

第三章、臉部偵測

本章在介紹臉部的偵測工作，其流程如圖 3.1 所示，包括擷取膚色部位、修補破碎膚色以及計算近似橢圓三個工作，首先會簡介擷取膚色時所使用的顏色空間，接下來章節 3.1 中，說明如何利用此空間的顏色值來擷取膚色部位，而 3.2、3.3 及 3.4 則分別介紹擷取膚色部位、修補破碎膚色及計算近似橢圓三個工作的詳細內容。

3.1 截取膚色部位的方法

$YCrCb$ color space 廣為使用在擷取影像中的膚色部位[Cha99]，是由 RGB color space 經過線性轉換而成，其數學式如下所示。

$$\begin{bmatrix} Y \\ Cr \\ Cb \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.500 & -0.4187 & -0.0813 \\ -0.1687 & -0.3313 & 0.500 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} 1 \\ 128 \\ 128 \end{bmatrix} \quad (3.1)$$

其中 Y 代表亮度值， Cr 、 Cb 皆表示色度值， R 代表代表紅色， G 代表綠色， B 代表藍色。色度值與亮度值分離的好處，是在擷取膚色過程中不易受光線的變化而干擾結果。[Cha99] 的研究指出由色度值 Cr 與 Cb 共同定義出膚色值，其範圍分別為 $Cr = [133 \ 173]$ 與 $Cb = [77 \ 127]$ ，可擷取影像中不同人種的膚色。

由於光線強烈地變化下，由 $YCrCb$ color space 擷取膚色的結果還是會受影響，所以 Lievin 等人在 2004 年提出 LUX color space，是從 $YCrCb$ 經過非線性的數學轉換而得，可以使色度從亮度中分離更完整，其數學式如下所示（取自[Lie04]）：

$$\begin{aligned}
L &= (R + 1)^{0.3}(G + 1)^{0.6}(B + 1)^{0.1} - 1 \\
U &= \begin{cases} \frac{M}{2} \left(\frac{R+1}{L+1} \right) & \text{if } R < L, \\ M - \frac{M}{2} \left(\frac{L+1}{R+1} \right) & \text{otherwise.} \end{cases} \\
X &= \begin{cases} \frac{M}{2} \left(\frac{B+1}{L+1} \right) & \text{if } B < L, \\ M - \frac{M}{2} \left(\frac{L+1}{B+1} \right) & \text{otherwise.} \end{cases}
\end{aligned} \tag{3.2}$$

其中 L 代表亮度值， U 、 X 代表色度值， R 代表紅色， G 代表綠色， B 代表藍色。其中 Lievin 等人另外發現人類膚色的紅色值 (R) 會大於亮度值 (L)，可輔助於判斷膚色。同時亦發現自然界中亮度值 (L) 與綠色值 (G) 的差異很小，如圖 3.5.(a) 的比較可得 (取自 [Lie04])，故以 G 替換掉計算複雜的 L ，再將 U color model 定義成新的 color model: \hat{U} ，如下所示：

$$\hat{U} = \begin{cases} 256 \times \frac{G}{R} & \text{if } R > G, \\ 255 & \text{otherwise.} \end{cases} \tag{3.3}$$

其中 R 代表代表紅色， G 代表綠色。偵測膚色的方法為當影像中任一像點的 R 大於 G 時，則將 256 乘上 G/R 作為 \hat{U} 值，其膚色值的範圍則為 $\hat{U} = [0 \ 254]$ ([Lie04])，亦可偵測出嘴唇的顏色。Lievin 等人比較了 Cr ， H (HSI color space 中的 Hue 值)， U 和 \hat{U} 等四種偵測膚色的準確性，如圖 3.1 顯示， \hat{U} 可以較準確地偵測出影像中的膚色區域，改良了以 Cr 和 Cb 偵測膚色的方法。

經過偵測膚色的比較，如圖 3.2 所示，發現在光線較暗的情況下使用 Cr 和 Cb 偵測出來的膚色區域會破碎的很嚴重，使用 \hat{U} 偵測膚色則會將許多非膚色值誤判成膚色，因此同時採用 \hat{U} 與 Cb 值來偵測膚色，則可以解決上述問題，使偵測出的膚色區域比較完整。但是不管是使用 Cr 與 Cb 偵測膚色，還是使用 \hat{U} 與 Cb ，都會將紅色誤判成膚色。因

此，我們經過觀察而改良 \hat{U} color model，將判斷條件加上當 R 大於 G 時， R/G 亦須小於 1.5 的情況才是膚色值。故形成新的 color model，令為 U' ，其表示式如下：

$$U' = \begin{cases} 256 \times \frac{G}{R} & \text{if } \frac{R}{G} < 1.5 \text{ and } R > G > 0, \\ 255 & \text{otherwise.} \end{cases} \quad (3.4)$$

其中膚色值的範圍為 $U' = [0 \ 249]$ 和 $Cb = [77 \ 127]$ ，而由 U 與 Cb 偵測影像中的膚色，是目前最好的方法。

3.2 膚色部位截取

影像輸入系統後，先擷取影像中的膚色部位以便得到駕駛者的臉部位置。首先先降低影像的解析度，以便減少處理時間並去除一些雜訊點。降低一半解析度的影像令為 I' ，如圖 3.4(b) 所示。接下來開始偵測影像的膚色值，先計算出影像 I' 的 U' component 和 Cb component（如 3.1，3.4 式），分別存為影像 $F_{U'}$ 及 F_{Cb} （面積同等於 I' ），如圖 3.4(c) 與 (d)。 $F_{U'}$ 中若有一像點的值介於膚色值範圍內（ $R_{U'} = [0 \ 249]$ ），而且 F_{Cb} 中相同位置的值也介於膚色值範圍內（ $R_{Cb} = [77 \ 127]$ ），此像點則判斷為膚色點，將其值設為 0，另外存成一張影像為 F_S ；若不符合上述條件的像點則為非膚色點，其值設為 255，存於 F_S 中，上述以數學表示式如下：

$$F_S(x, y) = \begin{cases} 1, & F_{Cb}(x, y) \in R_{Cb}, F_{U'}(x, y) \in R_{U'} \\ 0, & \text{otherwise} \end{cases} \quad (3.5)$$

其中 (x, y) 表示像點在影像中的位置，而 $F(x, y)$ 表示像點在影像 F 中位置為 (x, y) 的值。此時 F_S 即為擷取自影像 I' 的膚色區域，如圖 3.4(e) 所示。

3.3 修復破碎膚色

由於強光下的影像，由任一方法擷取出來的膚色區塊皆會是嚴重破碎的，如圖 3.5 所示，因此接下來進行修補動作，以便得到完整的膚色區塊。根據觀察，臉部輪廓的邊緣較不受光線的變化（如圖 3.5.(e)所示），所以可以使用 Sobel edge detection 的方法將影像 I' 取出邊緣。於是影像分割成數個區塊後，在這些分割的區塊內，若含有膚色點，而且破碎區塊的亮度值大於 170，則將此區塊填補成膚色，如圖 3.5.(f)，在此圖中黑色點表示處理後的膚色區塊，由此可看到其呈現較完整的臉部區塊。其他人的實驗如圖 3.6，將強光下擷取的破碎的膚色區域修補好。

利用邊緣將膚色部位修補好後，影像中會包含一些不必要的、細小的膚色區塊，如圖 3.7.(a)所示。因為駕駛者的膚色範圍在影像畫面中很廣，面積最大的膚色區塊即為駕駛者的臉部，所以利用連通區塊（connected component）的方法可將膚色點聯成一個個的連通區塊，同時計算面積大小，保留面積最大的連通區塊，就可移除小碎塊，結果為圖 3.7.(b)所示。

3.4 計算近似橢圓

擷取最大塊的膚色區塊後，發現其通常還會包含駕駛者的脖子部分，故須依據人臉比例找出符合臉部的橢圓，找出臉部正確的位置。所以先設定橢圓的三個參數：長軸（令為 a ）、短軸（令為 b ）和中心（令為 (x', y') ）。首先是找出橢圓的中心，於是將影像 F_S 中每

一像點的水平方向及垂直方向的座標位置加總後除以區塊的面積大小而求得座標的平均值，即為區塊的重心所在，當作橢圓的中心點 (x',y') ，如圖 3.8.(a) 的 (old center) 處，其數學式如下所示：

$$\begin{aligned} x' &= \frac{1}{|R_S|} \sum_{F_S(x,y) \in R_S} x, \\ y' &= \frac{1}{|R_S|} \sum_{F_S(x,y) \in R_S} y \end{aligned} \quad (3.7)$$

上式中 R_S 表示膚色區域， $|R_S|$ 表示面積大小。接著從影像上方設定 R_1 範圍， R_1 寬度為重心往左右各距離數個像點，在這範圍內算出膚色區塊的平均的頂點位置，而此頂點與重心的距離則當作橢圓長軸（令為 a ）。接著要算出臉部區塊的平均臉寬，我們設定 R_2 範圍（如圖 3.8.(a)所示），範圍由 (x',y') 分別往上及往下延伸 a 的 0.4 倍的長度，在此範圍內找出膚色區塊左半邊的平均界線 X_{left} 。同理在臉部區塊的右邊處 R_3 範圍內，算出膚色區塊平均的右邊界線 X_{right} ，則 $(X_{right} - X_{left})$ 即為臉部的臉寬，取其一半當作橢圓的短軸，如圖 3.8. (b) 中的 b 值。

此時得到橢圓的中心和長短軸，但是膚色區塊中通常會包含脖子部分，所以中心點會略為往下偏移，長軸也會過長，所以還需要對中心點及長軸作一次調整。根據觀察，人類的臉長通常為臉寬的 1.1 至 1.5 倍。這裡根據實驗將比例定為 1.1、1.2 和 1.5，調整更新長短軸的條件如下所示：

$$a = \begin{cases} 1.1b & , \text{if } 0.8b < a < 1.2b; \\ 1.2b & , \text{if } 1.2b < a < 1.5b; \\ 1.5b & , \text{if } 1.5b < a \end{cases} \quad (3.8)$$

若長軸 a 是短軸 b 的 0.8~1.2 倍，則 a 更新為 1.1 倍的 b ；若長軸 a 是短軸 b 的 1.2~1.5 倍，

則 a 更新為 1.2 倍的 b ；若長軸 a 是短軸 b 的 1.5 倍以上，則 a 更新為 1.5 倍的 b 。更新好橢圓的長短軸後，接著更新中心點 (x',y') ，做法是取臉寬的中心位置為 x' 新的座標值；而新的 y' 值 則為臉部區塊的頂點向下延伸 a 處，公式如下：

$$\begin{aligned}x' &= \frac{X_{left} + X_{right}}{2} \\y' &= h - a\end{aligned}\tag{3.9}$$

其中 h 表示臉部區塊的頂點。結果如圖 3.8.(a)所示，臉部區塊內舊中心 (old center) 調整為新中心 (new center)，此時的橢圓即完成，可得到臉部的正確位置。實驗如圖 3.9，不同人的臉型皆可套用相符的橢圓，以便找出臉部的位置。

於是到此步驟可得到影像中駕駛者正確的臉部位置，接下來由此臉部偵測駕駛者的眼睛，而此工作將於下一章節介紹。

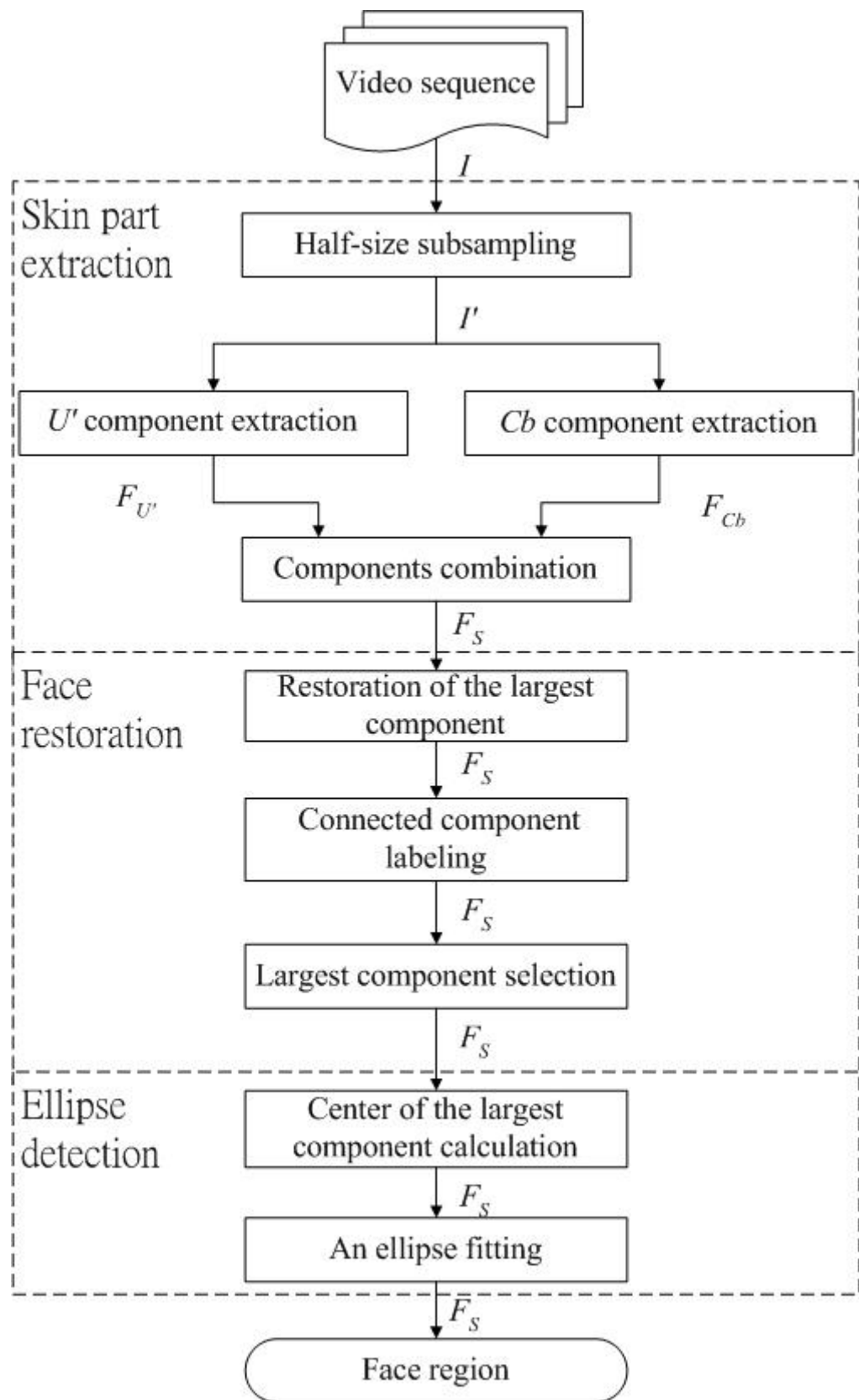


圖 3.1 臉部偵測系統流程圖。



圖 3.2 (取自[Lie04]) 使用不同的 color space 轉換後的色度影像，(a) 由左至右分別為不同 color space 的 Y , I , L 和綠色值 G ，發現彼此間差異度很小；(b) 由左至右分別為 Cr , H , U 和 \hat{U} 。

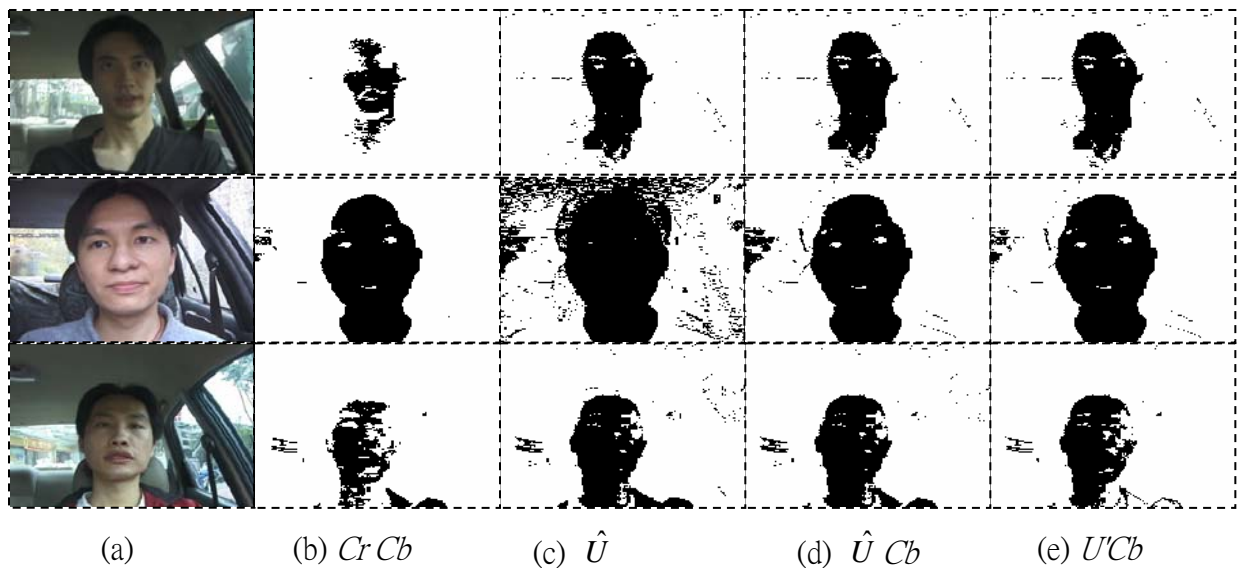


圖 3.3 使用不同的 color space 偵測膚色。(a)為輸入影像；(b) 使用 Cr model 和 Cb model 偵測出(a)圖的膚色部位，以黑點表示膚色；(c) \hat{U} color model；(d) \hat{U} color model 和 Cb color model，發現紅色也會判斷成膚色；(e)使用 U' color model 和 Cb color model 偵測膚色，改善了紅色偵測為膚色的缺點。

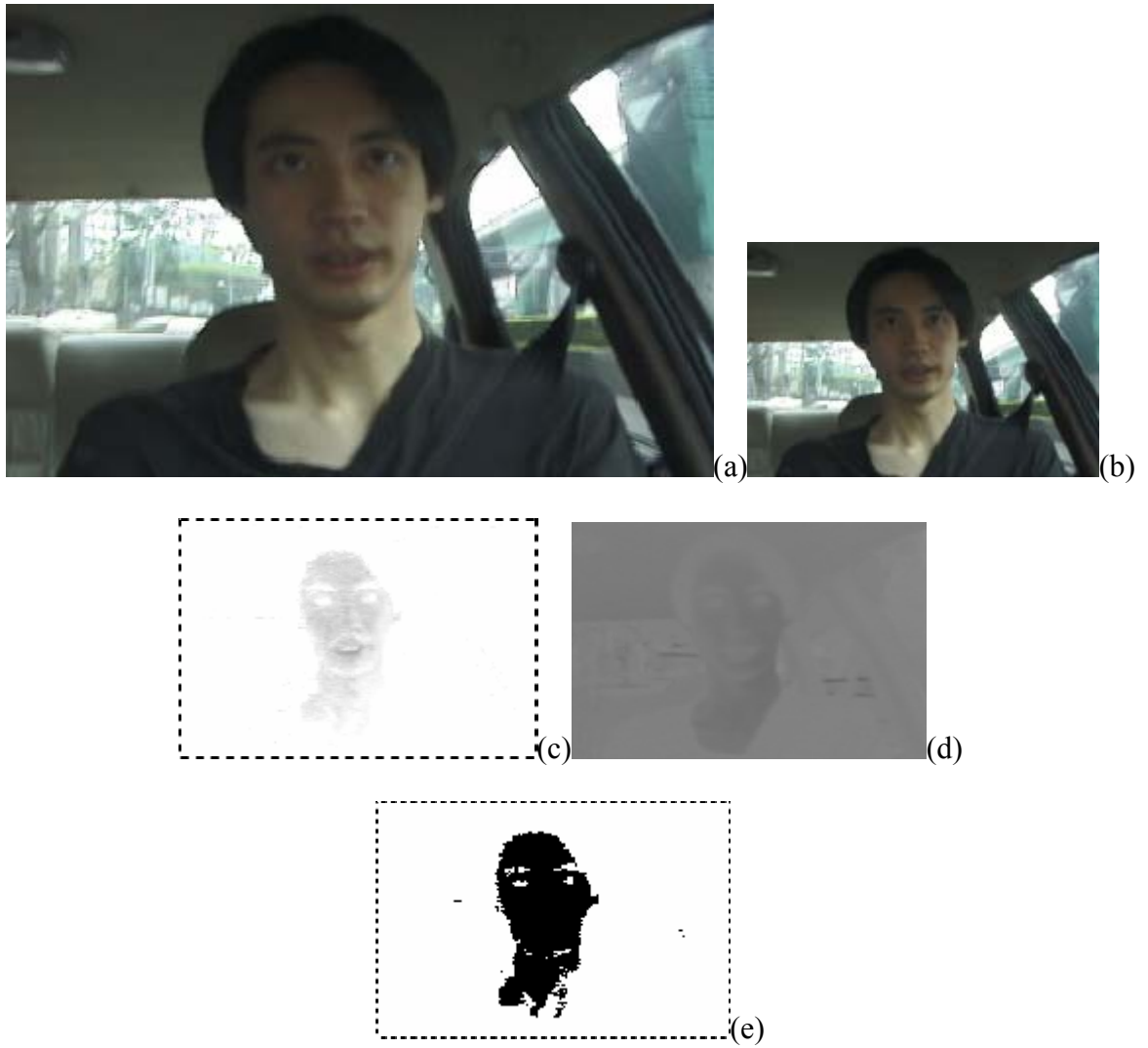


圖 3.4 偵測影像中的膚色區域。(a)輸入的影像；(b)將(a)降低解析度；(c)為(b)的 U' 值，存成一張影像；(d)為(b)的 Cb 值，存成一張影像；(e)由(c)與(d)的膚色點交集而成，是(b)中的皮膚部位，以黑色點表示。

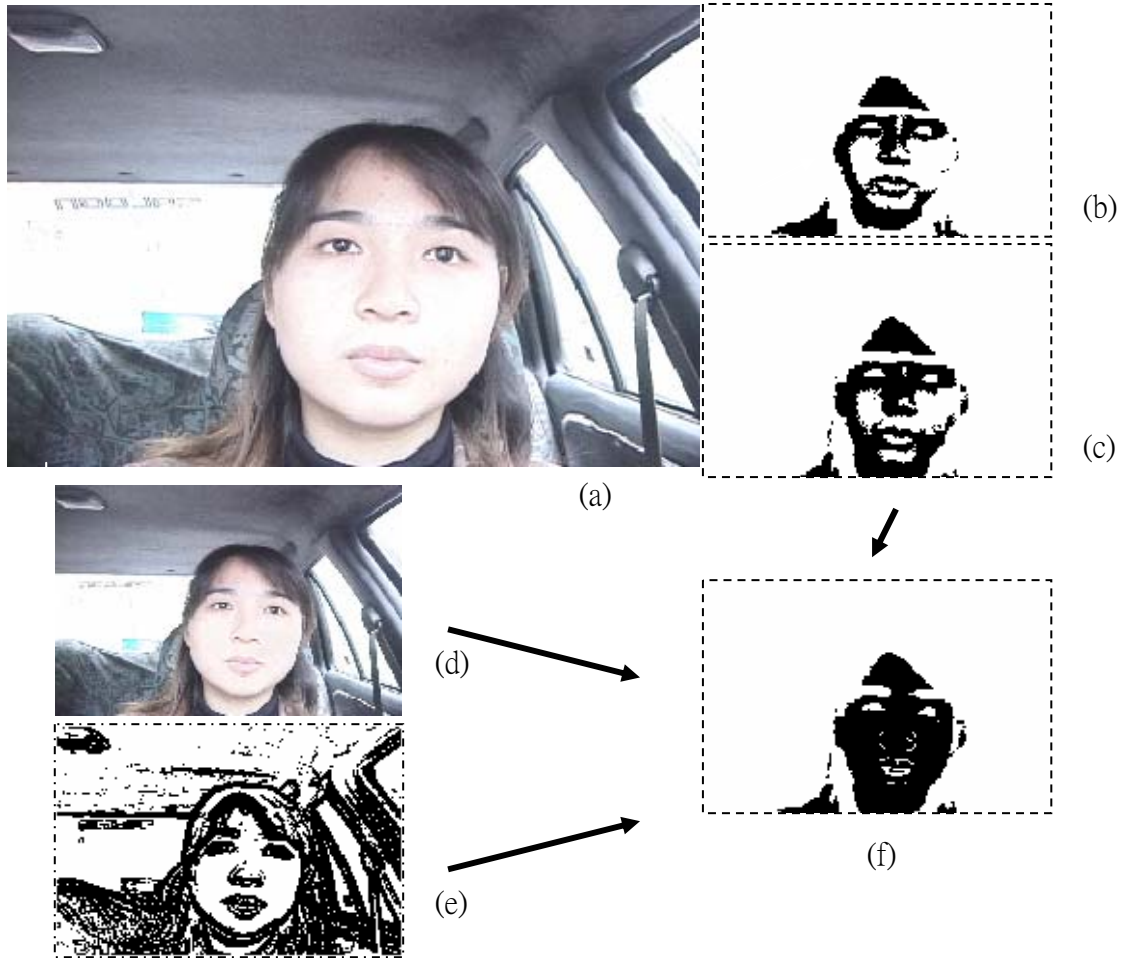


圖 3.5 修復膚色的過程。(a)為強光下的輸入影像，駕駛者的臉部呈現曝光現象。(b)使用 C_r 和 C_b 擷取影像中的膚色部位，呈現嚴重的破碎。(c)使用 U' 和 C_b 擷取的膚色部位稍微比(b)好，但是曝光部分還是破碎的。(d)將(a)降低一半解析度的影像。(e)將(d)經過 Sobel 取邊後的影像。(f)利用(d)(e)兩張修補後取出完整輪廓的膚色部位，以黑色點表示。



圖 3.6 修復破碎的膚色之實驗結果。其第一直排影像皆為強光下拍攝的，所以臉部右方顏色呈現白色，在偵測膚色時會呈現嚴重的破碎如第三直排影像。第二直排影像為邊線影像，臉部輪廓有清楚的邊線圍繞。經過修復動作後如第四直排影像，可呈現清晰的輪廓而找出近似橢圓。（第一~第三橫排為同一地點、同一時間拍攝的，而第四橫排則為不同時間拍攝的。）

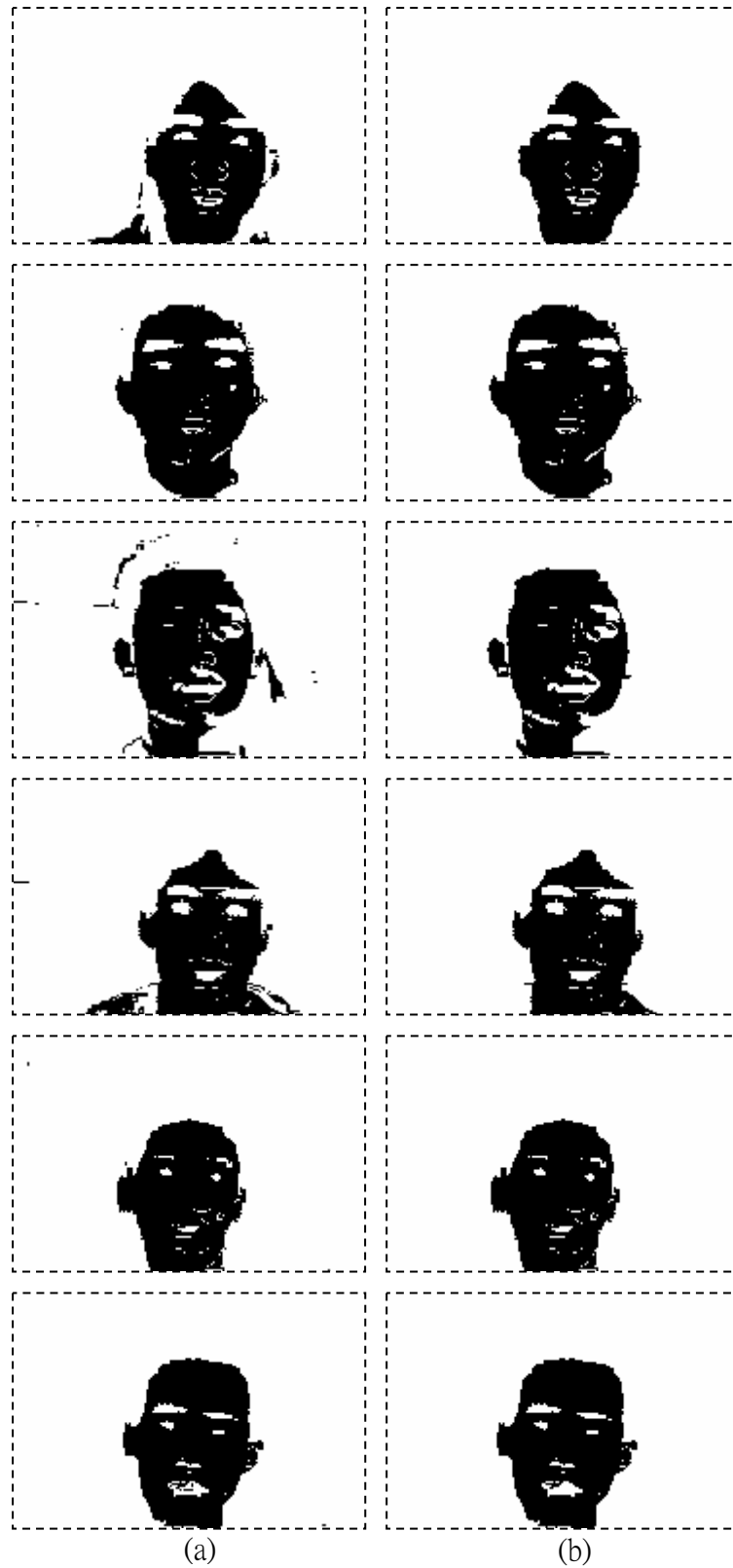
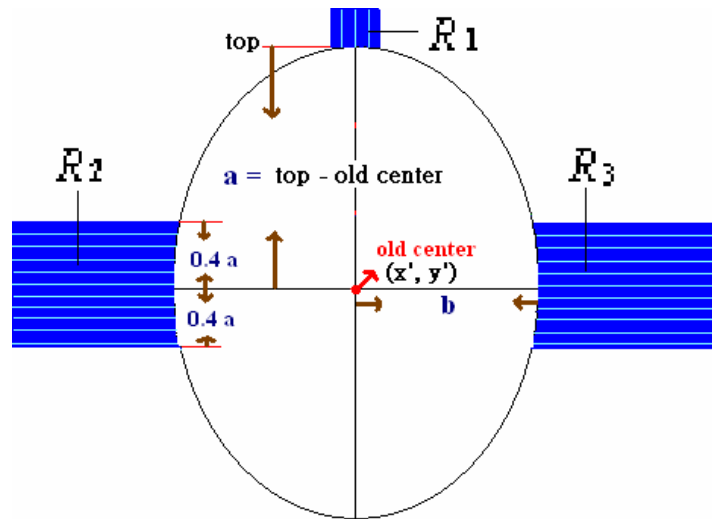
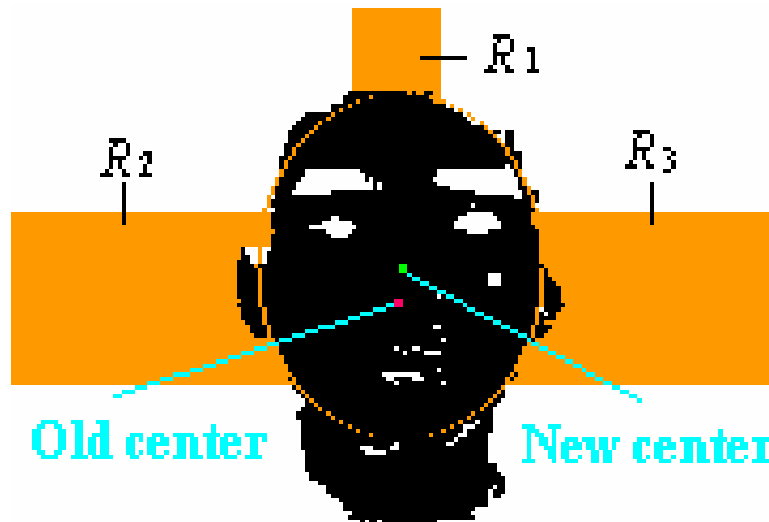


圖 3.7 去除碎塊的膚色區域。(a)從三張不同的輸入影像中擷取膚色部分，含有不屬於臉部的碎塊；(b)選擇最大塊的連通區塊，可去除碎塊。



(a)



(b)

圖 3.8 找出近似臉部的橢圓的示意圖：(a)先算出重心（old center）後，由 R2、R3 區域分別算出平均的臉寬，當作橢圓短軸（(b)中的 b 值），此短軸乘上一個倍數（1.2）得到橢圓長軸（(b)中的 a 值）。接著從 R1 區塊算出近似橢圓的頂端，在頂端向下延伸一個長軸的長度之點，重新定義為新的重心（new center），此點才為較準確的橢圓中心。並且從新重心往上取高度為 0.8 倍的長軸、寬度為兩倍短軸的長方形，當作眼睛帶，一雙眼睛勢必在此範圍內。

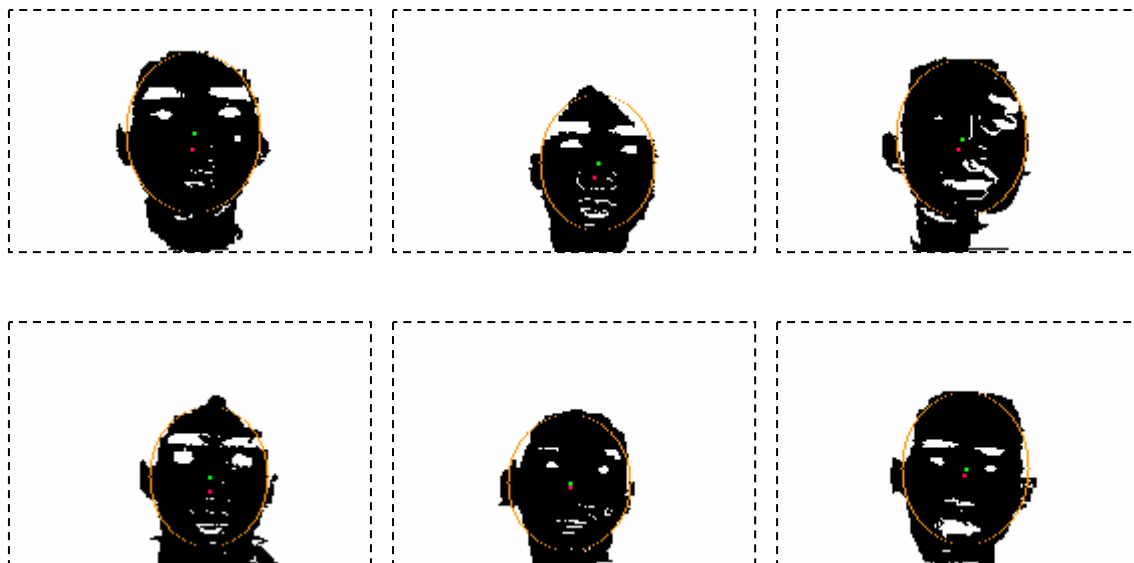


圖 3.9 不同臉型有不同比例的近似橢圓。