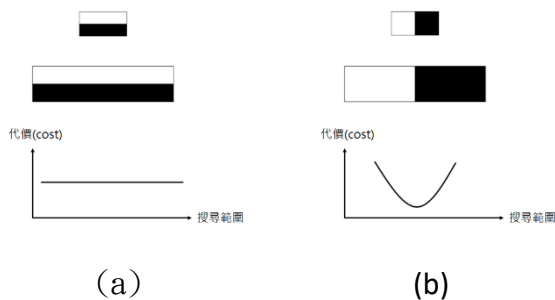


圖 3.6 代價-搜尋範圍關係示意圖 (a) 水平邊緣 (b) 垂直邊緣

而右邊的曲線圖則是呈現上拋的曲線，可

以明顯看出在曲線中心點為最小代價，即比對到相似度最高的像素點，因此由圖 3.6 的比較可以發現垂直邊緣點對於視差的比對準確許多。當左右影像皆校正完成後，本文以右影像獲得像素的視差為主，並採用 3×3 遮罩對影像進行 Sobel 垂直邊緣偵測。式(3.1)則是判斷該像素是否為垂直邊緣點， x 及 y 為座標點， $G_y(x,y)$ 為水平梯度值，門檻值固定為 50。水平梯度大於門檻值的為垂直邊緣，即 $p(x,y)=1$ ，反之則不是。

$$p(x,y) = \begin{cases} 1, & G_y(x,y) > 50 \\ 0, & G_y(x,y) \leq 50 \end{cases} \quad (3.1)$$

四、人流計數演算法

在完成影像前處理後，接下來說明如何運用改良的 SAD 區塊比對方法尋找影像視差資訊及如何運用連通物件方法濾除雜訊，接著統計邊緣點的視差直方圖，運用最小平方方法將直方圖中的各個波峰做高斯分佈函數的曲線擬合，自動決定出每個波峰的視差範圍以切出不同的視差平面，最後，則是透過垂直投影邊點的方式來確定行人在影像中的位置及數量。

4-1 改良 SAD 區塊比對

圖 4.1 為本研究所改良的 SAD 區塊比對流程圖，主要針對重複紋理區域容易比對錯誤的問題進行改善。流程步驟與傳統 SAD 方法的差異主要在找尋最小代價後會加上一判斷式用以濾除代價大於 10000 的邊緣點，主要原因在於所有搜尋範圍內找到的最小代價數值還是偏高，表示該區塊極有可能在另一張影像中找不到相同的區塊，這種現象稱為遮蔽效應。因此，透過簡單的門檻值設定將這些大於門檻值的邊緣點濾除。

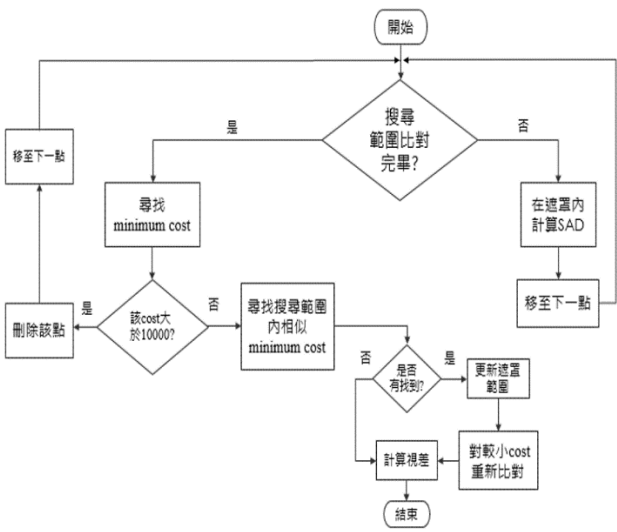


圖 4.1 改良 SAD 區塊比對流程圖

接著使用 3 組立體視覺資料庫 Middlebury[23]中標準影像 Reindeer、Books 及 Flowerpots(如圖 4.2)來測試視差準確率，並以常見的區塊比對演算法 SAD、SSD、NCC 及本研究所提出的改良 SAD 區塊比對方法，針對垂直邊緣的像素做比對，其垂直邊緣的門檻值設定在 50，可以看出來本研究所提的方法大致上都優於其他三種方法，如表 4.1 所示。

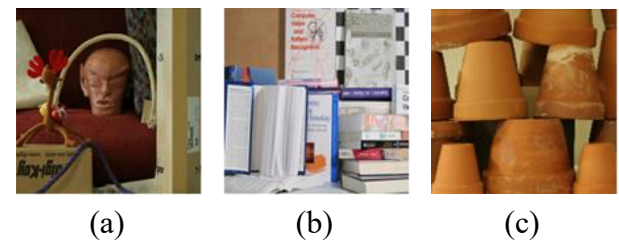


圖 4.2 標準影像[23](a) Reindeer (b) Books(c) Flowerpots

表 4.1 垂直邊緣視差準確率比較表

Method	Reindeer	Books	Flowerpots
SAD	77.38%	83.46%	74.42%
SSD	76.45%	83.67%	74.30%
NCC	74.82%	83.07%	74.31%
Proposed	78.30%	84.67%	76.07%

4-2 連通物件濾除雜訊

當區塊比對完成獲得所有垂直邊緣的視差後，我們首先會先做視差背景的濾除，理由是拍攝有距離限制，較遠的距離不是我們感興趣的地方。接著會利用連通物件將每個垂直邊緣相連接，並對鄰近的邊緣點標記相同的編號，以表示為同一個物件，進而濾除背景雜訊，降低對後續處理所產生的影響。但是連通物件是有相鄰點就會相連，不完全符合我們的需求。因此，在遇到相鄰點時，會額外加上相鄰點的視差必須與中心點的視差值小於 3 的判斷條件，不符合這個條件的邊緣點，就會給予新的編號。當完成物件連通後，將物件長度較短的邊緣點濾除並保留住大部分的主體邊緣。圖 4.3 呈現濾除物件前與濾除物件後實際拍攝的影像，圖 4.3(a)為輸入影像，圖 4.3(b)為垂直邊緣偵測後之結果。圖 4.3(c)為濾除背景視差的結果與圖 4.3(d)則是濾除連通長度較短後的結果。

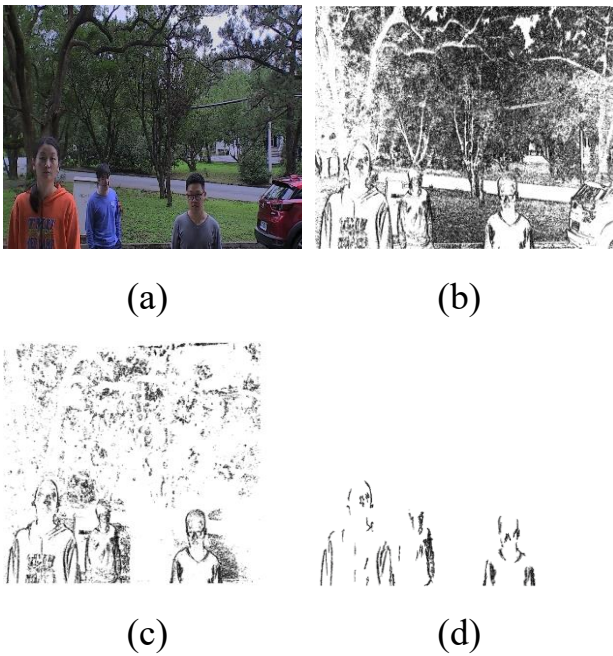
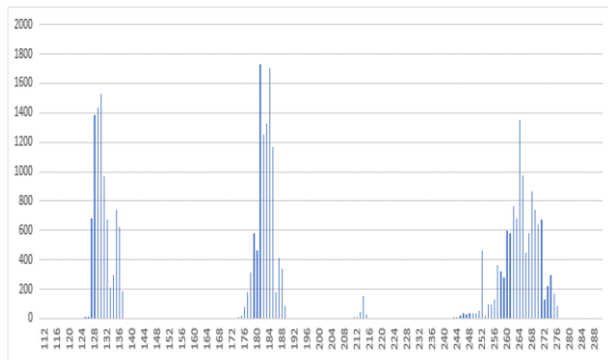


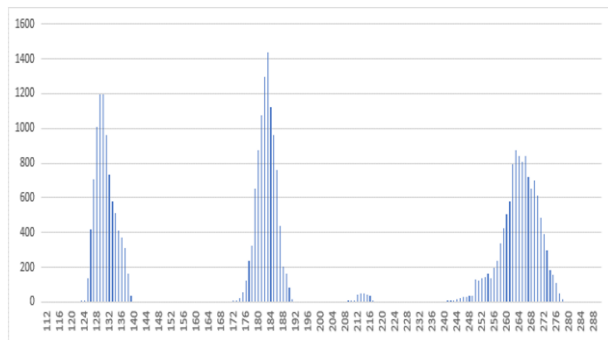
圖 4.3 濾除雜訊結果 (a)輸入影像 (b)垂直邊緣偵測結果 (c)濾除背景視差結果 (d)連通後濾除雜訊結果

4-3 直方圖尋找波峰

直方圖在影像處理中是一個非常重要的統計資料，常應用在影像增強中統計各個灰階值的數量，因此，我們將影像中垂直邊緣點的視差值做直方圖統計，因為對於同一個物件或人而言，視差值應相當接近。所以可利用這點來統計垂直邊緣的視差值。圖 4.4(a)為圖 4.3(d)所統計出的視差直方圖，為了讓曲線看起來平滑一些，首先將直方圖做一個平滑化，在此使用簡單的平均濾波，取中心點向外兩點，總共五點做平均，圖 4.4(b)即為平均濾波後的結果。



(a)



(b)

圖 4.4 直方圖平滑濾波前後結果 (a)平滑前直方圖 (b)平滑後直方圖

直方圖平滑濾波後，我們可以看到圖 4.4(b)有明顯的三個波峰，也就是在圖 4.3 中的三個行人。因此我們的目標就是找出直方圖中所有的波峰，其流程圖如圖 4.5 所示。

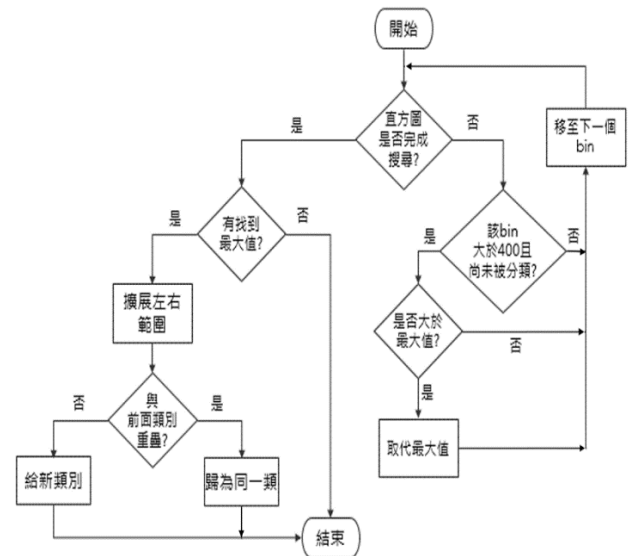


圖 4.5 尋找波峰流程圖

接著將圖 4.4(b)的視差直方圖找到的所有波峰整理到表 4.2 中，該表格是依序優先找到最大值然後由上排列到下，可以看到每個波峰形狀都不相同，尤其最後一個波峰看起來較為平緩，因此視差範圍就相對較廣。

表 4.2 垂直邊緣視差準確率比較表

視差值	視差範圍(個數)
183	175~191(17)
129	123~135(13)
263	248~278(31)

4-4 高斯分佈函數曲線擬合

由於每個波峰形狀都不相同，我們利用高斯分佈函數來擬合這些波峰，利用求得的高斯函數標準差，自動決定每個波峰的範圍。而擬合是採用最小平方法，由於函數的關係，需先將高斯函數轉換成一元二次方程式[24]。標準高斯函數公式如式(4.2)，其中， σ 為標準差， μ 為平均值，在此為波峰之最大值。

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (4.2)$$

轉換的公式如式(4.3)，先將式(4.2)中的 $1/\sigma\sqrt{2\pi}$ 代替成 A ，然後在等號兩邊同乘自然對數，整理後可以得到一個一元二次方程式如式(4.5)，其中 $a = \ln(A) - \mu/(2\sigma^2)$ 、 $b = \mu/(\sigma^2)$ 及 $c = -1/(2\sigma^2)$ 。

$$\ln(y) = \ln(A) + \frac{-(x - \mu)^2}{2\sigma^2} \quad (4.3)$$

$$= \ln(A) - \frac{\mu^2}{2\sigma^2} + \frac{2\mu x}{2\sigma^2} - \frac{x^2}{2\sigma^2} \quad (4.4)$$

$$= a + bx + cx^2 \quad (4.5)$$

接著，我們以最小平方方法擬合曲線。首先，其一元二次方程組可以寫成式(4.6)， n 為輸入資料個數。

$$\begin{cases} a + bx_1 + cx_1^2 = \ln(y_1) \\ a + bx_2 + cx_2^2 = \ln(y_2) \\ a + bx_3 + cx_3^2 = \ln(y_3) \\ \vdots \\ a + bx_n + cx_n^2 = \ln(y_n) \end{cases} \quad (4.6)$$

進一步將式(4.6)改寫成矩陣型式，如式(4.7)所示。

$$\begin{bmatrix} 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \\ 1 & x_3 & x_3^2 \\ \vdots & \vdots & \vdots \\ 1 & x_n & x_n^2 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} \ln(y_1) \\ \ln(y_2) \\ \ln(y_3) \\ \vdots \\ \ln(y_n) \end{bmatrix} \quad (4.7)$$

式(4.7)可寫成 $Ax = b$ ，但要求出係數矩陣 x ，還要先把左邊的矩陣 A 變成方陣才能求解，因此在等號兩邊同乘 A 的轉置矩陣，可得式(4.8)， N 為輸入資料個數， Σ 為每筆輸入資料的總和，以此類推。最後將係數矩陣 x 左邊的方陣同乘該方陣之反矩陣，就可以得到 a 、 b 及 c ，由於我們只需要標準差及平均值，因此在式(4.9)與式(4.10)透過 b 與 c 係數就能求出。透過式(4.9)及式(4.10)求得標準差及平均值後，我們在表 4.2 第一欄的波峰求出的擬合曲線如圖 4.6 所示。在圖中上方的函式，平均值為 146.58，標準差為 2.99，黑色虛線為原始資料，紅色實心曲線則是擬合後的結果。

$$\begin{bmatrix} N & \Sigma x & \Sigma x^2 \\ \Sigma x & \Sigma x^2 & \Sigma x^3 \\ \Sigma x^2 & \Sigma x^3 & \Sigma x^4 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} \Sigma \ln(y) \\ \Sigma x \ln(y) \\ \Sigma x^2 \ln(y) \end{bmatrix} \quad (4.8)$$

$$\sigma = \sqrt{\frac{-1}{2c}} \quad (4.9)$$

$$\mu = \frac{-b}{2c} \quad (4.10)$$

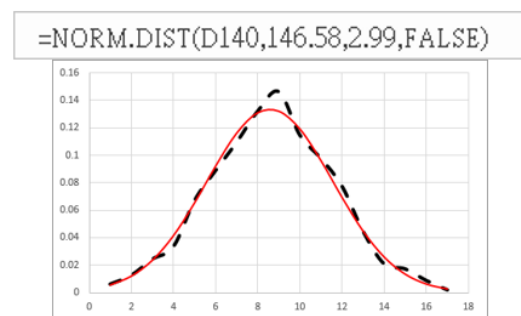


圖 4.6 一次曲線擬合結果

4-5 垂直投影

高斯分佈函數擬合每個波峰後，圖 4.7 呈現各個波峰所代表的視差平面，圖 4.7(a)距離攝影機最近，視差愈大，因此該視差平面屬於直方圖最右邊的波峰。圖 4.7(b)較 4.7(a)遠，屬於中間的波峰，則圖 4.7(c)則是屬於最左邊的波峰。

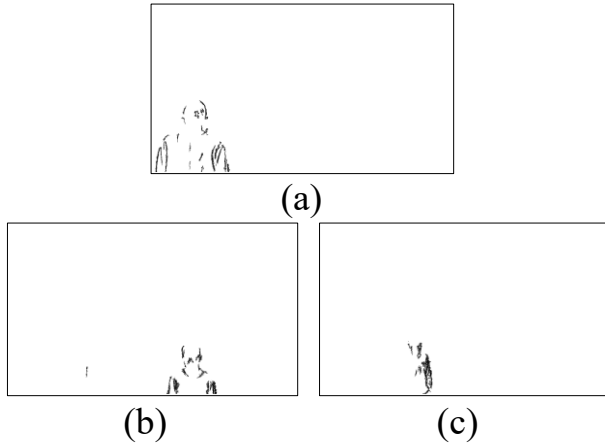


圖 4.7 各視差平面 (a)最大視差平面 (b)第二視差平面 (c) 最小視差平面

立體視覺公式如式(4.11)，其中 z 為與攝影機的距離， B 為基線，即兩支鏡頭間的距離， f 為鏡頭焦距， W_{img} 為影像寬度之解析度， W_{ccd} 為攝影機感光元件的寬度， d 為視差。影像與真實世界的關係為相似三角形，公式如(4.12)所示，其中 H 為物體真實世界的寬度，一般行人的寬度為 55 公分， N 則是在影像中物體的寬度，將式(4.11)的 Z 代換至式(4.12)，可得到式(4.13)。利用式(4.13)求得物體的寬度時，就可以檢查每個視差平面的寬度是否合理。

$$z = \frac{B \cdot f}{\frac{W_{ccd}}{W_{img}} \cdot d} \quad (4.11)$$

$$\frac{N \cdot \frac{W_{ccd}}{W_{img}}}{H} = \frac{f}{Z} \quad (4.12)$$

$$N = \frac{H \cdot d}{B} \quad (4.13)$$

基本上，利用視差平面的邊緣點即可找出方框的邊界。但是仍會有不合理的情況發生，像是圖 4.8 所示，其邊界因為左邊的雜訊而導致發生不合理的情況。



圖 4.8 投影前定位行人結果

因此，我們將不合理的寬度利用垂直投影邊緣點的方式，以中心點分為左右兩部分，接著，判斷左右邊緣點數差異，以圖 4.7(b)為例，下方小方框為投影結果，紅色實心線為中心點，因左邊點數與右邊點數相差懸殊，因此中心點會向右移動，一直移動至合理的邊界為止。圖 4.9 呈現圖 4.7(b)的投影結果，而圖 4.10 則是投影後定位行人的結果，黃色方框則是投影後更新的方框。

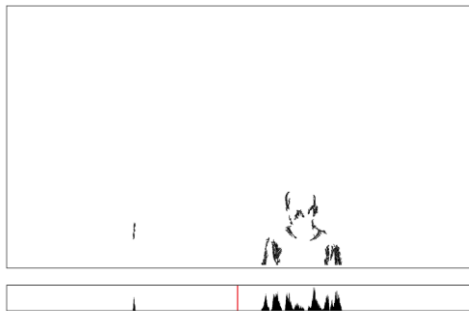


圖 4.9 垂直邊緣點投影結果



圖 4.10 投影後定位行人結果

五、實驗結果

本章節展示了二個部分，分別為二個人及三個人的情況，每組實驗依所提的人流計數演算法步驟呈現，依序為校正後的左、右輸入影像、垂直邊緣視差圖、依雜訊濾除後所統計垂直邊緣點的視差直方圖尋找視差直方圖波峰並對波峰進行高斯函數曲線擬合及最後行人定位的結果。在此，我們設定雙鏡頭的基線為 25cm，焦距為 6mm，感光元件尺寸為 1/3 吋，影像解析度為 1920×1080 像素。

第一部分為兩人的實驗結果，第一組呈現兩人站在不同距離的情況下的測試，視差搜尋範圍為 0~185。圖 5.1(d)呈現兩個不同顏色表示兩個不同視差平面，但是呈現在同一張影像中。

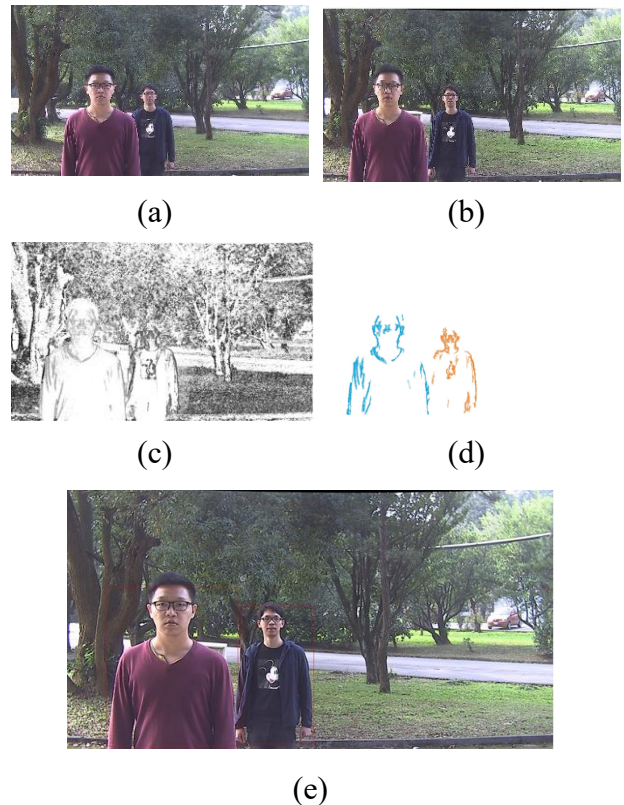
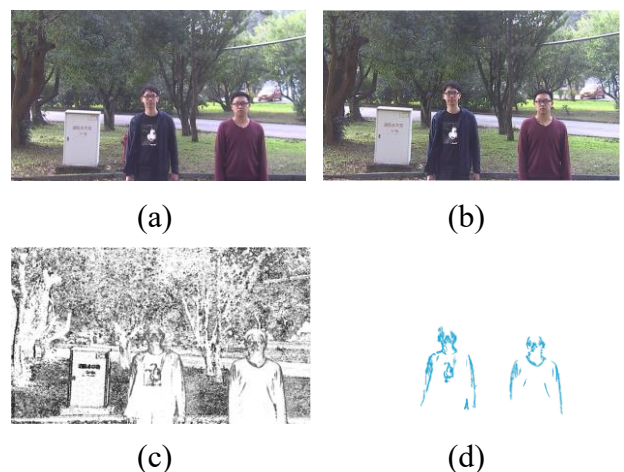


圖 5.1 兩人不同距離之實驗結果 (a)左影像 (b)右影像 (c)垂直邊緣視差圖 (d)高斯函數擬合之投影結果 (e)行人定位結果

第二組兩人則是考慮站在相同距離的實驗結果，由於只有一個明顯的波峰，故擬合後的結果也會切到同一個視差平面，如圖 5.2(d)，但是透過垂直投影的判斷後，最終圖 5.2(e)所呈現的結果則是可以將相同距離的行人分開。





(e)

圖 5.2 兩人相同距離之實驗結果 (a)左影像 (b)右影像 (c)垂直邊緣視差圖 (d)高斯函數擬合之投影結果 (e)行人定位結果

第二部分為三個人之實驗結果，視差搜尋範圍為 36~290。圖 5.3 呈現三個人的情況，前方行人在交疊物體的左側，儘管交疊會導致遮蔽的情形，但實驗結果也都能夠正確框出所有人，計算正確的行人數量。



(a)



(b)



(c)



(d)



(e)

圖 5.3 前後交疊三人之實驗結果 (a)左影像 (b)右影像 (c)垂直邊緣視差圖 (d)高斯函數擬合之投影結果 (e) 行人定位結果

六、結論與未來工作

本文提出只針對垂直邊緣進行區塊比對的人流計數方法與多數應用整張影像的比對不同，可以大量減少在平滑區域及水平邊緣點的比對時間，且能夠有效的利用行人的輪廓偵測出行人。此外，在區塊比對上也提出改變區塊大小的比對方式，用單一灰階值的區塊比對，解決了邊緣點在重複紋理上會容易發生比對錯誤的問題。並以連通物件方法加上判斷相鄰視差是否一致來濾除雜訊，進一步在後續直方圖的統計中獲得非常不錯的效果。最後，運用高斯分佈函數對直方圖中的波峰進行曲線擬合。實驗結果測試了兩人及三人的情況也顯示了不論在室外複雜的背景或是行人交疊的情況下，皆能有著良好的偵測結果。

參考文獻

- [1] T.-Y. Chen, C.-H. Chen, D.-J. Wang, and Y.-L. Kuo, "A people counting system based on face-detection," in *Genetic and Evolutionary Computing (ICGEC)*, 2010 Fourth International Conference on, pp. 699–702, 2010.
- [2] T. Ahonen, A. Hadid, and M. Pietikäinen, "Face Recognition with Local Binary Patterns," *Proc. Eighth European Conf. Computer Vision*, pp. 469–481, 2004.
- [3] T. Ma, Q. Ji and N Li, "Scene Invariant Crowd Counting Using Multi-Scales Head Detection in Video Surveillance," *IET Image Processing*, vol. 12, pp. 2258–2263, Aug. 2018.
- [4] C.-C. Chang and C.-J. Lin, "LIBSVM: A Library for Support Vector Machines," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol.2, no.3, pp. 1–27, Apr. 2011.
- [5] Q. Ye, "A Robust Method for Counting People in Complex Indoor Spaces," in *Education Technology and Computer (ICETC)*, 2010 2nd International Conference on, pp. 450–454, 2010.

- [6] N. Bernini, L. Bombini, M. Buzzoni, P. Cerri and P. Grisleri, "An Embedded System for Counting Passengers in Public Transportation Vehicles, " in *Proc. IEEE/ASME 10th Int. Conf. Mechatronic Embedded Syst. Appl. (MESA)*, pp. 1-6, Sep. 2014.
- [7] T. Darrell, G. Gordon, M. Harville, and J. Woodfill, "Integrated Person Tracking Using Stereo, Color, and Pattern Detection," *Proc. Conf. Computer Vision and Pattern Recognition*, pp. 601-609, June 1998.
- [8] V. Kolmogorov and R. Zabih, "Computing visual correspondence with occlusions using graph cuts," in *International Conference for Computer Vision*, vol. 2, pp. 508–515, 2001.
- [9] S. Birchfield and C. Tomasi, "Depth discontinuities by pixel-to-pixel stereo," in *Proceedings of the Sixth IEEE International Conference on Computer Vision*, Mumbai, India, pp. 1073–1080, January 1998.
- [10] R. Zabih and J. Woodfill, "Non-Parametric Local Transforms for Computing Visual Correspondence," *Proc. European Conf. Computer Vision*, vol. 2, pp. 151-158, 1994.
- [11] R. K. Gupta and S.-Y. Cho, "Window-based approach for fast stereo correspondence," *IET Comput. Vis.*, vol. 7, no. 2, pp. 123–134, May. 2013.
- [12] K. Zhang, J. Lu, and G. Lafruit, "Cross-based local stereo matching using orthogonal integral images," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 7, pp. 1073–1079, Jul. 2009.
- [13] H. Hirschmüller, "Stereo processing by semiglobal matching and mutual information," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 328–341, Feb. 2008.
- [14] "OpenCV Stereo BinarySGBM Class Reference," https://docs.opencv.org/3.4/d1/d9f/classcv_1_1Stereo_1_1StereoBinarySGBM.html
- [15] C. C. Pham and J. W. Jeon, "Domain transformation-based efficient cost aggregation for local stereo matching," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 7, pp. 1119–1130, Jul. 2013.
- [16] "Stereo Vision: Algorithm and Applications," www.vision.deis.unibo.it/smatt
- [17] R. A. Hamzah and H. Ibrahim, "Literature survey on stereo vision disparity map algorithms," *Journal of Sensors*, 2016.
- [18] "Richard's blog: 攝像頭校正," <http://wycwang.blogspot.com/2012/10/camera-calibration-part-2-calibration.html>
- [19] "Pinhole camera: 相機校正 (Camera Calibration)," <http://silverwind1982.pixnet.net/blog/post/153218861-pinhole-camera%3A-%E7%9B%B8%E6%A9%9F%E6%A0%A1%E6%AD%A3-%28camera-calibration%29>
- [20] Z. Zhang, "A flexible new technique for camera calibration," in *Proc. IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 11, pp. 1330–1334, Nov. 2000.
- [21] D.-W. Yang, L.-C. Chu, C.-W. Chen, J. Wang, M.-D. Shieh, "Depth reliability-based Stereo Matching Algorithm and its VLSI Architecture Design," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol.25, no.6, pp.1038–1050, Jun. 2015
- [22] S. Birchfield and C. Tomasi, "Depth Discontinuities by Pixel-to Pixel Stereo," *Proc. Sixth IEEE Intl. Conf. Computer Vision*, pp. 1073- 1080, Jan. 1998.
- [23] "Middlebury 立體視覺影像資料庫," <http://vision.middlebury.edu/stereo/>
- [24] L. He, Y. Chao, K. Suzuki, and K. Wu, "Fast connected-component labeling," *Pattern Recog.*, vol. 42, no. 9, pp. 1977–1987, Sep. 2008.