

# 融合色彩和深度資訊的動態人像切割

吳明霓  
國立臺中科技大學  
資訊管理系副教授  
mnwu@nutc.edu.tw

鄒豐懋  
國立臺中科技大學  
資訊管理系碩士生  
ddzoetracy@hotmail.com

## 摘要

本文研究提出的方法改善 Kinect 動態人像擷取，因為深度流失所產生人像切割不完整的問題。利用色彩資訊補強深度不確定的區域，得到較為完整的前景人像切割。首先我們將深度資訊分類為前景、背景、不確定的區域並產生一個只針對前景的不確定區域做為判斷區域，接著色彩資訊經由 Sobel 邊緣偵測、分水嶺等步驟進行影像切割，然後分別算出色彩區塊的平均值和標準差來做為色彩的特徵，最後再以前景邊緣對映到的色彩特徵與判斷區域進行比對，找出最好的結果，取得我們所要的人像。結果顯示能將人像完整切出，並且能大幅降低錯誤率。

**關鍵字：**Kinect、邊緣偵測、分水嶺、動態人像切割

## Abstract

The methods proposed in this article improve the dynamic human segmentation in Kinect. The problem of incomplete human segmentation caused by the loss of the depth can be solved by using color information to reinforce the areas where the depth is uncertain. By doing so, a more complete foreground human segmentation can be obtained. We begin with the classification of the depth information into foreground, background, and uncertain areas, producing a judging area, the foreground's uncertain areas. Afterwards, through Sobel edge detection, Watershed and other steps of color

information, images are segmented, and the mean and standard deviation of the color region are calculated respectively as the color feature. Finally, we compare the color feature of the foreground edge and the judging area to find the best results, obtaining the human segmentation we want. The results show that a complete human image can be cut out, and the error rate reduces significantly.

**Keywords:** Kinect、Edge detection、Watershed、Dynamic human segmentation

## 1.前言

在近幾年流行的智慧型裝置中，非常強調所謂的人機互動 (Human-Computer Interaction)，主要是讓人與機器之間能以簡單、正確、迅速的方式進行溝通，也就是能以直覺的方式來進行操作，進一步結合應用在擴增實境(Augmented Reality)上，增加多種的特徵定位，就能讓互動呈現更為細膩、豐富，這些提供特徵定位的方式例如：人臉辨識、手部姿勢辨識以及體感辨識[7]...等，而其中的體感辨識是目前經常被使用在商業界、娛樂界中，是一種較為新穎熱門的技術。使用前述這幾項技術的關鍵就在於影像處理，而處理通常是對於人物來進行。再任意的場景環境中，區分出前景影像的切割是個複雜的問題，儘管有許多的學者研究各種方式來解決，像是針對低景深影像進行顯著物件的切割[10]、影像顏色特徵使用粒子群演算法來分類物體群[1][5]、透過光度立體技術來達到基於形狀的圖像切割[2]或是可以利用三角測量的方式找出影像中人的

肢體進行切割[6]...等方式，但是影像特徵都必須要符合在指定的條件下來進行，所以影像切割仍具有很大的挑戰性。

在一個場景影像中大致可以分為前景和背景，而顯著物體大多會位於前景之中[9]，而當一個場景影像結合了 3D 的深度資訊，藉由物體所對映的深度位置，就可以快速的分類出前景和背景的背景區域，也能在任意的場景環境中進行所謂的動態人像切割。然而 3D 空間的資訊需要使用特殊儀器設備才能取得，但是這些設備的價格通常都居高不下[3]。如今 Microsoft 所推出的 Kinect 體感設備不但能夠降低成本的花費，並且能夠同時取得色彩資訊、深度資訊、人像辨識和人體骨架這些豐富資訊。Kinect 是使用紅外線投影的方式計算深度，由於紅外線在整個投影的過程中可能被光滑物體所折射或是被較黑的物體所吸收掉，會造成無法接收到訊息的問題，讓深度的資訊流失，最後造成人像切割在邊緣產生不規則的鋸齒狀或破洞的結果。

本文研究目的就是要來改良 Kinect 人像擷取所產生人像切割不完整的問題。在以前的研究中，Carlo 等人[3]在對於各種深度擷取儀器的比較實驗裡，提出了色彩資訊融合深度資訊可以使影像切割的結果更為良好。而 Zhi Liu 等人[10]則是利用低景深的場景進行切割，經由分水嶺切割後，區塊使用 trimap 分類為前景、背景和不确定區域時，針對不确定區域進行 Alpha matte 的處理，讓擷取的影像更為完整。我們結合了上述研究的優點，將色彩資訊融合深度資訊，提出了利用色彩資訊的特徵來比較歸類深度資訊中不確定的區域的方法。使這兩種資訊在切割的過程中達到相輔相成的效果，讓人像的切割更為完整。

## 2.深度區域的處理

本研究的目的是要把場景前景的人像完整的切割出來，過程中圖 1 概要的顯示了本文

方法的系統流程，一開始的流程是讓深度資訊和色彩資訊各別進行加工處理，深度資訊進行判斷區域的產生，色彩資訊進行特徵值得產生，接著對處理後的資訊進行結合，經由篩選的方法取得我們所要的人像結果。

首先在深度處理的部分，Kinect 會內部追蹤校準找出人像的深度區域，我們把這個區域拿來當作場景的前景，其他有偵測到的深度區域則當作背景，而未偵測到的深度設為不确定區域，以圖 2(c) 所示，紅色為前景區域，綠色為背景區域，白色為不确定區域，圖 2(d)則是表示用前景區域直接分出的人像。我們的研究是主要是找出在不确定區域中前景所流失的區域，然而我們可以發現前景區域不但有未偵測到資訊的問題，還有一些部分背景資訊被誤歸類到前景區域的錯誤，所以我們必須要定義一個新的不确定區域稱之為判斷區域，這個區域會排除掉資訊歸類錯誤的問題。

要產生判斷區域，首先我們先以[5x5]的圓侵蝕背景區域和前景區域，用來消除被錯誤分

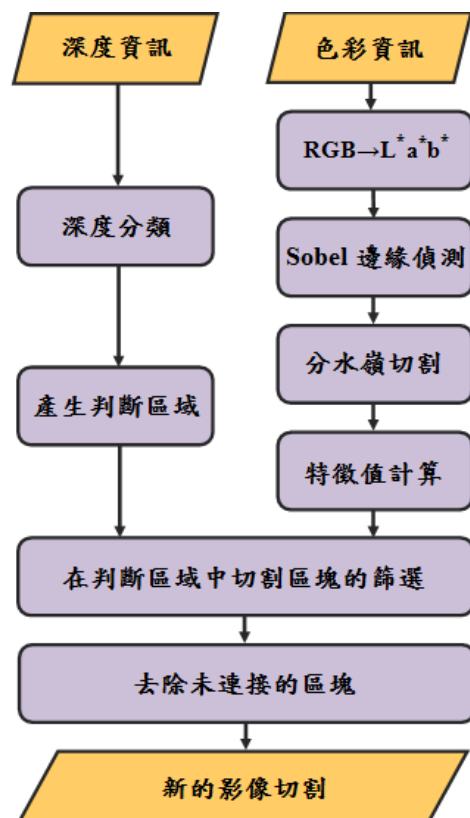


圖 1、系統流程圖

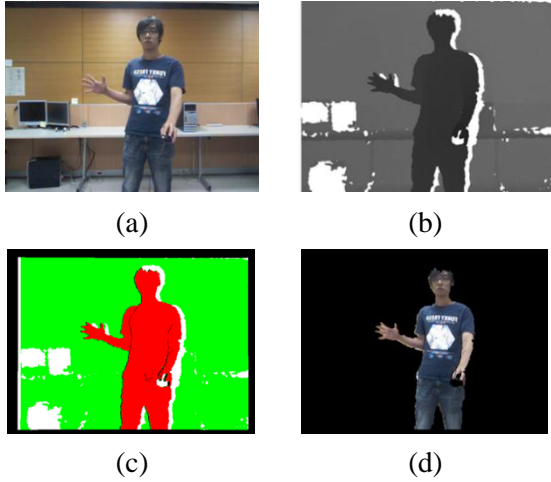


圖 2、Kinect 內所產生的資訊：(a)色彩資訊  
(b)深度資訊 (c)分類後的深度區域  
(d)追蹤到的人像直接切割

類到背景區域的前景資訊，接著我們為了要讓判斷的區域能產生在前景區域的附近，所以對前景區域進行 $[\text{odd}(D) \times \text{odd}(D)]$ 的圓進行擴張， $\text{odd}$ 表示讓數值無條件進位為奇數， $D$ 則表示在平面影像上頭部的骨架點到頸部骨架點之間的歐幾里得距離，這個參數可直接從設備上取得，用骨架是因為前景對於人像所失去的資訊中以頭部最為嚴重，有時候會嚴重到失去半顆頭，而 $D$ 這段骨架剛好大約為半顆頭部的距離，所以拿來當我們的擴張範圍，讓擴張的結果能夠包住整個人像。針對人像的可能區域轉換公式如(1)所示：

$$Hu^{possible}(i) = \max(Fg^{expansion}(i) - Bg^{erosion}(i), 0) \quad (1)$$

其中 $i$ 為目前的像素點； $\max()$ 為取最大值，而取出的最小值為0； $Fg^{expansion}$ 代表擴張後的前景區域； $Bg^{erosion}$ 代表侵蝕後的背景區域； $Hu^{possible}$ 則代表人像的可能區域。為了進一步縮小可能區域的範圍，我們要減去資訊沒有分類錯誤的前景區域來形成判斷區域，轉換公式如下(2)所示：

$$Judge^{area}(i) = \max(Hu^{possible}(i) - Hu^{identified}(i), 0) \quad (2)$$

$Hu^{identified}$ 代表著侵蝕後的前景區域，這個區域也代表在本研究中不會變動的已確定人像區域； $Judge^{area}$ 則代表判斷區域，作為後面實驗用來選取加入人像區域的重要區域。以上實驗處理的過程以圖3所示，圖3(a)和圖3(b)分別為深度資訊經由分類後所產生的背景和前景；圖3(c)為圖3(a)侵蝕後的結果，去除了部分背景資訊被歸類到前景的問題；圖3(d)為圖3(b)擴張後的結果，為一個包含完整人像的區域；圖3(e)為人像的範圍去掉確定背景區域所產生的可能區域；圖3(f)則為圖3(b)侵蝕後的結果，去除掉前景人像中所參雜到背景資訊的問題，後面的實驗會利用此區域算出人像的相關特徵；圖3(f)為可能區域中去掉確定的人像區域，最後產生的判斷區域，是接下來實驗用來判斷前景和背景的重要區域；圖3(f)為判斷區域對映到色彩資訊的結果，我們能看見判斷區域能同時包含前景人像和背景資訊。

### 3. 色彩資訊的切割

在色彩資訊處理的部分，由於 $L^*a^*b^*$ 被設計較接近人類的視覺，所以我們選擇它來進行處理，首先讓RGB的色彩空間轉為 $L^*a^*b^*$ 色彩空間，接著我利用上述特性進行接下來的運算，色彩資訊切割的方法我們將依序分為Sobel邊緣偵測、梯度值削弱和分水嶺切割、特徵值計算，三個步驟。

#### 3.1 sobel 邊緣偵測

由於亮度的變化在邊緣的呈現上比顏色變化較為豐富，所以我們使用 $L^*$ 來進行邊緣偵測。一開始我們為了讓之後分水嶺的切割不受雜訊影響，針對 $L^*$ 進行 $[3 \times 3]$ 的平滑濾波去除掉些微的雜訊。Sobel的方法會對影像中水平和垂直的邊緣產生梯度值，而這個梯度值越高則表示邊緣越明顯。

#### 3.2 梯度值削弱和分水嶺切割

分水嶺演算法[8]的分割概念是假設影像

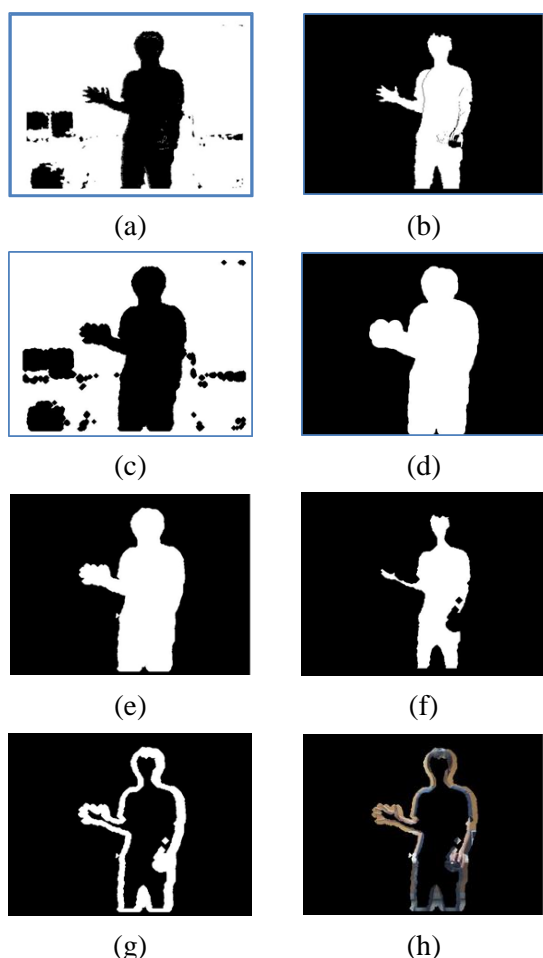


圖 3、判斷區域的產生：(a)背景區域 (b)前景區域 (c)背景侵蝕 (d)前景擴張 (e)可能區域 (f)確認人像區域 (g)判斷區域 (h)對映到判斷區域的色彩資訊

中像素點的值代表著高度，而當一個局部區域中找到最低點，該點所影響到的範圍就稱為蓄水盆，在多個蓄水盆之間的所交接邊界就稱為分水嶺。如圖 4 所示，假設圖 4(a)為 *Gradient* 梯度圖的梯度值，在圖 4(b)中梯度箭頭的指向將由像素點周圍 8 個方位的梯度值所決定，箭頭指向較小的值也就是較低的點，而當周圍的梯度值都大於目前像素點的梯度值時，就產生了這個區域的最低點，最後在影響範圍內劃分出區塊，進而形成分水嶺的切割。

由於 Sobel 的梯度圖中有些微弱的梯度值用來表示人眼難以察覺的邊緣特徵，這些值會影響分水嶺演算法的判斷，進而造成影像過度

切割的現象，因此我們想要削弱這些值，讓一些區塊能夠合併起來，排除過度切割的問題，並且能大量降低後面方法的運算量。對微弱梯度值削弱的公式如(3)(4)(5)所示：

$$e = (\max(\text{Gradient}) - \min(\text{Gradient})) \times t + \min(\text{Gradient}) \quad (3)$$

$$\text{NewGradient}(i) = \max(\text{Gradient}(i) - e, 0) \quad (4)$$

$$S = \text{watershed}(\text{NewGradient}) \quad (5)$$

在公式(3)中  $\max()$  表示為取最大值； $\min()$  為取最小值； $t$  表示在最大梯度值和最小的梯度值之間要削弱的程度範圍；而  $e$  代表著門檻值； $i$  為目前像素位置；在公式(4)裡，*Gradient* 為 sobel 所產生的梯度圖；*NewGradient* 表示為被削弱過後的梯度圖，過程中表示當像素中梯度值小於  $e$  時，則新的梯度值設為 0；*watershed()* 為進行分水嶺切割；最後公式(5)的  $S$  代表 *NewGradient* 經由分水嶺切割後的切割圖。其以上結果如圖 5 所示，圖 5(a)為經由 Sobel 邊緣偵測後所產生的梯度圖；圖 5(b)為分水嶺切割後，由於微小梯度的干擾太多所造成的過度切割；圖 5(c)為梯度圖削弱後經過分水嶺切割，我們能看出切出來的區塊較能呈現物體間的區塊。

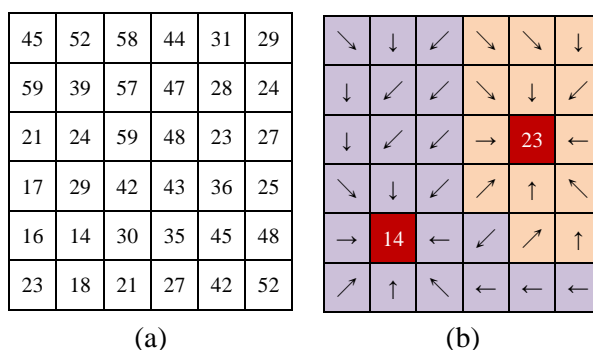


圖 4、分水嶺切割示意圖：(a)梯度圖的值 (b)分水嶺切割

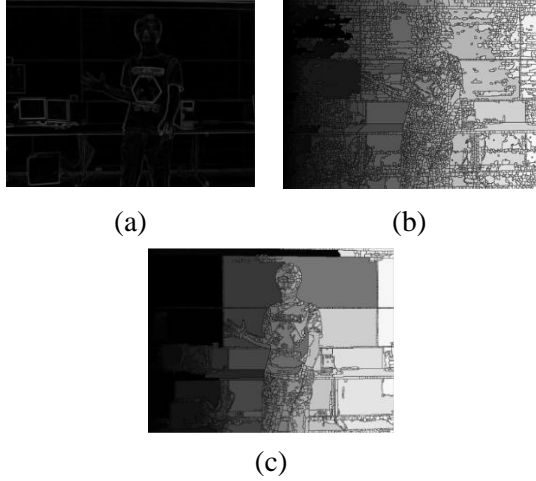


圖 5、邊緣偵測和分水嶺的結果：  
 (a)Soble 梯度圖 (b)分水嶺的過度切割  
 (c)梯度圖削弱後的水分嶺切割

### 3.3 特徵值計算

在本研究中的特徵值是指在一區塊內特徵數值化所代表的值，我們使用區塊的平均值表示該區塊整體的數值傾向，接著使用區塊的標準差表示區塊中數值的浮動程度。在計算的部分我們會對 $L^*$ 、 $a^*$ 、 $b^*$ 色彩資訊的質分別對映到 $S$ 中的區塊進行平均值與標準差的運算，以 $j$ 代表為目前在 $S$ 的區塊位置、 $S_j$ 代表為分水嶺切割後的區塊，而 $\bar{L}(S_j)$ 、 $\bar{a}(S_j)$ 、 $\bar{b}(S_j)$ 代表區塊 $S_j$ 所對映色彩資訊的平均值特徵， $L\sigma(S_j)$ 、 $a\sigma(S_j)$ 、 $b\sigma(S_j)$ 即為區塊 $S_j$ 所對映色彩資訊的標準差特徵。

### 4.判斷區域中的篩選

本研究提出的方法是利用在人像邊界區塊中特徵值與判斷區域裡區塊的特徵值來做比較，首先我們要先找到在人像邊緣的區域後，對映到 $S$ 的邊緣區塊，我們稱為人像邊界區塊，對映到 $S$ 的判斷區域，則稱為判斷區塊。公式的表達如(6)(7)(8)所示：

$$FB = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix},$$

$$Hu^{boundary} = \text{filter}(Hu^{identified}, FB) \quad (6)$$

$$HB_m^{region} = S_j, \text{ if}(Hu_i^{boundary} \in S_j^p) \quad (7)$$

$$Judge_n^{region} = S_j, \text{ if}(Judge_i^{area} \in S_j^p) \quad (8)$$

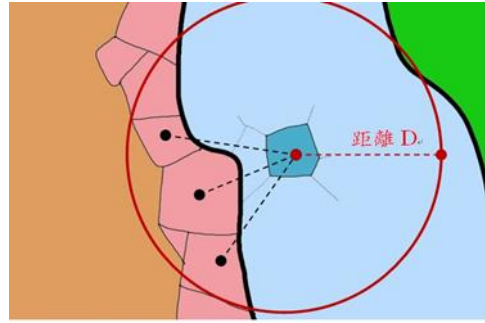
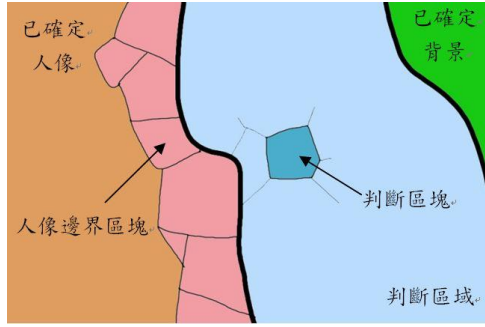
$FB$ 代表為邊緣偵測的遮罩；在公式(6)中 $\text{filter}()$ 代表濾波器； $Hu_i^{boundary}$ 代表為已確定人像區域( $Hu^{identified}$ )的邊緣；在公式(7)中 $HB_m^{region}$ 代表邊緣 $Hu^{boundary}$ 經過切割圖 $S$ 中的區塊，稱為人像邊界區塊；而公式(8)裡的 $Judge_n^{region}$ 代表判斷區域 $Judge_i^{area}$ 所涉及切割圖 $S$ 中的區塊，稱為判斷區塊；而 $m, n$ 則代表各自獨立的區塊位置。接著我們要讓判斷區塊跟人像邊界區塊來進行不相似的比較，而我們設計一個規則，找出那些區域能符合比較的條件規則如(9)所示：

$$\overline{Judge_n^{region}, HB_m^{region}} < D \quad (9)$$

其中 $\overline{Judge_n^{region}, HB_m^{region}}$ 代表各個區塊間最短的歐幾里得距離； $D$ 則代表前面所提到的頭部骨架點到頸部骨架點之間的半顆頭部骨架距離。整體主要是在表達以判斷區塊 $Judge_n^{region}$ 為中心，然後找出人像邊界區塊群 $HB^{region}$ 中兩區塊距離在 $D$ 範圍內的區塊，在這些被選中的區塊裡，我們會選擇最近的3個區塊當我們的比較區塊，而只要其中一段的區塊有符合接下來的比較條件， $Judge_n^{region}$ 這個區塊進一步成為我們所需要的結果，在這裡是一對多的關係，而這3段距離還必須要符合小於 $D$ 的限制，以達到縮小判斷的收尋範圍，才不會讓位於在不可能位置上的區塊因相似而被加入近來。圖6為示意圖。

上述我們已經取得了要比較的區塊，接著我們要算出區塊間的差異性，我們要讓找到的人像邊界區塊來減去判斷區塊的特徵值，而 $\bar{L}_{n,k}^{different}$ 、 $\bar{a}_{n,k}^{different}$ 、 $\bar{b}_{n,k}^{different}$ 表示兩區塊間的平均值的差值； $L\sigma_{n,k}^{different}$ 、 $a\sigma_{n,k}^{different}$ 、 $b\sigma_{n,k}^{different}$ 則表





● 要比較的區塊

圖 6、尋找比較區塊示意圖

示兩區塊間的標準差的差，值其中 $k$ 表示由最短距離選中 3 個區塊中的其中一個區塊的位置，這 3 個區塊之後都將獨自進行各自的運算。最後利用整體區塊間的平均值的差減去整體區塊間標準差的差，可以計算出各區塊間的不相似程度，值越大表示切割品質越佳[4]。所以我們參考這個特性來比較兩區塊間的不相似程度，所設計方法的公式如(10)(11)所示

$$\begin{aligned} L_{n,k}^{Not\_similar} &= \text{abs}(\bar{L}_{n,k}^{different} - L\sigma_{n,k}^{different}), \\ a_{n,k}^{Not\_similar} &= \text{abs}(\bar{a}_{n,k}^{different} - a\sigma_{n,k}^{different}), \\ b_{n,k}^{Not\_similar} &= \text{abs}(\bar{b}_{n,k}^{different} - b\sigma_{n,k}^{different}) \end{aligned} \quad (10)$$

$$\begin{aligned} pass_j &= S_j \text{ if } (S_j \in Hu_i^{identified}) \\ \text{or if } &\left( \begin{aligned} &L_{n,k}^{Not\_similar} < T_1 \\ &\text{and } a_{n,k}^{Not\_similar} < T_2 \\ &\text{and } b_{n,k}^{Not\_similar} < T_2 \\ &\text{and } Judge_n^{region} \in S_j \end{aligned} \right) \end{aligned} \quad (11)$$

在公式(10)中 $\text{abs}()$ 代表取絕對值，使用的原因是因為 $a^*$ 、 $b^*$ 的資訊中含有負值；而在(11)中 $L_{n,k}^{Not\_similar}$ 、 $a_{n,k}^{Not\_similar}$ 、 $b_{n,k}^{Not\_similar}$ 代表 $Judge_n^{region}$ 計算出來的不相似程度； $T_1$ 、 $T_2$ 則表示入選的門檻值，由於 $L^*$ 屬於亮度資訊， $a^*$ 、 $b^*$ 則為色彩資訊所以我們使用 2 種門檻值。在 $pass_j$ 的部分是用來統計結合有被通過的人像區塊，一開始我們讓，以確定的人像所涉及的區塊通過 $pass_j$ ，接著之前的方法，我們會經由 $k$ 的變化最多進行 3 次的不相似運算，而當其中一次結果符合我們要的條件，我們就把這個區塊 $Judge_n^{region}$ 所對應到的 $S_j$ 加進我們的 $pass_j$ 中。當 $pass$ 結果出來時我們會發現有些小區塊因相似而被選進來，但卻沒有和我們的人像相連接起來，所以我們比較區域間的面積，選擇較大的面積為我們所要的結果，所以那些小區塊自然的被排除掉。以上篩選的過程以圖 7 所示，圖 7 (a)為圖 3 (f)邊緣化後的人像邊緣；圖 7 (b)紅色的區塊為人像邊緣所涉及到的區塊，這些代表了要比較的人像資訊；圖 7 (c) 紅色的區塊為判斷區域所涉及到的區塊，用來賦予判斷那些區塊能加進到人像，最後形成完整的人像；

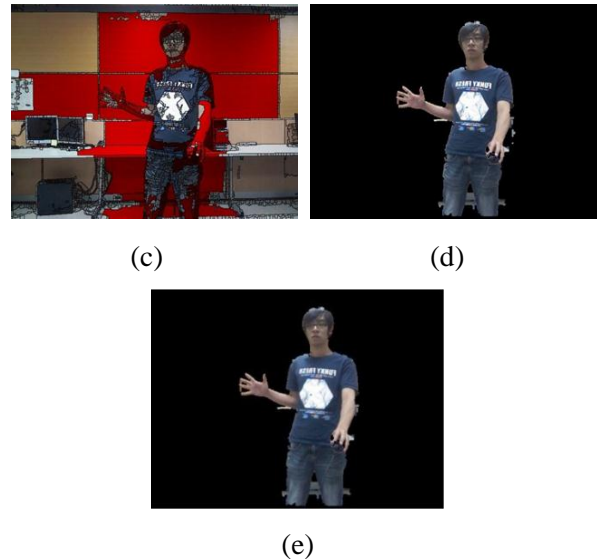


圖 7、篩選過程：(a)已確定人像的邊緣 (b)人像邊界區塊(c)判斷區塊 (d)有多餘區塊的結果 (e)實驗結果。

圖 7 (d)為篩選後的結果，我們發現有些小區塊沒有跟人像相連；圖 7 (e)為把沒相連的區塊去除後的結果，為我們實驗的最終結果。

## 5. 實驗

本研究在光線充足的實驗環境的環境下，經由多次的實驗結論，將梯度的門檻值 $e$ 設定為 0.01；亮度門檻值 $T_1$ 設為 25；色彩檻值 $T_2$ 設為 15，實驗結果以圖 8 表示。本實驗是針對單

張的影像進行處理，在圖 8 中分別以不同的衣服、姿勢、背景來分別實驗。我們利用統計切割錯誤率[10]的方法來測量實驗的結果，方法公式如(12)所示：

$$error(M_{seg}, M_{gt}) = \frac{\sum_{(x,y)} M_{seg}(x,y) \otimes M_{gt}(x,y)}{\sum_{(x,y)} M_{gt}(x,y)} \quad (12)$$

其中 $error$ 為結果的錯誤率； $M_{seg}$ 則代表輸入

















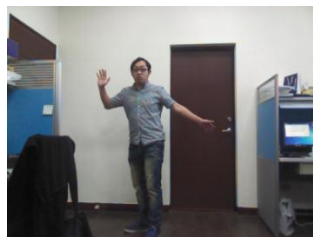



編號	原始影像	手動切割	Kinect 分割	本文方法
(a)				
(b)				
(c)				
(d)				
(e)				

圖 8、實驗結果

表 1、圖 8 的錯誤率和運算時間

編號	Kinect 錯誤率	本文方法 錯誤率	運算時間 (秒)
(a)	24.47%	9.39%	0.354476
(b)	14.26%	8.53%	0.347745
(c)	14.05%	7.79%	0.343879
(d)	16.28%	6.12%	0.342203
(e)	13.05%	7.09%	0.345199

Kinect 的切割結果或本文方法的切割結果 $M_{gt}$ 表示為手動切割的圖；⊗為 XOR 布林運算，所以輸入的切割資料離手動切割的差距越大，錯誤率則越高，表 1 為依據圖 8 的 5 張影像所算出的錯誤率和運算時間。在錯誤率方面，從 Kinect 分割結果中我們發現人像的頭都有產生很大的資訊流失，人臉是人像最為重要的地方，而我們提出的方法較能把人像的頭部完整呈現出來。在運算時間方面，本文方法使用 MATLAB 來進行單張影像處理，運算時間大約落在 0.35 秒附近，優於 Carlo Dal Mutto [3] 等人所提出方法所花費的 0.5 秒，本文的方法在圖 8(e)中我們發現了人像的右手因為顏色的關係，能很容易的被切出來，可是在左手陰影部分的顏色和門的顏色因相似而被歸為一類，所以導致部分區域被歸類在門的一方，這是難以避免的情況。整體的實驗結果表示本文方法能把人像較為完整的呈現，而錯誤率大約都在 10% 以內，改善了原本 Kinect 的切割。

## 6. 結論

本文提出了改善 Kinect 動態人像切割的方法，經由深度資訊的區域分類來結合色彩資訊的分水嶺切割並進行不相似程度的特徵值篩選，實驗結果顯示已大幅改善原本只利用深度資訊所進行的動態人像切割效果，而取得完整的人像資訊未來能夠被應用於擴增實境、人機互動的環境上。

## 參考文獻

- [1] Asanga Ratnaweera, Saman K. Halgamuge, Harry C. Watson, "Self-Organizing Hierarchical Particle Swarm Optimizer With Time-Varying Acceleration Coefficients", IEEE Transactions on Evolutionary Computation, VOL. 8, NO. 3, p.240-255, 2004.
- [2] Carme Julià, Rodrigo Moreno, Domènec Puig, Miguel Angel Garcia, "Shape-based image segmentation through photometric stereo", Computer Vision and Image Understanding 115, p.91-104, 2011.
- [3] Carlo Dal Mutto, Pietro Zanuttigh, Guido M. Cortelazzo, "Fusion of Geometry and Color Information for Scene Segmentation", IEEE Journal of Selected Topics in Signal Processing, VOL. 6, NO. 5, p.505-521, 2012.
- [4] Carlo Dal Mutto, Pietro Zanuttigh, Guido M. Cortelazzo, "A probabilistic approach to ToF and stereo data fusion", in Proc. 3DPVT, Paris, France, 2010.
- [5] Chi-Yu Lee, Jin-Jang Leou, Han-Hui Hsiao, "Saliency-directed color image segmentation using modified particle swarm optimization", Signal Processing 92, p.1-18, 2012.
- [6] Jun-Wei Hsieh, Chi-Hung Chuang, Sin-Yu Chen, Chih-Chiang Chen, Kuo-Chin Fan, "Segmentation of Human Body Parts Using Deformable Triangulation", IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, VOL. 40, NO. 3, MAY, p.596-610, 2010.
- [7] Kittiya Khongkrapan, Pakorn Kaewtrakulpong, "A Novel Reconstruction and Tracking of 3D-Articulated Human



- Body from 2D Point Correspondences of a Monocular Image Sequence”, IEICE Transactions on Information and Systems, VOL.E94–D, NO.5, p.1090-1098, 2011.
- [8] Luc Vincent, Pierre Soille, “ Watersheds in Digital Spaces: An Efficient Algorithm Based on Immersion Simulations”, IEEE Transactions on Pattern Analysis and Machine Intelligence, VOL. 13, NO. 6, p.583-598, 1991.
- [9] Wang Luo, Hongliang Li, Guanghui Liu, King NgiNgan, “Global salient information maximization for saliency detection”, Signal Processing: Image Communication 27, p.238-248, 2012.
- [10] Zhi Liu, Weiwei Li, LiquanShen, Zhongmin Han , Zhaoyang Zhang, “Automatic segmentation of focused objects from images with low depth of field”, Pattern Recognition Letters 31, p.572-581, 2010.