

國立臺北大學電機工程學系專題報告  
以最小生成樹進行頭殼辨識

Using the Minimum Spanning Tree Method for Solving the Skull Positioning Problem as  
Identity Recognition

組員：陳元弈、余梵農、邱柏儒

指導教授：何善輝、鄭穎仁 教授

零、摘要

針對問題的存在，無論是在頭殼定位，身份辨識，以及頭殼定位用於當作身分辨識??

本研究提出一個新的方法使用人頭的三維模型加上特徵點為一個網狀圖，以建構最小生成樹的方式作為身分辨識。首先我們使用一個基本的人頭模型，並且在正面及側面照片上透過影像處理的方式找出個關鍵點，透過這些關鍵點調整跟人頭模型上相對應的關鍵點，以建立與圖片相符的人頭模型。從調整過後的人頭模型中取出調整後關鍵點的網狀圖，選取其中一個關鍵點當 **root** 後，再用最小生成樹的生物計算平行處理的方法來解決關鍵點的網狀圖，藉由比較不同頭殼網狀圖的最小生成樹權重，我們可以判定做為身分辨識用。

在本專題中，我們利用 **prim** 平行處理演算法的分子運算來解決生成樹問題，並應用於身分辨識。首先我們架構所有可行的解空間，接著我們將不合理的路徑組合刪除後，在每個路徑組合的尾端加上各路徑所對應的成本並算出總成本。最後，在其中找到具有最低成本的樹為最佳解。

關鍵字：人臉辨識、3D 模型、最小生成樹、普里姆(**prim**)演算法、最短路徑、人頭模型

## 一、引言 (Introduction)

隨著網路與現今科技的普及化，越來越多的資訊安全及個人隱私也慢慢的被受重視。早期應用廣泛、相對比較傳統的身分辨識方法，例如：密碼、信用卡、鑰匙等辨識方法都容易被偽造，並且有遺失或遺忘的風險，進而在使用上造成不便。因此，近幾年學者與業界人士開始高度的重視類臉部辨識系統，並且深切期盼對於治安或門禁系統上能有突破性的貢獻。

以治安層面而言，可以在重要的公共場所，例如機場、賣場等地裝設攝影機，利用無線網路與當局連線，即時辨識是否有可疑人士出沒。以門禁系統而言，除了可以運用傳統的警衛、磁卡、密碼以外，還可以利用生物認證技術，如：視網膜、聲紋、指紋、或人像等資料作為識別的依據，其中則以人臉識別最為友善。

以廣義的人臉辨識來說，包括了人臉影像擷取、人臉偵測、人臉辨識的預處理、身分確認以及查詢身分等；而狹義的人臉辨識，則是指通過人臉進行身分確認或者查詢身分的技術。

本專題的目標係建立一類兼具快速、辨識度高的辨識方法。透過照片進行的人臉辨識可以有許多的用途，利用最小生成樹的演算法，讓辨識結果可以建檔，並存存於雲端系統，讓使用者可以在何時何地都有辦法進行分析。

本專題預期效益係為提供個快速、精準的人臉辨識方法，最終目標則為只需兩張不同角度的照片即可取點並建立 3D 模型同時獲得最小生成樹。

## 二、背景和文獻探討

生物辨識技術在近年來快速的發展，因為傳統的身分辨識如密碼、鑰匙等容易遭到複製，且會有被遺忘的可能性，而生物辨識使用生物特徵作為辨識依據，具有難以複製的特性，因此可大幅提升安全性與可信賴度。人臉辨識自古以來有許許多多種方式進行，早期以繪畫的方式來做相親或是用來指認通緝犯，近年來最為人所知的人臉辨識用途是美軍為緝捕恐怖分子賓拉登所使用的紅外線臉部辨識系統，而近年的保全系統也將臉部辨識系統應用於其中。

近年來美國賓州一個小鎮率先採用人類臉部辨識系統來擔任警衛的工作，但是成果卻不能盡如人意。雖然小鎮採用的系統在測試的時候效果極佳，臨場的實際效用卻未能發揮功用。推測其系統失靈的原因，應該是光線等外在因素使得資料庫中的照片和實際採樣的照片會有辨識誤差，導致系統的效能不如預期。

鑒於人類臉部辨識的警衛系統考量因素較為複雜，例如：必須快速辨識罪犯，導致設備的門檻較高，以及在辨識環境、角度、光線條件不佳時，仍需要擁有高辨識能力等因素。

人臉的辨識，主要是建立在臉部特徵的差異上，由於光線、角度、表情的不同，將會造成臉部特徵有很大的差異。尤其是光線，若其光反射強度不相同，將導致成像於電腦的圖形會跟著產生變化。

一般來說，人臉辨識的技術核心都包含三個步驟：人臉偵測 (Face Detection)、人臉追蹤 (Face Tracking)，與人臉辨識 (Face Identification)。想像一個旅行者在叢林中由遠而近向你接近，如果你要確定他是否為你認識的人，你的腦與眼開始進行如下動作：先找到此人並希望看清他的臉，此即為人臉偵測 (Face Detection)；若此人被找到時，可能因距離太遠或叢林中陰影太重讓你無法看清，你可能會注視著他，並且目光隨之而動，此即為人臉追蹤 (Face

Tracking);待你看清此人，經腦中記憶比對，你確定他是你的鄰居，此即為人臉辨識(Face Identification)。由此可知，若要能進行人臉辨識，必須先要有記憶，人腦如此，電腦也如此。

先前提到人臉辨識的相關研究愈來愈多，進行人臉辨識研究的過程，需要大量的人臉影像，人臉資料庫也發展出許多種類。以下就列舉一些人臉資料庫的資訊以供參考：

### 2.1 AT&T 人臉資料庫 (ORL 人臉資料庫)

這個資料庫是由 AT & T Laboratories Cambridge (劍橋實驗室) 於 1992 年 4 月到 1994 年 4 月，兩年間在實驗室中收集的。為了劍橋大學工程系(Cambridge University Engineering Department)中的 Speech、Vision and Robotics Group 進行有關人臉辨識的研究而建立的。目前已成為人臉辨識領域中，最普遍、也是最具代表性的人臉資料庫之一。資料庫中共有 400 張影像，包含了 40 個不同的人，且每個人有 10 張影像，都是在不同時間、燈光、表情下所拍攝的。而拍攝的背景固定為黑色，且人臉皆為正面的（僅允許一點小角度的偏差）。所有原始圖檔皆以 256 色灰階，使用 PGM 檔的形式儲存，其大小為 92x112 像素。以數張影像作比較，確實可以提升在辨識時的精確性，然而只有正面，在其他角度做辨識時，使否能準確套要辨識的資料則有待商榷，而且一個人就使用了 10 張影像，對於資料的儲存實為一項負擔。詳細資料可參考如網址：

<http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html>。

## 2.2 Yale 人臉資料庫

耶魯大學所蒐集的人臉資料庫包含了兩個一個為 Yale 人臉資料庫以及後來蒐集的 Yale 人臉資料庫 B。兩個資料庫均可免費使用於學術用途上，但須註明其來源出處。Yale 資料庫中包含 15 個不同的人，每人 11 張照片，共有 165 張影像。每個人的 11 張影像各有不同的情緒與光照條件，分別有正面打光、戴眼鏡或不戴眼鏡、快樂的表情、左邊打光、正常表情、右邊打光、傷心的表情、想睡覺的表情、驚訝的表情和眨眼表情。資料十分詳細，在辨識依據上大大提升了可靠度，可惜儲存的資料比 AT&T 更加龐大，負擔更重，且人臉辨識的變數多這項問題還是沒有解決。詳細資料可參考如網址：

<http://cvc.yale.edu/projects/yalefaces/yalefaces.html>。

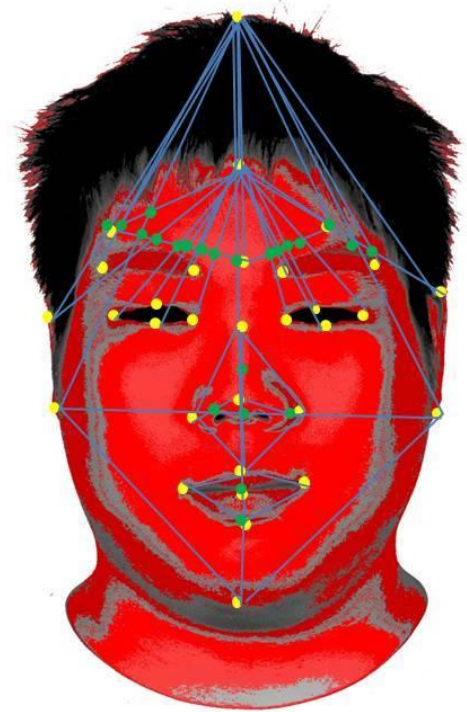
## 2.3 INDIAN 人臉資料庫

此資料庫於 2002 年在 Indian Institute of Technology Kanpur 校區內蒐集的，目前所引用的文獻也差不多有 50 篇左右。其中資料庫中包含了男性 39 人與女性 22 人，每人各有 11 張影像各有五種不同的角度及四種表情，分別為向前看、向左看、向右看、向上看、朝左上方看、朝右上方看、向下看，及面無表情、微笑的狀態、大笑的樣子、難過或厭惡的表情。同 Yale 人臉資料庫，資料十分詳細，在辨識依據上大大提升了可靠度，可惜儲存的資料比 AT&T 更加龐大，負擔更重，且人臉辨識的變數多這項問題還是沒有解決。詳細資料可參考如網址：<http://vis-www.cs.umass.edu/~vidit/IndianFaceDatabase>。

### 三、問題（Problems）

#### 3.1 頭殼定位問題

我們進行的頭殼定位目標在找出頭殼的特徵點，如何取得，並獲得頭殼的特徵點並以其進行運算為一大問題。



#### 3.2 身份辨識問題

人臉單純分辨出不同人不算困難，但是要篩選出特定人物十分艱鉅，尤其是並不一定有辦法取得正面，角度正確的照片，如何特定臉部是個重要的實行重點。

## 四、解決方法(Themes)

### 4.1 最小生成樹

最小生成樹是一副連通加權無向圖中一棵權值最小的生成樹。在一給定的無向圖  $G = (V, E)$  中， $(u, v)$  代表連接頂點  $u$  與頂點  $v$  的邊（即  $(u, v) \in E$ ），而  $w(u, v)$  代表此邊的權重，若存在  $T$  為  $E$  的子集（即  $T \subset E$ ）且為無循環圖，使得的

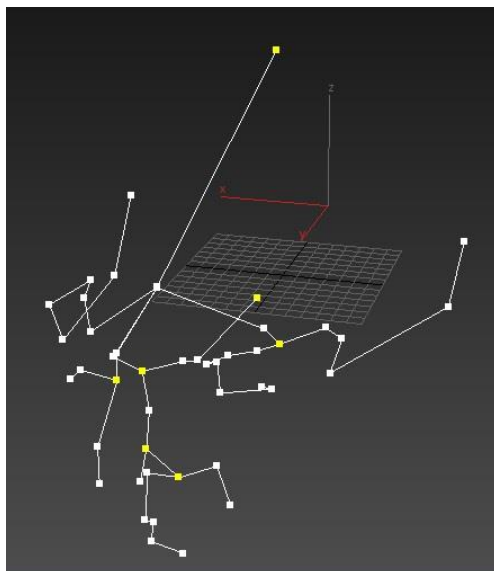
$$w(T) = \sum_{(u,v) \in T} w(u, v)$$

$w(T)$  最小，則此  $T$  為  $G$  的最小生成樹。

最小生成樹其實是最小權重生成樹的簡稱。一個連通圖可能有多個生成樹。當圖中的邊具有權值時，總會有一個生成樹的邊的權值之和小於或者等於其它生成樹的邊的權值之和。

廣義上而言，對於非連通無向圖來說，它的每一連通分量同樣有最小生成樹。以有線電視電纜的架設為例，若只能沿著街道布線，則以街道為邊，而路口為頂點，其中必然有一最小生成樹能使布線成本最低。簡單點說有幾個城市你要設計一個路線，這個路線能走完所有的這幾個城市，且路程最短，這個路線就是最小生成樹的含義。

計算稠密圖的最小生成樹最早是由羅伯特·普里姆在 1957 年發明的，即 Prim 演算法。



### 5.1. Prim 演算法

Prim 演算法的每一步都會為一棵生長中的樹添加一條邊，該樹最開始只有一個頂點，然後會添加  $V - 1$  個邊。每次總是添加生長中的樹和樹中除該生長的樹以外的部碎形成的切分的具有最小權值的橫切邊。

Prim 演算法的時間複雜度為  $O(E + V \log V)$

### 4.2 頭殼定位方法

透過兩張照片的組合以及比對，建立 3D 模型用於比較

#### 4.2.1 特徵點

人的五官是人臉上最易於表現人的特徵的部位，透過在相對位置取得特徵點可以將部位位置特徵化，方便計算，儲存

#### 4.2.2 相片不同角度探討

目前使用的相片如非正面側面之照片將會難以進行模型的建立，角度不同必須進行矯正，而矯正的方式為特定長度來計算角度，得出角度後矯正即可獲得正確的數據。

### 4.3 用最小生成樹解頭殼定位

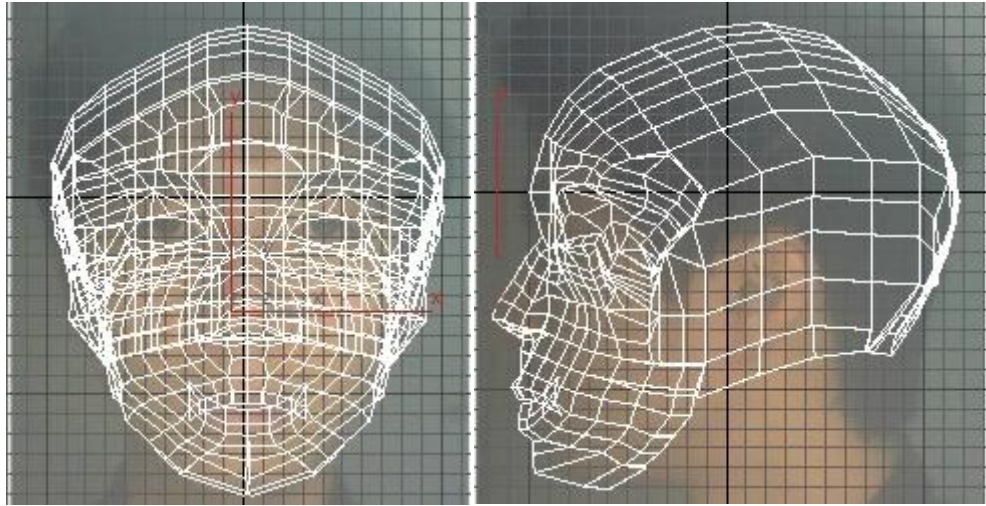
取得特徵點之後，將數據進行最小生成樹之運算，可獲取目標之特徵函數以及最小生成樹，並可以此來特定目標。

## 五、實驗和結果(Experimental Results)

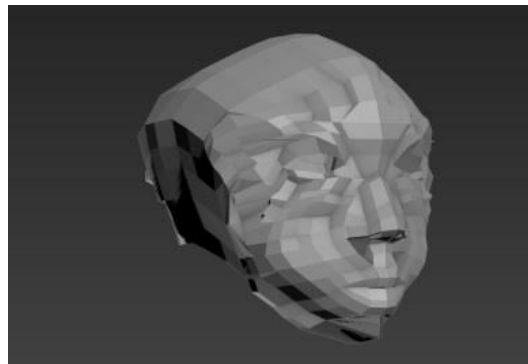
### 5.1. 初步架構：

先以兩張相片建立人頭 3D 圖，傳統的構想是將人頭的 3D 圖建檔作成資料庫，之後再將其套用到其他角度的照片以辨識

身分，要建立一個頭部 3D 圖，需要「正面」以及「側面」兩張相片，並一點一點將 3D 的「面」覆蓋到相片上：



於正面包裹 3D 皮膚，側面調整傾斜角度，最後拷貝出頭型：

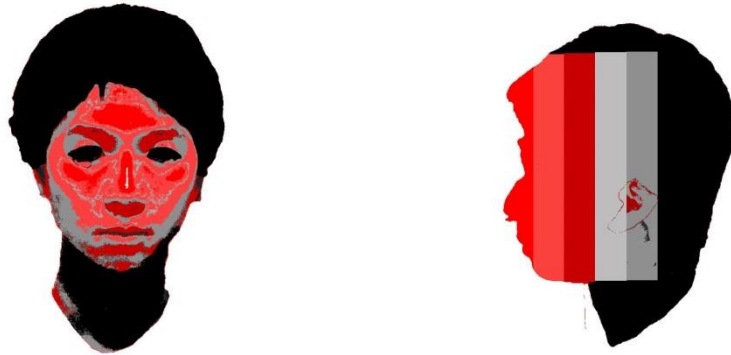


然而，這種做法相當耗時耗力，而且須對人頭比例有一定的掌握程度，否則就會造成如上圖臉後面頭顱整個變形的情況，另外，這種做法，對做為樣本的照片要求也很高，幾乎需要專業的設備才能把頭部正面跟側面的數個特徵對齊，不然，可能會有做到一半發現鼻尖對不上，眼角位置錯開等等，造成模型製作的困難。

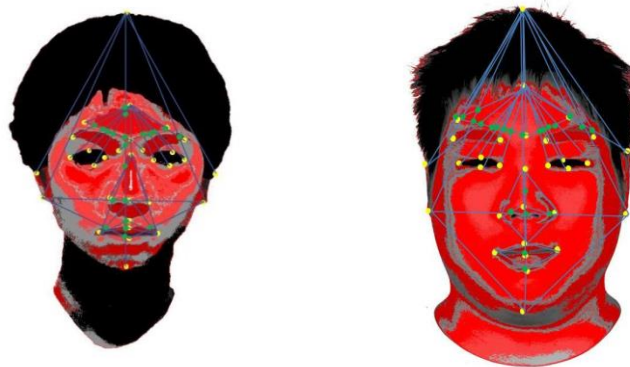
## 5.2. 中期架構：特徵點

有鑑於傳統方式效率太差，因此我們開始思考「有沒有一種方式，只需幾個特徵作為辨識依據，就能有效的辨識出個人身分呢？」於是我們想出一種構想－「特徵點」。我們在人頭的照片中，取出幾個作為關鍵的點，當作存入頭殼資料庫的根據。

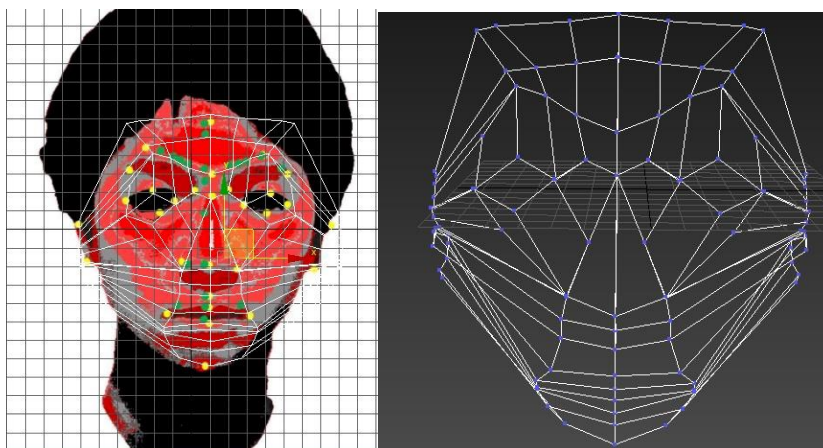
然而，取特徵點不能無憑無據，於是我們將頭部依照凹凸深度的不同，劃分出五個顏色：

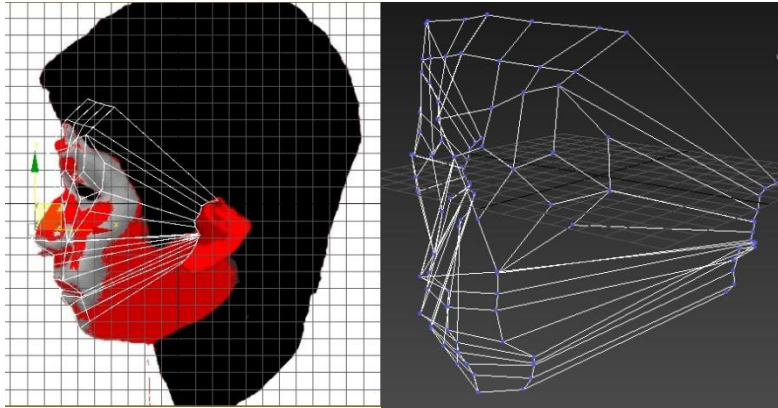


接下來，在顏色變化有明顯落差的交界處取特徵點，於此可判讀出人頭最明顯的部位：

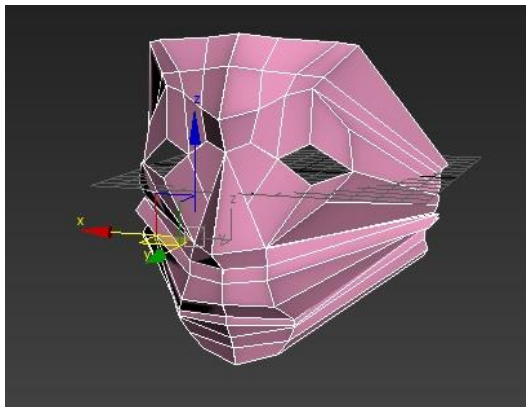


再依特徵點編輯人頭的 3D 圖：





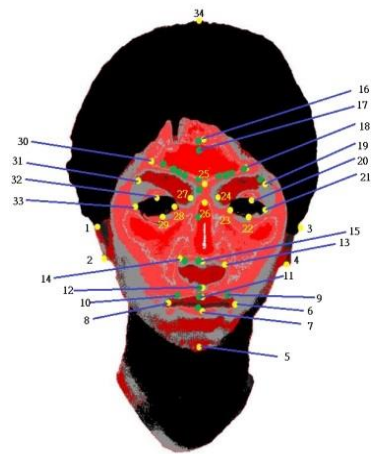
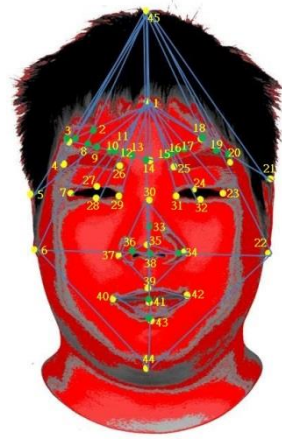
最後，完成一個較簡略，但不失特徵的頭部模型：



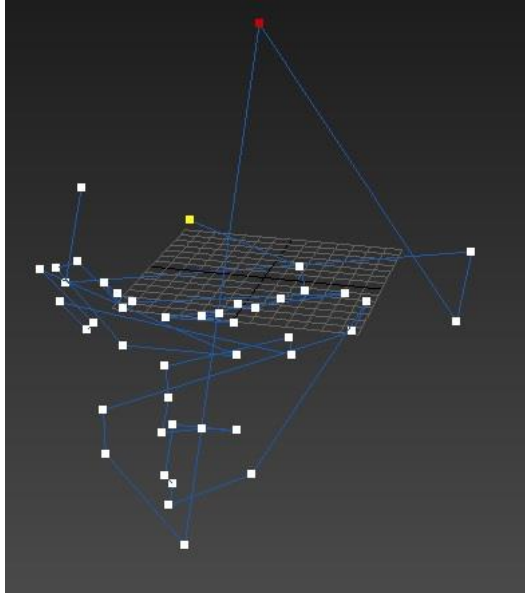
### 5.3. 後期：最小生成樹:

有了特徵點，是否還有更簡便的方式儲存人頭資料呢？用面來製作，失真的程度還是有的，如果不侷限在一定要符合頭部的形狀，那麼，應可用「線」來試著製作頭部的資料，既然每個人的頭部都是獨一無二的（即使是同卵雙胞胎），那麼特徵點理論上也是，那麼，應用資料庫的理論，訪問每個特徵點的方式也會是獨一無二的。

因此綜合以上結論，我們採用「最小生成樹」這個方式，來架構頭殼的資料與模型。



取得特徵點後，透過 3Ds MAX 連接每個點並取得其座標：



為了方便讀取，為每個點加上編號：

| 特徵點 | X        | Y       | Z       |  |      |     |     |
|-----|----------|---------|---------|--|------|-----|-----|
| 1   | 3.11     | 130.714 | 149.261 |  | 3    | 131 | 149 |
| 2   | -52      | 132.927 | 60      |  | -52  | 133 | 60  |
| 3   | -76.889  | 143.665 | 52.889  |  | -77  | 144 | 53  |
| 4   | -83.111  | 117.974 | 37.778  |  | -83  | 118 | 38  |
| 5   | -116     | -28.942 | 17.333  |  | -116 | -29 | 17  |
| 6   | -112.444 | -9.759  | -17.333 |  | -112 | -10 | -17 |
| 7   | -76.889  | 113.412 | 19.111  |  | -77  | 113 | 19  |
| 8   | -58.222  | 143.719 | 51.111  |  | -58  | 144 | 51  |
| 9   | -47.556  | 153.228 | 49.333  |  | -48  | 153 | 49  |
| 10  | -36      | 159.566 | 46.667  |  | -36  | 160 | 47  |
| 11  | -28      | 163.716 | 46.667  |  | -28  | 164 | 47  |
| 12  | -20.889  | 166.04  | 45.778  |  | -21  | 166 | 46  |
| 13  | -12      | 169.075 | 42.222  |  | -12  | 169 | 42  |
| 14  | 5.778    | 170.79  | 41.333  |  | 6    | 171 | 41  |
| 15  | 23.556   | 162.047 | 44.889  |  | 24   | 162 | 45  |
| 16  | 34.222   | 158.69  | 47.556  |  | 34   | 159 | 48  |
| 17  | 44.889   | 152.971 | 50.222  |  | 45   | 153 | 50  |
| 18  | 59.871   | 143.642 | 55.334  |  | 60   | 144 | 55  |

透過計算座標點間的最小生成樹，透過計算最小生成樹的方式：

$$\begin{aligned}
 |P_1P_2|^2 &= |P_1A|^2 + |AB|^2 + |BP_2|^2 \\
 &= |x_2 - x_1|^2 + |y_2 - y_1|^2 + |z_2 - z_1|^2 \\
 &= (x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2 \\
 |P_1P_2| &= \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}
 \end{aligned}$$

```

for (x=1;x<=k;x++) {
    for (i=0;i<=k-x-1;i++) {
        a=co[x+1+i][y]-co[x][y];
        b=co[x+1+i][y+1]-co[x][y+1];
        c=co[x+1+i][y+2]-co[x][y+2];
        r=a*a+b*b+c*c;

        fprintf(fPtr,"第%d點到第%d點%f\n",x,x+1+i,sqrt(r));
    }
}

```

以此獲取以下數據：

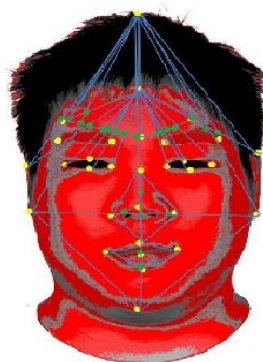
|    | 1       | 2       | 3       | 4       | 5       | 6       | 7       | 8       | 9       | 10      | 11      | 12      | 13      | 14      | 15      | 16      | 17      | 18      |
|----|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| 1  | 0       | 44.317  | 145.003 | 145.925 | 134.172 | 139.632 | 131.412 | 110.892 | 139.345 | 116.069 | 131.126 | 128.176 | 127.648 | 109.426 | 135.473 | 122.577 | 125.575 | 136.301 |
| 2  | 44.317  | 0       | 145.623 | 133.004 | 97.9694 | 110.82  | 99.5038 | 75.3724 | 111.969 | 84.3564 | 101.814 | 101.415 | 106.555 | 85.9884 | 112.361 | 131.183 | 128.503 | 137.047 |
| 3  | 145.003 | 145.623 | 0       | 44.317  | 134.097 | 110.585 | 131.252 | 139.746 | 116.52  | 138.687 | 130.927 | 127.973 | 109.133 | 127.632 | 135.185 | 122.528 | 125.431 | 96.3639 |
| 4  | 145.925 | 133.004 | 44.317  | 0       | 98.0306 | 75.1332 | 99.4535 | 111.108 | 84.9058 | 111.49  | 101.715 | 101.316 | 85.8021 | 106.686 | 112.156 | 131.259 | 128.487 | 101.44  |
| 5  | 134.172 | 97.9694 | 134.097 | 98.0306 | 0       | 31.4166 | 22.8254 | 31.4166 | 36.6197 | 36.3593 | 34      | 42.1545 | 54.5711 | 54.5711 | 59.5399 | 137.058 | 125.256 | 120.399 |
| 6  | 139.632 | 110.82  | 110.585 | 75.1332 | 31.4166 | 0       | 27.7489 | 50      | 15.6844 | 47.1699 | 32.1714 | 36.2491 | 35.3977 | 52.469  | 51.1273 | 120.657 | 109.279 | 97.2985 |
| 7  | 131.412 | 99.5038 | 131.252 | 99.4535 | 22.8254 | 27.7489 | 0       | 27.7489 | 21.9545 | 22.0227 | 11.1803 | 19.6469 | 35.5668 | 35.5668 | 36.7151 | 117.958 | 105.043 | 102.299 |
| 8  | 110.892 | 75.3724 | 139.746 | 111.108 | 31.4166 | 50      | 27.7489 | 0       | 47.392  | 15      | 32.1714 | 36.2491 | 52.469  | 35.3977 | 51.1273 | 120.657 | 109.279 | 112.548 |
| 9  | 139.345 | 111.969 | 116.52  | 84.9058 | 36.6197 | 15.6844 | 21.9545 | 47.392  | 0       | 40.0125 | 20.9045 | 23.1517 | 23.7276 | 41.9881 | 35.9166 | 110.725 | 98.0816 | 88.9326 |
| 10 | 116.069 | 84.3564 | 138.687 | 111.49  | 36.3593 | 47.1699 | 22.0227 | 15      | 40.0125 | 0       | 21.2132 | 23.4307 | 41.9285 | 23.622  | 36.4555 | 110.603 | 98.0663 | 102.186 |
| 11 | 131.126 | 101.814 | 130.927 | 101.715 | 34      | 32.1714 | 11.1803 | 32.1714 | 20.9045 | 21.2132 | 0       | 9       | 28.1069 | 28.1069 | 25.5539 | 108.853 | 95.3362 | 93.9149 |
| 12 | 128.176 | 101.415 | 127.973 | 101.316 | 42.1545 | 36.2491 | 19.6469 | 36.2491 | 23.1517 | 23.4307 | 9       | 0       | 21.7945 | 21.7945 | 18.3848 | 100.02  | 86.3713 | 85.7496 |
| 13 | 127.648 | 106.555 | 109.133 | 85.8021 | 54.5711 | 35.3977 | 35.5668 | 52.469  | 23.7276 | 41.9285 | 28.1069 | 21.7945 | 0       | 30      | 26.6271 | 87.0115 | 74.5319 | 67.1267 |
| 14 | 109.426 | 85.9884 | 127.632 | 106.686 | 54.5711 | 52.469  | 35.5668 | 35.3977 | 41.9881 | 23.622  | 28.1069 | 21.7945 | 30      | 0       | 26.6271 | 87.0115 | 74.5319 | 80.1623 |
| 15 | 135.473 | 112.361 | 135.185 | 112.156 | 59.5399 | 51.1273 | 36.7151 | 51.1273 | 35.9166 | 36.4555 | 25.5539 | 18.3848 | 26.6271 | 26.6271 | 0       | 91.1811 | 75.9605 | 78.7972 |
| 16 | 122.577 | 131.183 | 122.528 | 131.259 | 137.058 | 120.657 | 117.958 | 120.657 | 110.725 | 110.603 | 108.853 | 100.02  | 87.0115 | 87.0115 | 91.1811 | 0       | 16.9706 | 38.2753 |
| 17 | 125.575 | 128.503 | 125.431 | 128.487 | 125.256 | 109.279 | 105.043 | 109.279 | 98.0816 | 98.0663 | 95.3362 | 86.3713 | 74.5319 | 74.5319 | 75.9605 | 16.9706 | 0       | 35.3412 |
| 18 | 136.301 | 137.047 | 96.3639 | 101.44  | 120.399 | 97.2985 | 102.299 | 112.548 | 88.9326 | 102.186 | 93.9149 | 85.7496 | 67.1267 | 80.1623 | 78.7972 | 38.2753 | 35.3412 | 0       |

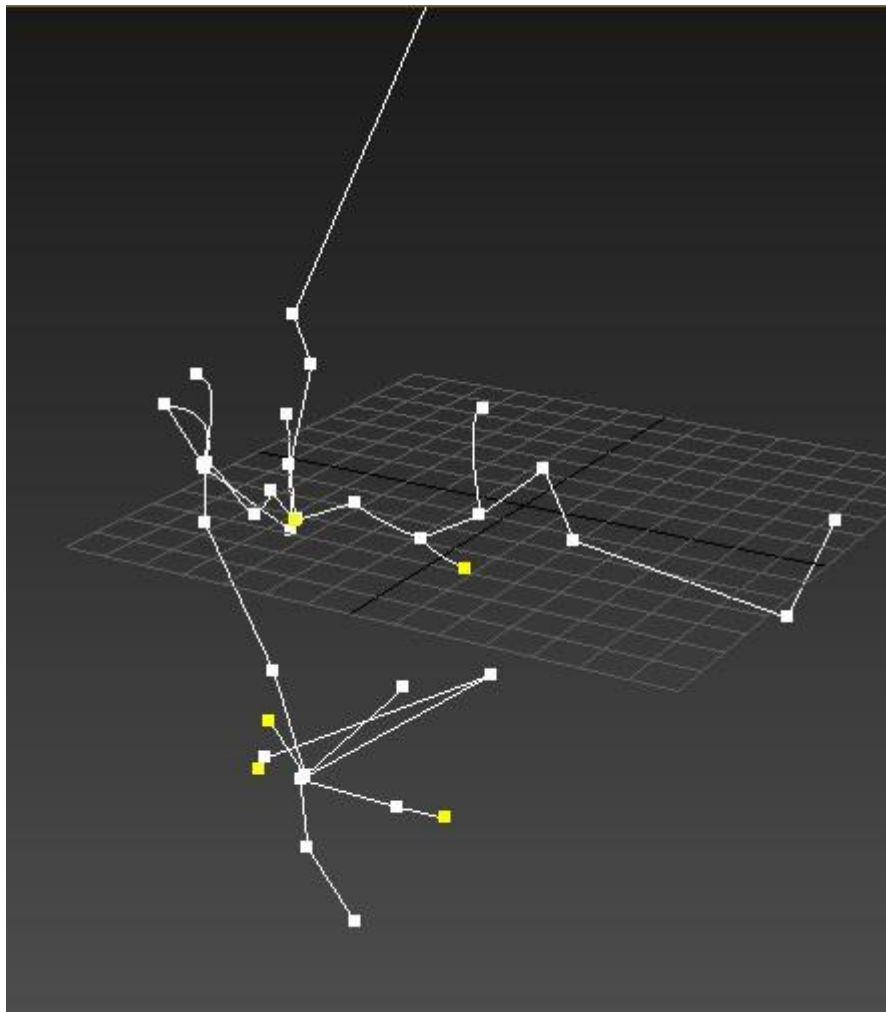
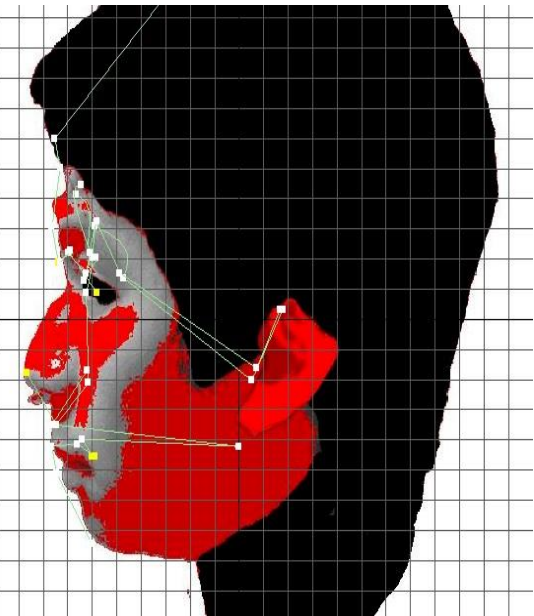
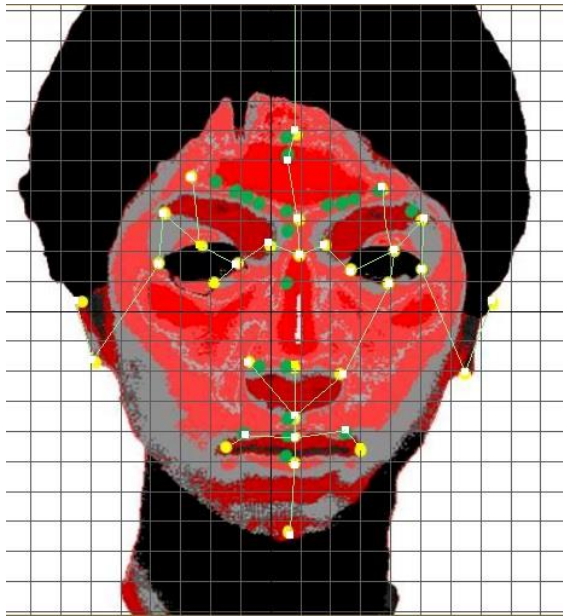
透過此資料我們建立最小生成樹並以此作為人臉辨識的依據：

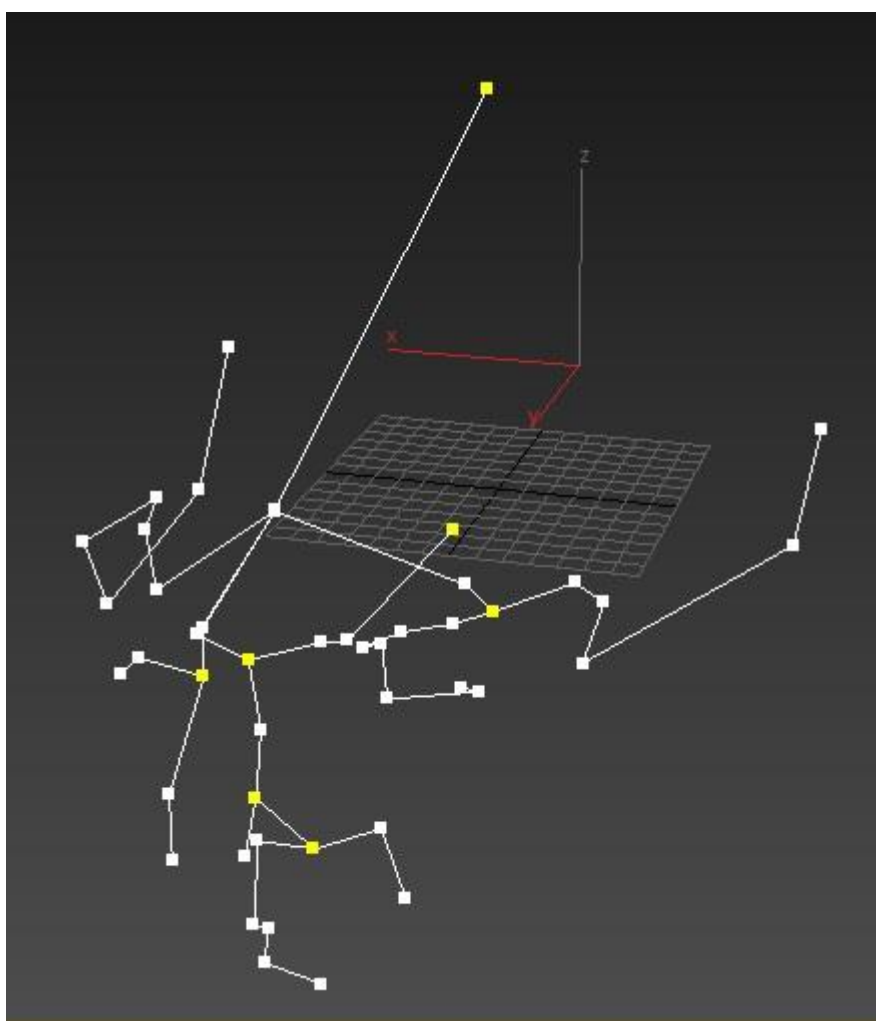
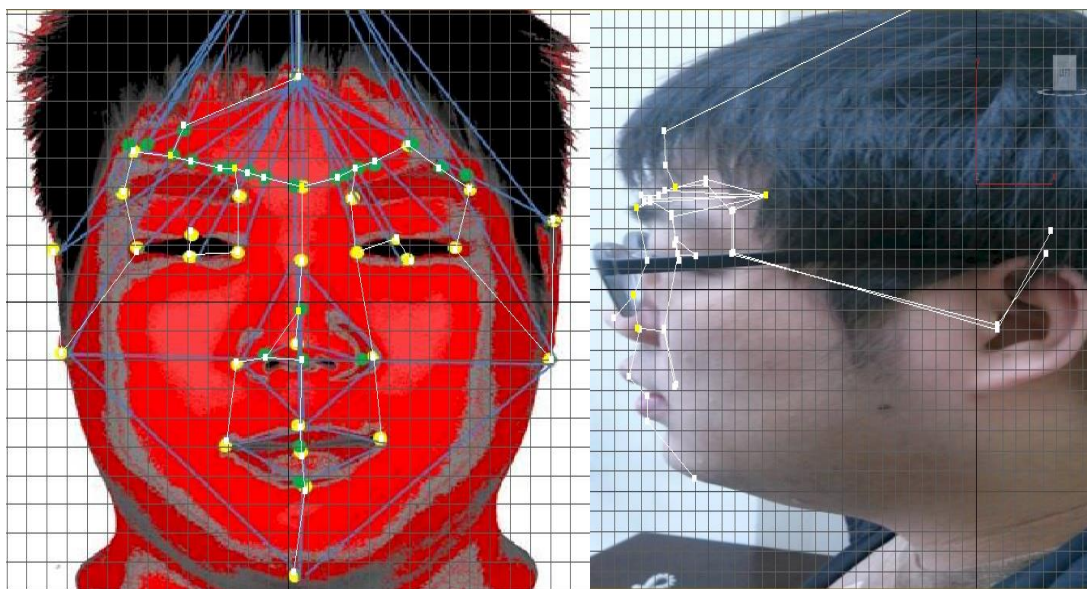
```
1 --> 2
2 --> 33
33 --> 31
31 --> 32
32 --> 28
28 --> 27
27 --> 26
26 --> 24
24 --> 23
23 --> 20
20 --> 25
25 --> 22
22 --> 13
13 --> 12
12 --> 11
11 --> 7
7 --> 9
9 --> 6
6 --> 10
10 --> 8
8 --> 14
14 --> 5
5 --> 4
4 --> 3
3 --> 16
16 --> 34
```



```
1 --> 2
2 --> 8
8 --> 9
9 --> 10
10 --> 11
11 --> 12
12 --> 13
13 --> 26
26 --> 14
14 --> 3
3 --> 15
15 --> 16
16 --> 17
17 --> 25
25 --> 18
18 --> 19
19 --> 31
31 --> 29
29 --> 20
20 --> 23
23 --> 24
24 --> 32
32 --> 30
30 --> 33
33 --> 35
35 --> 36
36 --> 37
37 --> 38
38 --> 28
28 --> 27
27 --> 39
39 --> 41
41 --> 43
43 --> 40
40 --> 4
4 --> 7
7 --> 44
44 --> 34
34 --> 42
42 --> 1
1 --> 45
```







同時為驗證，使用不同人臉不同角度進行對比準，卻展現能明顯分辨不同人臉之特徵。

## 六、主要成果與評估

### 1、結語

美軍抓捕賓拉登即透過人臉辨識進而鎖定目標，並將其逮捕歸案，本套系統期望能發揮相近的功能。

透過最小生成樹，已能夠確實分析人臉之差距，並能做出簡單的對象分別，而目前最需要改進的重點為分析速度以及特徵點取得自動化、標準化，同時未來希望能夠追加允許角度不必要求 90 度，不同時間不同地點拍攝之照片也能夠進行分析、辨識。

## 七、結語與展望

本系統透過取得照片獲得基本資料，分析後取得的資料，將其彙整成辨識系統的一套標準，相較於之前要先將人頭捏出來的方式，的確提升了不少效率，面部辨識系統除了應用在軍事上，居家生活的保全系統亦可涉足，相較數位密碼，這種生物性的密碼更難破解，安全品質將有一定程度的提升，我們期待，當這套系統完善後，能造福世上的許多人。