

自動頭髮去背系統

學生：郭佳甯 指導教授：楊傳凱

國立台灣科技大學資訊管理學系

摘要

髮型預覽系統能在理髮之前就看到理髮後的樣子，能使人更容易找到適合自己的髮型，但是髮型預覽系統需要大量的髮型資料庫。我們的自動頭髮去背可以輸入一張人臉圖片，自動對圖片作去背後只留下頭髮，故使用我們的系統即可迅速的產生髮型資料庫。

我們提出一個自動頭髮去背演算法的架構，其演算法流程是繪製頭髮線、臉部線、邊界線，再作頭部去背和臉部去背，並利用兩張去背的 alpha 建立 alpha 空間，找到品質較好的 trimap，再使用 trimap 對原圖作去背得到結果。我們另外提出一個良好的評價函數:alpha 相似度，用來評量實驗結果，也在演算法中使用。我們的 ground truth 是以手動的方式對原圖畫 trimap 產生品質最好的去背 alpha，實驗結果有 86.2% 的相似度。

An Automatic Hair Matting System

Student:Chia-Ning kuo Advisor:Chuan-Kai Yang

Department of Information Management

National Taiwan University of Science and Technology

Abstract

Automatic hair extraction from a given 2D image has been a challenging problem for a long time, especially when complex backgrounds and a wide variety of hairstyles are involved. This paper has made its contribution in the following two aspects. First, it proposes a novel framework that successfully combines the techniques of face recognition, outlier-aware initial stroke placement and matting to extract the desired hairstyle from an input image. Second, it defines a new comparison metric that is well suited for the alpha matte comparison. Our results show that, compared with the manually drawn trimaps for hair extraction, the proposed automatic algorithm can achieve about 86.2% extraction accuracy.

誌謝

感謝我的指導教授楊傳凱老師在各方面的協助和支持和老師對我們的愛，讓我這兩年得到很大的成長。口試委員鮑興國老師和林伯慎老師的耐心聆聽和寶貴的建議，讓我了解到自己的不足，最後順利完成論文和口試。

感謝實驗室的同學們：幸卿、佑鈞、彥秀、錦芬、家愷、昱志、青揚、士賢、君豪、雅軒、琬潔、美杏、柏諺、孟庭、家仔。我在實驗室的生活是快樂的，就是因為我有一群可愛的同學們。

我感到很慶幸，因為我進了一個最棒的實驗室。



目錄

1.	簡介.....	1
2.	文獻探討.....	3
2.1.	人臉偵測.....	3
2.2.	離群值偵測.....	3
2.3.	去背.....	4
2.4.	自動頭髮去背.....	5
3.	演算法流程.....	7
3.1.	筆畫繪製.....	8
3.1.1.	人臉偵測.....	8
3.1.2.	臉部線繪製.....	9
3.1.3.	邊界線繪製.....	10
3.1.4.	頭髮線繪製.....	10
3.1.5.	頭髮線微調.....	11
3.2.	Trimap 繪製.....	15
3.2.1.	頭部去背.....	15
3.2.2.	臉部去背.....	16
3.2.3.	Alpha 空間.....	16
3.2.4.	產生 Trimap.....	18
4.	實驗結果與討論.....	21
4.1.	實驗結果統計.....	22
4.2.	Alpha difference 與最終 Alpha 的關係.....	29
4.3.	DB outlier 的參數比較：.....	31
4.4.	與其他 paper 比較：.....	34
4.5.	演算法時間效能分析：.....	38
4.6.	系統限制.....	39
5.	結論與未來展望.....	42
	參考文獻.....	43
	附錄.....	45

圖目錄

圖 3-1 系統流程示意圖	7
圖 3-2 ASM 特徵點示意圖	9
圖 3-3 頭髮線示意圖	10
圖 3-4 頭髮線 alpha 示意圖	11
圖 3-5 alpha 平均示意圖	13
圖 3-6 模糊後的 alpha 平均和調整後的頭髮點位置	13
圖 3-7 頭髮點可移動範圍 HM 示意圖	14
圖 3-8 頭髮去背 trimap 和 alpha.....	15
圖 3-9 頭部去背 trimap 和 alpha.....	16
圖 3-10 臉部去背 trimap 和 alpha.....	16
圖 3-11 alpha 空間示意圖	17
圖 3-12 最終 trimap 的前景	18
圖 3-13 最終 trimap 的背景	18
圖 3-14 眼睛區域示意圖	19
圖 3-15 最終 Trimap.....	19
圖 3-16 最終 alpha.....	19
圖 4-1 CostAverage 品質最好前 10 張以 MSE 評價	23
圖 4-2 CostAverage 品質最好前 10 張以 SAD 評價	23
圖 4-3 CostAverage 品質最好前 10 張以 Alpha 相似度評價	23
圖 4-4 CostAverage 品質最差前 10 張以 MSE 評價.....	24
圖 4-5 CostAverage 品質最差前 10 張以 SAD 評價.....	24
圖 4-6 CostAverage 品質最差前 10 張以 Alpha 相似度評價.....	24
圖 4-7 CostMin 品質最好前 10 張以 MSE 評價	25
圖 4-8 CostMin 品質最好前 10 張以 SAD 評價	25
圖 4-9 CostMin 品質最好前 10 張以 Alpha 相似度評價	25
圖 4-10 CostMin 品質最差前 10 張以 MSE 評價	26
圖 4-11 CostMin 品質最差前 10 張以 SAD 評價	26
圖 4-12 CostMin 品質最差前 10 張以 Alpha 相似度評價	26
圖 4-13 CostSum 品質最好前 10 張以 MSE 評價	27
圖 4-14 CostSum 品質最好前 10 張以 SAD 評價	27
圖 4-15 CostSum 品質最好前 10 張以 Alpha 相似度評價	27
圖 4-16 CostSum 品質最差前 10 張以 MSE 評價	28
圖 4-17 CostSum 品質最差前 10 張以 SAD 評價	28
圖 4-18 CostSum 品質最差前 10 張以 Alpha 相似度評價	28
圖 4-19 圖片編號 49 的人臉偵測結果	32
圖 4-20 圖片編號 49 使用 DB outlier 參數 $r=0.7$ 的最終 alpha.....	32

圖 4-21 圖片編號 49 使用 DB outlier 參數 $r=0.5$ 的最終 alpha.....	32
圖 4-22 與 paper1 比較-我們的臉部去背 trimap.....	34
圖 4-23 與 paper1 比較-我們的最終 trimap.....	34
圖 4-24 與 paper1 比較-我們的最終結果.....	34
圖 4-25 與 paper1 比較-我們的最終 alpha.....	34
圖 4-26 與 paper1 比較-paper1 的最終 alpha.....	35
圖 4-27 與 paper1 比較-原圖裁剪後我們的臉部去背 trimap.....	35
圖 4-28 與 paper1 比較-原圖裁剪後我們的最終 trimap.....	35
圖 4-29 與 paper1 比較-原圖裁剪後我們的最終結果.....	36
圖 4-30 與 paper1 比較-原圖裁剪後我們的最終 alpha.....	36
圖 4-31 與 paper2 比較-臉部去背 trimap.....	36
圖 4-32 與 paper2 比較-最終 trimap.....	37
圖 4-33 與 paper2 比較-最終結果.....	37
圖 4-34 與 paper2 比較-最終 alpha.....	37
圖 4-35 與 paper2 比較-paper2 的 hair mask.....	38
圖 4-36 人臉偵測失敗.....	39
圖 4-37 Closed-form matting alpha 破碎 case1-頭髮去背 trimap....	39
圖 4-38 Closed-form matting alpha 破碎 case1-頭髮去背 alpha.....	40
圖 4-39 Closed-form matting alpha 破碎 case2-頭部去背 trimap....	40
圖 4-40 Closed-form matting alpha 破碎 case2-頭部去背 alpha.....	40
圖 4-41 背景有小色塊與頭髮相連-頭髮去背 trimap.....	41
圖 4-42 背景有小色塊與頭髮相連-頭髮去背 alpha.....	41

表目錄

表 3-1 頭髮線 alpha 相似度矩陣	12
表 4-1 Alpha difference 的評價	22
表 4-2 Final Alpha 的評價	22
表 4-3 Alpha difference 與最終 Alpha 的關係 - CostAverage	29
表 4-4 Alpha difference 與最終 Alpha 的關係 - CostMin	29
表 4-5 Alpha difference 與最終 Alpha 的關係 - CostSum	30
表 4-6 DB outlier 參數 $r=0.7, n=6$ 得到的 final Alpha 統計	31
表 4-7 DB outlier 參數 $r=0.5, n=6$ 得到的 final Alpha 統計	31
表 4-8 表格相減： $r=0.5$ 的表格減掉 $r=0.7$ 的表格	31
表 4-9 圖片編號 46 相似度矩陣沉澱圖	33
表 4-10 極端的例子相似度矩陣沉澱圖	33



1. 簡介

現代人注重外表，髮型對於外表的影響力很大，而美髮卻是個短時間內不可逆的過程。有些髮型在別人身上很好看，但不代表在自己身上很好看，所以大部分人多多少少都害怕嘗試新髮型，就是怕髮型不適合自己，或是理髮後的樣子不如預期。問題是出自於目前的美髮業是不可預覽結果的。從 2008 年到現在陸續出現了一些髮型網站，提供髮型預覽系統能讓使用者可以上傳自己的照片，再從髮型資料庫中選取頭髮，套到自己的頭上，藉由這些系統找到適合自己的髮型。現在只要在 google 搜尋 virtual hairstyle online 就能找到很多這種網站。在此我們看到了美髮業界對於髮型預覽系統的迫切需求，而髮型預覽系統的前置作業為髮型資料庫。世界上髮型千萬種，使用人工手動一張張去背擷取出頭髮是成本高而且耗費人力的工作，如何自動產生髮型資料庫就成為一個問題。

具體而言我們面臨的問題是對一張含有單一人臉的圖片，自動產生高品質的 trimap 並套用去背(Matting)演算法，擷取出圖片中的頭髮。其中的困難點在於各式各樣的髮型千奇百怪，我們不能預設頭髮長度、頭髮形狀、顏色，唯一的線索只有頭髮與頭部相鄰。在線索極少的情況下，如何正確擷取出頭髮就是其困難之處。

我們的演算法流程是輸入圖片，對圖片作人臉偵測，計算頭髮線、臉部線、和邊界線，使用這三條線去計算頭部去背和臉部去背，再利用頭部去背和臉部去背的結果及使用 alpha 空間的技術，找到品質較好的 trimap，最終使用上述產生的 trimap 對原圖作去背得到結果。其中，計算頭髮線的過程中使用了離群值偵測(outlier detection)和動態規劃等方法。

我們也作了一連串的實驗結果分析，像是各項參數對於結果的影響、與其他篇論文結果的比較，討論去背失敗的情形，演算法時間效能分析等等。為了評量實驗結果，我們使用一組 ground truth，是對每張測試圖片以手動畫 trimap 的

方式所產生最佳的去背結果。

我們在自動頭髮去背這方面提出了一個完整的架構。也提出良好的評價函數:alpha 相似度，可用來估計兩張 alpha 之間的相似度，並使用 alpha 相似度來評量實驗結果，而我們的實驗結果經統計後有 86.2%的相似度。另外我們也提出了 alpha 空間的概念，拿兩張 alpha 當作基底向量張出一個向量空間，使我們可以在此空間中尋找品質較好的 alpha。

接下來的章節分別是第 2 章文獻探討，介紹人臉偵測、離群值偵測、去背和自動頭髮去背的相關論文。第 3 章為演算法流程，詳細介紹演算法的每一個步驟。第 4 章為實驗結果與討論，最後第 5 章為結論與未來展望。



2. 文獻探討

2.1. 人臉偵測

人臉偵測問題是輸入一張圖片，透過人臉偵測的演算法，可以知道圖片中有沒有人臉。如果圖片中含有人臉，就計算出人臉在圖片中的哪裡。一般的作法是將物件偵測的演算法，套用在人臉上來解決問題。Viola 等人[1]使用機器學習的方式進行物件偵測，其採用的特徵值(Feature)是 Haar basis functions，他們定義了大量的特徵值，再利用 AdaBoost 技術來選擇較良好的 Feature，並使用了 detection cascade 來加速，最後輸出的結果是以一矩形框住物件。Active Shape Model(ASM)[2]也是使用機器學習的方式進行物件偵測，不過 ASM 不是使用矩形來描述物件，而是使用一堆特徵點集合來描述物件。他們對特徵點集合使用主成分分析(Principal Component Analysis, PCA)得到一個可漸變的模型，再用這個可漸變模型去找到圖片中的物件。Active Appearance Models(AAM)[3]的模型則是除了定義物件形狀之外，還包含了物件材質的資訊。

2.2. 離群值偵測

離群值偵測(Outlier detection)[4]問題是給定一堆資料，找出其中的離群值。離群值是在資料中明顯與其他資料不同的資料。在資料內容方面可分為單變數(univariate)和多變數(multivariate)。而在演算法方面又可分為參數化(parametric)和非參數化(non-parametric)，其中非參數化又分為 distance-based 和 clustering-based 和 spatial-based。

其中 distance-based outlier(DB outlier)[5]的定義：若一個觀察值與百分之 β 的其他觀察值超過 r 以上的距離，則該觀察值是 outlier，或者不定義 β ，改為定義若第 n 個近的鄰居距離超過 r ，則該觀察值是 outlier。在某種意義上這兩種定義是相同的。

2.3. 去背

去背(Matting)問題[6]是給定一張圖，假設圖片是由前景和背景的內插結果，目標是將每個 pixel 分離出前景顏色、背景顏色和前景的不透明度值，通常寫成下式：

$$I_p = \alpha_p F_p + (1 - \alpha_p) B_p$$

P 是座標， I 是原圖， F 是前景， B 是背景， α 是前景的不透明度，由於未知數過多的關係，上面的式子沒辦法直接解，而需要使用者輸入。通常使用者輸入時是把 pixel 分為 3 類(前景、背景、未知)，即所謂的 trimap 來表示使用者輸入。Robust Matting[10] 將過去的去背演算法分為三大類演算法(sampling-based、propagation-based、整合兩種方法)，以下一一介紹。

Sampling-based 演算法透過統計已知的前景和背景的顏色分布，利用分布資訊去內插出未知像素的值。舉例來說 Bayesian matting[7] 是利用對數化最大事後機率(Log-MAP) (maximum a posteriori) 解出問題。這類型的方法對於稀疏的使用者輸入，或者是前景和背景顏色分布有重疊時，效果不好。

Propagation-based 演算法透過定義鄰近像素之間的相似度(affinity)，從已知像素去估計鄰近像素的 α 值，通常會轉化成解矩陣的問題。Poisson matting[8] 是假設前景和背景顏色是區域性的平滑(locally smooth)，對去背定義的式子微分化簡後得到 $\Delta\alpha = \text{div}(\nabla I / (F - B))$ 的 Poisson equation，最後使用迭代的方式解出問題。Closed-form matting[9] 也是假設前景和背景顏色是區域性的平滑，直接假設在小區域(3*3 window)中 $1/(F - B)$ 和 $B/(F - B)$ 是常數，化簡後推得 closed-form 的解方程，這類型的方法作出的結果通常會比 sampling-based 演算法來得模糊。

整合兩種方法的 Robust matting 結合了 sampling-based、propagation-based 兩種方法，他們擴充 closed-form 的式子，把前景跟背景變

成 super node 並視為所有 pixel 的鄰居，並提出新的 color sampling 策略來定義前景和背景與所有 pixel 之間的相似度。

對於去背演算法的評價，通常是先作出 ground truth alpha map，然後將結果 alpha map 與 ground truth alpha map 作相似度的比較。在 <http://www.alphamatting.com/> [12] 他們使用四種可以用來評價結果的方法，分別是 Sum of absolute differences (SAD)、Mean square error (MSE)、Gradient 和 Connectivity。其中 Gradient 是比較兩張 alpha map 的梯度差，而 Connectivity 則是定義連通性之後，比較兩張 alpha map 在連通性上的差異。

我們採用 closed-form matting 一方面是因為我們的輸入是非常稀疏的，Sampling-based 演算法不適合稀疏輸入，而 Propagation-based 演算法目前作最好的是 closed-form matting。實際上我們實驗了 closed-form matting 和 robust matting，結果是 closed-form matting 的表現較佳。

2.4. 自動頭髮去背



上面提到了去背演算法必須有使用者輸入來提供資訊，才能計算出對使用者來說有意義的結果，而所謂的自動去背演算法其實是自動產生正確的使用者輸入 trimap 來達到去背自動化的效果，其實就是先使用物件偵測，產生 trimap 然後套用去背演算法，在自動頭髮去背方面目前只有兩篇相關的論文。分別是 frequential and color analysis for hair mask segmentation [14] 和 Using integrated color and texture features for automatic hair detection [13]。

frequential and color analysis for hair mask segmentation 使用 Viola 的人臉偵測定義出頭部區域，他們提出了 Frequential Mask、Color Mask，然後利用這三個資訊去定義出 user 指定的前景背景後套用 closed-form matting 演算法。

Using integrated color and texture features for automatic hair

detection 也是使用 Viola 人臉偵測，然後用分群的技術找眼睛和嘴巴，再用機器學習方法找頭髮種子，接著對圖作分群進行種子區域擴張得到 segment，然後轉成 trimap，最後也是作 closed-form matting。



3. 演算法流程



圖 3-1 系統流程示意圖

我們假設使用者輸入的圖片包含完整的臉和頭髮，即圖片中的臉和頭髮完全落在圖片中，不碰到圖片邊界。如果圖片沒有包含完整的頭髮，那麼這張圖片的頭髮事實上是不適合納入頭髮資料庫的。我們所使用的去背皆使用 closed-form matting。

我們將演算法分為三大步驟：

1. 筆畫繪製
2. Trimap 繪製
3. 套用 matting

筆畫繪製包含頭髮線繪製、臉部線繪製和邊界線繪製，其中頭髮線繪製和臉部線繪製需要人臉偵測的資訊，而邊界線不用，只需要知道輸入圖片的長寬即可。而 trimap 繪製是利用頭髮線、臉部線和邊界線，作出頭部去背和臉部去背的 alpha map，接著頭部 alpha 減掉臉部 alpha 即可得到 alpha difference，再對 alpha difference 作二值化後加上型態學即可作出 trimap。接著拿原圖和 trimap 去作去背就得到最終的結果。接下來會對每一個步驟作詳細的說明。

3.1. 筆畫繪製

本章節將會詳細說明人臉偵測、頭髮線繪製、頭髮線微調、臉部線繪製和邊界線繪製。

3.1.1. 人臉偵測

我們使用 ASM 來作人臉偵測，我們沒有自己訓練模型，而是使用一個現成的模型。我們使用的模型是以 63 個點定義人臉形狀，模型特徵點最高只有到眉毛，因此我們沒有額頭位置的資訊。



圖 3-2 ASM 特徵點示意圖

在 ASM 特徵點示意圖中綠色點的部分，是人臉偵測出來的結果，以下會使用 ASM_i 表示第 i 個 ASM 特徵點。 ASM_1 到 ASM_{15} 是描述臉部外框， ASM_{16} 到 ASM_{21} 是左眉毛， ASM_{22} 到 ASM_{27} 是右眉毛， ASM_{28} 到 ASM_{32} 是右眼， ASM_{33} 到 ASM_{37} 是左眼， ASM_{38} 到 ASM_{48} 是鼻子， ASM_{49} 到 ASM_{63} 是嘴巴。而圖中紅色點的部分則是頭髮點 H_1 到 H_{15} 。

3.1.2. 臉部線繪製

臉部線的限制是不可以畫到脖子、眼睛和頭髮，不然會讓演算法效果變差。本篇的臉部線是由一個三角形所構成，三角形的三個頂點分別是 ASM_{38} 、 ASM_{46} 及 ASM_{58} 。

3.1.3. 邊界線繪製

邊界線是一張圖片邊緣的線。精確一點的定義是給定一張圖片其長寬為 w 和 h ，那麼邊界線就是點 $(0, 0)$ 連線到點 $(w-1, 0)$ ，再連到 $(w-1, h-1)$ ，再連到 $(0, h-1)$ ，最後連回 $(0, 0)$ 。由於輸入圖片的假設，我們就可以保證邊界線不會畫到頭髮。

3.1.4. 頭髮線繪製

根據三庭五眼，可以知道眼睛通常是在頭頂到下巴的正中間，所以我們以雙眼連線作為鏡射軸，將臉部外框往上翻，視為頭緣，即可得到頭的上半部。又因為頭髮是從頭部開始向外生長，所以初始的頭髮點需要再稍微向外移動。其中雙眼連線指的是 ASM_{32} 和 ASM_{37} 的連線。而臉部外框往上翻後的點叫作 $Mirror_1$ 到 $Mirror_{15}$ 。

使用 ASM_1 和 ASM_{15} 的平均當作是臉部中心點 C 。

$$C = \frac{ASM_1 + ASM_{15}}{2}$$

那麼頭髮點 i 的位置就是

$$H_i = \left[1 + 0.2 \left(\frac{8-i}{7} \right)^2 \right] (Mirror_i - C) + C$$

相鄰的頭髮點 H_i 和 H_{i+1} 連線，稱為頭髮線段 L_i 。



圖 3-3 頭髮線示意圖

3.1.5. 頭髮線微調

先前的頭髮點可能不夠準確，由於我們必須確保頭髮線不會畫到皮膚和背景，所以再作一個調整使得頭髮點有更高的機率落在頭髮上。我們認為初始頭髮線有50%以上的機率會落在頭髮上，先計算頭髮線之間的相似度，再根據相似度矩陣去排除離群值(outlier)，以排除掉那些不是落在頭髮上的頭髮線，然後根據剩下的頭髮線的資訊去調整頭髮線。

頭髮線的相似度定義是以頭髮線 L_i 為前景，以邊界線為背景作 matting 得到的 alpha 作為特徵向量 A_i ，再使用 alpha 的相似度比較方法，得到相似度。



圖 3-4 頭髮線 alpha 示意圖

我們根據 Jaccard Index 的定義：相似度等於交集除以聯集，提出 Segment 的相似度定義： $\Sigma(A \& B) / \Sigma(A | B)$

$$\text{Segment 相似度}(A, B) = \frac{\sum_p A_p \& B_p}{\sum_p A_p | B_p}$$

其中 $\&$ 和 $|$ 是位元運算 and 和 or。這種相似度定義的特色是只考慮兩張圖前景區域的相似度，而不考慮背景區域，所以兩個位置完全不同的前景，不管前景大小為何，其相似度會是 0。將上式推廣到 alpha 相似度定義： $\Sigma(\min(A, B)) / \Sigma(\max(A, B))$

$$\text{alpha 相似度}(A, B) = \frac{\sum_p \min(A_p, B_p)}{\sum_p \max(A_p, B_p)}$$

因為 alpha 可以解釋成 pixel 是前景的機率，若 A 與 B 是統計獨立，則 A 交集 B 為 AB ，A 聯集 B 為 $A+B-AB$ ，但是很明顯 A 和 B 皆是同一張圖的 alpha，所

以並不獨立。我們假設 A 和 B 具有包含關係，定義 A 交集 B 為 $\min(A, B)$ ，A 聯集 B 為 $\max(A, B)$ 。

我們使用這個相似度定義去排除頭髮線中的離群值

	H1	H2	H3	H4	H5	H6	H7	H8	H9	H10	H11	H12	H13	H14
H1	100%	72%	69%	92%	68%	56%	52%	46%	43%	44%	41%	41%	41%	42%
H2	72%	100%	96%	72%	92%	74%	70%	63%	59%	60%	58%	58%	57%	58%
H3	69%	96%	100%	70%	93%	75%	71%	64%	60%	61%	59%	59%	58%	59%
H4	92%	72%	70%	100%	69%	56%	53%	46%	43%	44%	42%	41%	41%	42%
H5	68%	92%	93%	69%	100%	80%	75%	68%	64%	65%	62%	62%	62%	62%
H6	56%	74%	75%	56%	80%	100%	94%	78%	80%	81%	76%	76%	75%	72%
H7	52%	70%	71%	53%	75%	94%	100%	79%	83%	84%	77%	77%	76%	71%
H8	46%	63%	64%	46%	68%	78%	79%	100%	92%	93%	90%	90%	89%	86%
H9	43%	59%	60%	43%	64%	80%	83%	92%	100%	98%	93%	92%	91%	86%
H10	44%	60%	61%	44%	65%	81%	84%	93%	98%	100%	91%	91%	90%	84%
H11	41%	58%	59%	42%	62%	76%	77%	90%	93%	91%	100%	99%	97%	92%
H12	41%	58%	59%	41%	62%	76%	77%	90%	92%	91%	99%	100%	98%	93%
H13	41%	57%	58%	41%	62%	75%	76%	89%	91%	90%	97%	98%	100%	94%
H14	42%	58%	59%	42%	62%	72%	71%	86%	86%	84%	92%	93%	94%	100%

表 3-1 頭髮線 alpha 相似度矩陣

我們使用的離群值偵測(outlier detection)演算法是 Distance-based，參數 $r=0.7$ ， $n=6$ ，但如果發現超過半數(7 個)的頭髮線都被判定為離群值時，則只取距離最遠的 7 個離群值。

接下來根據頭髮線資訊調整頭髮點，我們認為剩下的 q 個頭髮線其 alpha 值越高的地方，就有越高的機率是真正的頭髮，所以我們對剩下的頭髮線 alpha 取平均，以得到頭髮線對於真正頭髮位置的共識 S ，然後對 S 套用 5×5 的模糊濾鏡(uniform blur filter)。

S_p = alphaAverage 上點座標 p 所對應的值

$$S_p = \frac{1}{q} \sum_{i=1}^q A_{ip}$$



圖 3-5 alpha 平均示意圖

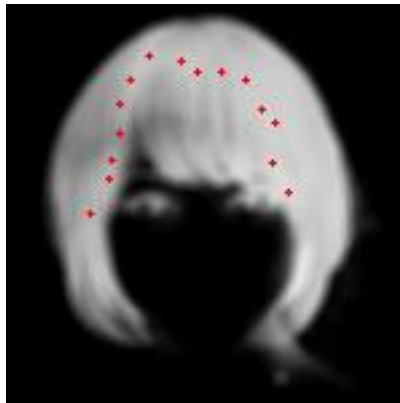


圖 3-6 模糊後的 alpha 平均和調整後的頭髮點位置

我們知道 S 圖中較高的值是頭髮，因此我們的目標是讓頭髮點移動後，頭髮線能夠經過較高的值，我們調整頭髮點座標時限制每一個頭髮點的可移動範圍，然後定義了三種目標：

1. 找到在 S 上平均值最高的頭髮線
2. 找到在 S 上總合最高的頭髮線
3. 找到在 S 上最小值最大的頭髮線

頭髮點 H_i 的可移動範圍 HM_i 是頭髮點根據臉部中心點作放射線狀方向 M_i 的移動軌跡，我們限制頭髮點的移動距離是 0.3 倍的頭髮點到臉部中心點的距離。

移動方向 = 頭髮點 - 中心點

$$M_i = H_i - C$$

頭髮點 x 的可移動範圍 = (頭髮點 + 0.7 移動方向) 連線到 (頭髮點 + 0.7 移動方向)

$$HM_i = \{point|(C + 0.7M_i) \text{ Line To } (C + 1.3M_i)\}$$

HM_i 是一個點集合，其中包含的點數量不固定，我們用符號 $HM_{i,j}$ 來表示其中的一個點， j 從 1 到 $|HM_i|$ ，表示 HM_i 中的第 j 個點。而 $LM_{i,j,k}$ 則是 $HM_{i,j}$ 與 $HM_{i-1,k}$ 的連線。



圖 3-7 頭髮點可移動範圍 HM 示意圖

HM 示意圖中有 15 條綠色線，分別代表著頭髮點的可移動範圍 HM_1 到 HM_{15} 。每條綠色線有三個紅點，中間的紅點是頭髮點。

我們的目標是在每一條綠色線 HM_i 上面都找到一個點 $HM_{i,j}$ ，使得這些點連線起來後得分最高，我們用一個長度 15 的向量 J 來表示每條綠色線上是哪些點被選取，假設在 HM_i 這條線上我們選取第 J_i 個點，那麼就得到目標函數如下：

$$J^* = \arg \max_J \left(\sum_{i=1}^{14} \text{cost}(LM_{i,J_{i+1},J_i}) \right)$$

其中的成本函數 $\text{cost}(L)$ 可以有以下三種可能：

線段上對應到的 alpha 總和

$$\text{costSum}(L) = \sum_{p \in L} S_p$$

線段上對應到的 alpha 平均

$$\text{costAverage}(L) = \frac{1}{|L|} \sum_{p \in L} S_p$$

線段上對應到的 alpha 最小值

$$\text{costMin}(L) = \min_{p \in L} S_p$$

假設每個頭髮點的可移動範圍大約是 30 個 pixel，那麼可能的組合數是 30^{15} ，這是很大的數字，表示我們不可能用窮舉的方式求解，因此我們就採用了動態規劃，設 $L_{i,j}$ 為以 $HM_{i,j}$ 為終點的最大 cost 值，以下是動態規劃的遞迴式。

$$L_{i,j} = \begin{cases} 0 & \text{if } i = 1 \\ \max_k (\text{cost}(LM_{i,j,k}) + L_{i-1,k}) & \text{if } i > 1 \end{cases}$$

動態規劃作完之後就得到移動後的頭髮點，但是我們會再對頭髮點作一次離群值偵測演算法，以排除掉一些選錯的可能性。

3.2. Trimap 繪製

直覺上的作法是以邊界線段為背景。為了保證臉部是背景，也把臉部線段也加入到背景，只留頭髮線作為前景。但是這樣的效果並不好，因為脖子的陰影很容易就會被誤判為頭髮。所以我們改成分別取出頭部去背和臉部去背，利用頭部 - 臉部 = 頭髮的式子得到頭髮 matting。



圖 3-8 頭髮去背 trimap 和 alpha

3.2.1. 頭部去背

頭部去背是以頭髮線、臉部線為前景，邊界為背景。



圖 3-9 頭部去背 trimap 和 alpha

3.2.2. 臉部去背

臉部去背是以臉部線為前景，頭髮線、邊界線為背景。



圖 3-10 臉部去背 trimap 和 alpha

我們利用 alpha 相似度的式子來比較頭部 alpha 減臉部 alpha 和頭髮 alpha 之間的差異性，結果有達到 90%以上的相似度。表示頭部 alpha 減臉部 alpha 大約等於頭髮 alpha。

3.2.3. Alpha 空間

Alpha 空間是把 alpha 視為基底向量，兩個線性獨立的 alpha 即可展開出一個向量空間，在此可以得到一個參數調整的機會。我們使用頭部 alpha 和臉部 alpha 當作兩個基底向量作線性組合，得到一個品質較好的 alpha 值，稱為 alpha difference。下圖中橫軸代表頭部 alpha，縱軸代表臉部 alpha，座標(1,1)的圖

代表是以 1 倍的頭部 alpha 減掉 1 倍的臉部 alpha，以此類推。

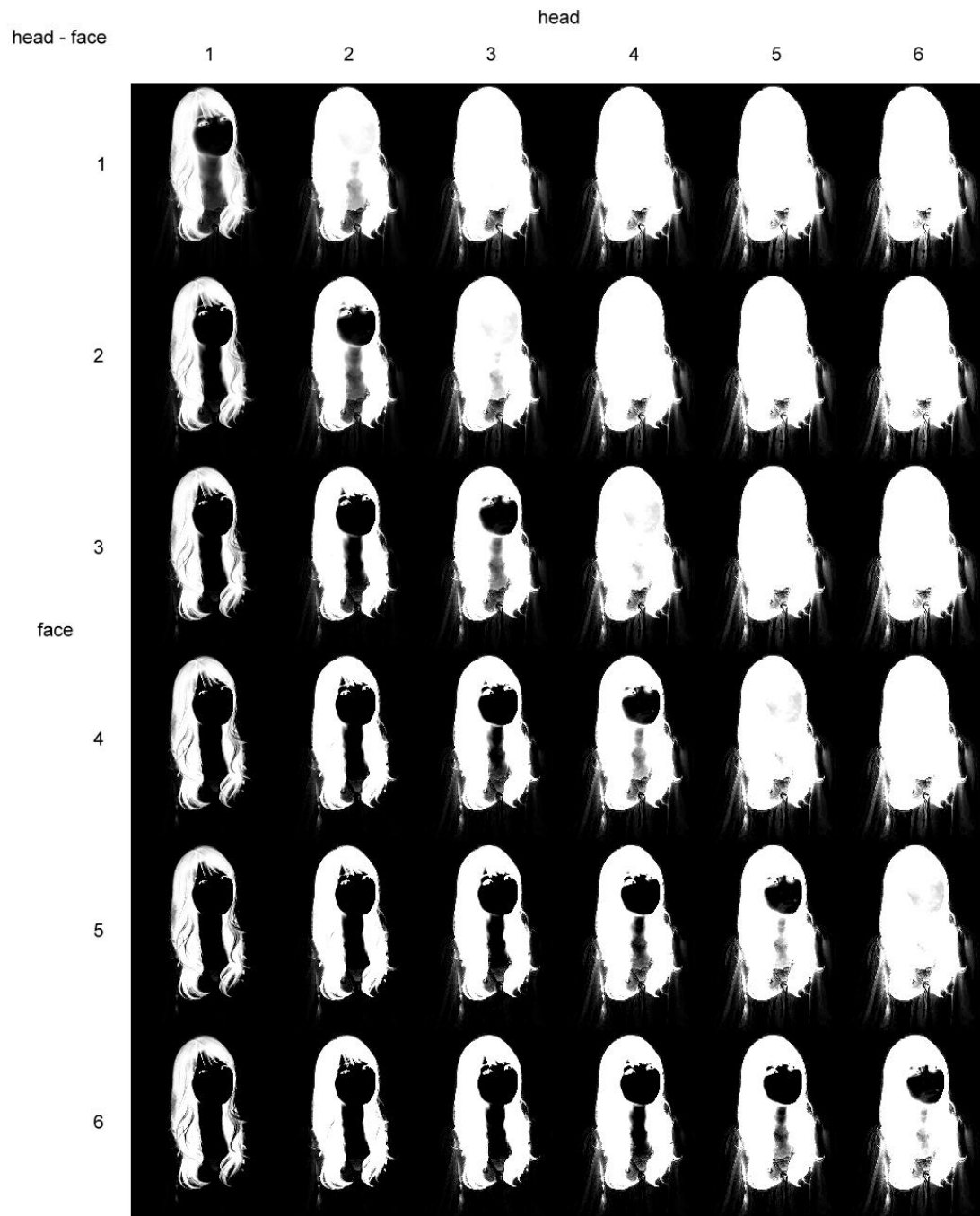


圖 3-11 alpha 空間示意圖

經過觀察，我們使用 $2 \times \text{頭部 alpha} - 4 \times \text{臉部 alpha}$ 得到去除脖子的頭髮 alpha。

3.2.4. 產生 Trimap

對 alpha difference 作二值化，然後作型態學的侵蝕運算即可得到 Trimap。我們利用臉部外框當比例尺，將侵蝕大小設為臉部外框除以 10，侵蝕大小必須大於眼睛高度的二分之一，因為這樣才能在侵蝕時將 alpha difference 中的眼睛從前景去除。根據三庭五眼我們知道臉部外框除以 5 大概就是眼睛寬度，而眼睛寬度的二分之一通常是大於眼睛高度的二分之一。設 $\text{alpha difference} > 0.5$ 的 pixel 作侵蝕後為前景(圖片中白色部分為前景)



圖 3-12 最終 trimap 的前景

設 $\text{alpha difference} < 0.5$ 的 pixel 作侵蝕後為背景(圖片中白色部分為背景)



圖 3-13 最終 trimap 的背景

最後將眼睛設為背景，以確保眼睛不會出現在最後結果上。眼睛的部分是透過 ASM 眼睛區域加上 2 個 pixel 的擴張。為了確保將眼睛設為背景時，不會影響到前景的部分，所以眼睛區域與前景區域的交集是判定為前景。Trimap 到此完

工，最後再執行一次 matting 就得到最終結果，而整個流程可件以下圖示。

首先將 ASM 眼睛區域標成白色的圖(未作擴張)



圖 3-14 眼睛區域示意圖

產生最終的 Trimap



圖 3-15 最終 Trimap

再套用 matting 後，最後得到的最終 Alpha 結果



圖 3-16 最終 alpha

套用到原圖後得到所擷取出的髮型。



圖表 3-1 最終結果



4. 實驗結果與討論

我們在日本髮型網站 <http://www.beauty-box.jp> 隨機抓了 70 張圖片，以手動的方式畫 trimap，並且將執行 closed-form matting 視為標準答案用來評價實驗結果。使用的評價函數參考 alpha evaluation 網站，使用 MSE、SAD 以及我們提出的 alpha 相似度。實驗了 DB outlier 的參數對結果的影響、Cost function 對結果的影響、alpha difference 與 final Alpha 的關係。使用的圖片大小為 1XX*200 或 200*1XX，對原圖使用等比例縮放。



4.1. 實驗結果統計

此處的 alpha 相似度為配合 MSE 與 SAD 值越小表示品質越好，改為 $1 - \alpha$ 相似度。（紅底為最佳品質）

評價函數	統計值	CostAverage	CostMin	CostSum
MSE	Max	86314	86264	121467
	Min	10654	10655	19610
	Avg	44899	45596	56876
SAD	Max	233	320	470
	Min	29	29	65
	Avg	118	121	176
Alpha 相似度	Max	0.805	0.88	0.96
	Min	0.071	0.07	0.111
	Avg	0.22	0.233	0.421

表 4-1 Alpha difference 的評價

評價函數	統計值	CostAverage	CostMin	CostSum
MSE	Max	86392	86842	119088
	Min	8843	8844	18053
	Avg	42807	43482	54997
SAD	Max	219	305	478
	Min	21	21	56
	Avg	104	108	155
Alpha 相似度	Max	0.804	0.841	0.957
	Min	0.024	0.024	0.057
	Avg	0.138	0.149	0.349

表 4-2 Final Alpha 的評價

統計值 Max 表示 70 張圖中，值最高的那張圖的值，Min 表示 70 張圖中，值最低的那張圖的值，Avg 表示 70 張圖的值平均，值則是由評價函數代入圖片和 ground truth 計算出。

在此看出 CostAverage 與 CostMin 的品質接近，但還是 CostAverage 勝出，而 CostSum 為了圈選到較多的 pixel，結果就是只考慮到 pixel 距離，不考慮圖片資訊，造成了最後品質差的結果。

以下圖片第一列是頭髮 trimap，第二列是 final Alpha，第三列是 ground

truth。由左至右，左邊是第 1 名，以此類推。

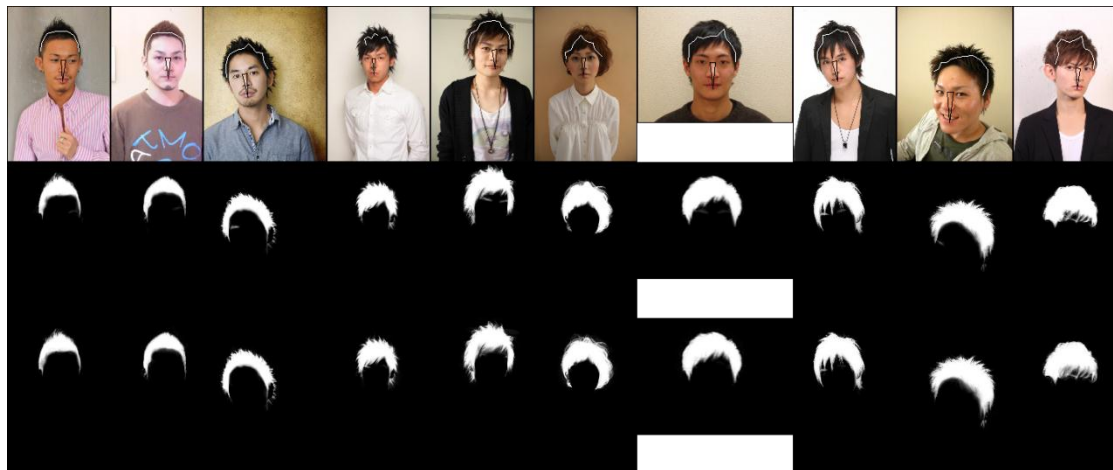


圖 4-1 CostAverage 品質最好前 10 張以 MSE 評價

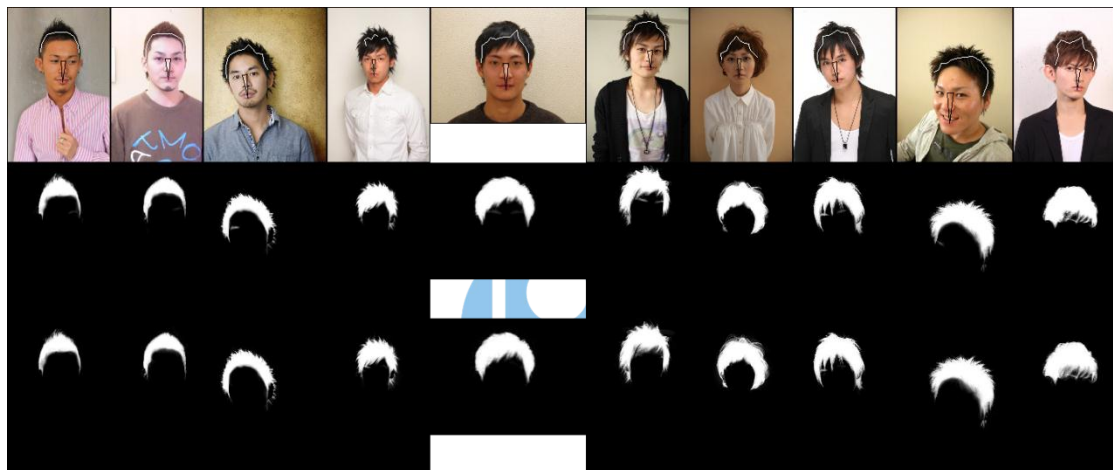


圖 4-2 CostAverage 品質最好前 10 張以 SAD 評價

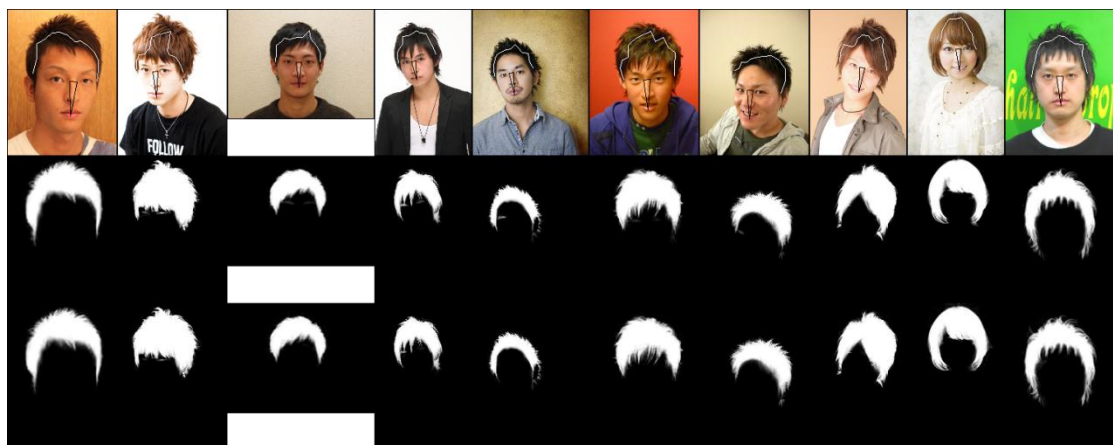


圖 4-3 CostAverage 品質最好前 10 張以 Alpha 相似度評價



圖 4-4 CostAverage 品質最差前 10 張以 MSE 評價

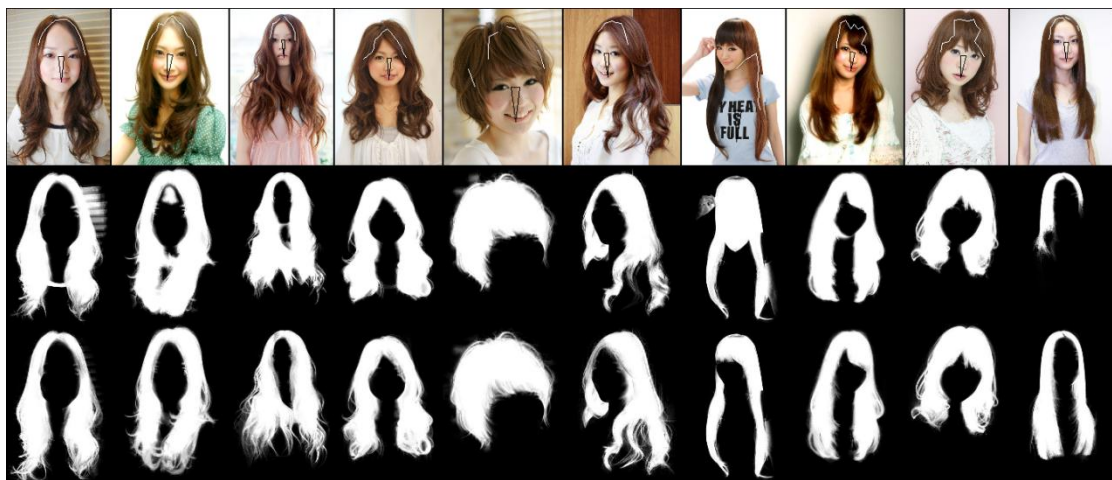


圖 4-5 CostAverage 品質最差前 10 張以 SAD 評價



圖 4-6 CostAverage 品質最差前 10 張以 Alpha 相似度評價

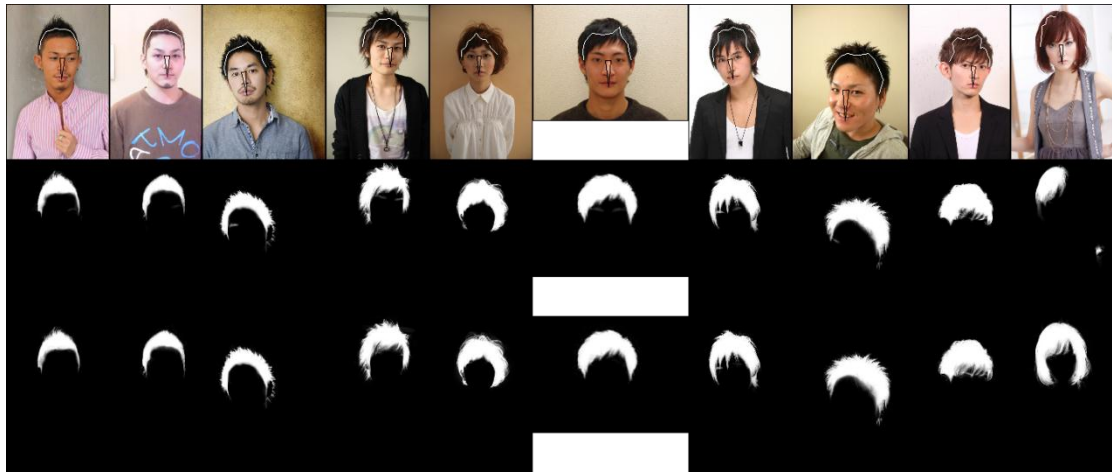


圖 4-7 CostMin 品質最好前 10 張以 MSE 評價

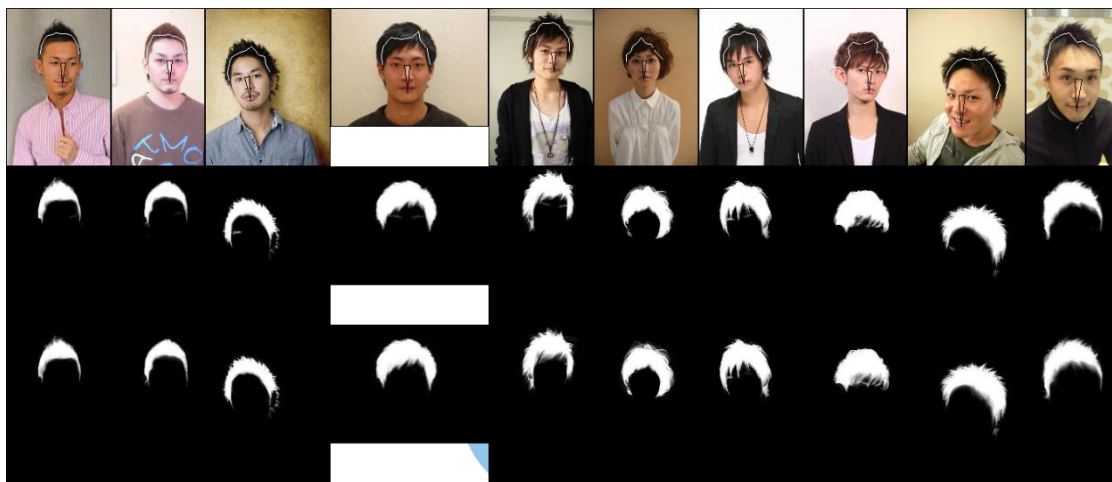


圖 4-8 CostMin 品質最好前 10 張以 SAD 評價



圖 4-9 CostMin 品質最好前 10 張以 Alpha 相似度評價



圖 4-10 CostMin 品質最差前 10 張以 MSE 評價



圖 4-11 CostMin 品質最差前 10 張以 SAD 評價



圖 4-12 CostMin 品質最差前 10 張以 Alpha 相似度評價



圖 4-13 CostSum 品質最好前 10 張以 MSE 評價



圖 4-14 CostSum 品質最好前 10 張以 SAD 評價



圖 4-15 CostSum 品質最好前 10 張以 Alpha 相似度評價



圖 4-16 CostSum 品質最差前 10 張以 MSE 評價



圖 4-17 CostSum 品質最差前 10 張以 SAD 評價



圖 4-18 CostSum 品質最差前 10 張以 Alpha 相似度評價

我們觀察圖片發現當頭髮線畫到背景或臉時品質會明顯變差。另外也可以發現 MSE 與 SAD 明顯傾向於選擇前景較小的圖，但我們提出的 alpha 相似度並不會因為前景大小而影響評價的值。可以看出 alpha 相似度確實比 MSE 和 SAD 好。

4.2. Alpha difference 與最終 Alpha 的關係

我們想知道是否 Alpha difference 的品質越好，final Alpha 的品質也會越好。所以比較 Alpha difference 前 10 名與 Final Alpha 前 10 名，若 Alpha difference 的前 10 名有出現在 Final Alpha 的前 10 名中，則標記上紅色。表格中的數字是指圖片編號。

圖片種類	評價方法	圖片來源	1	2	3	4	5	6	7	8	9	10
Alpha difference	Alpha 相似度	best 10	66	24	63	45	31	62	4	26	55	6
Final Alpha			30	66	38	12	54	69	49	31	57	28
Alpha difference	Alpha 相似度	worst 10	53	10	9	13	23	39	41	27	16	68
Final Alpha			53	9	23	10	13	41	68	5	34	40
Alpha difference	SAD	best 10	44	54	42	39	62	38	25	12	49	14
Final Alpha			39	44	54	42	38	25	62	12	49	14
Alpha difference	SAD	worst 10	11	52	7	47	4	17	41	8	64	45
Final Alpha			11	52	7	4	47	17	41	64	45	9
Alpha difference	MSE	best 10	39	44	54	42	25	62	38	12	14	23
Final Alpha			39	44	54	42	25	62	38	12	49	14
Alpha difference	MSE	worst 10	11	52	47	4	7	45	64	37	58	51
Final Alpha			11	52	4	47	7	45	64	37	17	58

表 4-3 Alpha difference 與最終 Alpha 的關係 - CostAverage

圖片種類	評價方法	圖片來源	1	2	3	4	5	6	7	8	9	10
Alpha difference	Alpha 相似度	best 10	66	24	63	31	45	62	4	55	26	6
Final Alpha			30	66	38	12	54	69	49	31	67	57
Alpha difference	Alpha 相似度	worst 10	42	43	23	9	13	10	39	18	41	27
Final Alpha			43	42	9	23	10	13	41	18	5	68
Alpha difference	SAD	best 10	44	54	39	62	38	12	25	26	14	49
Final Alpha			39	44	54	38	25	62	12	14	49	36
Alpha difference	SAD	worst 10	43	11	52	7	42	47	4	17	41	64
Final Alpha			43	11	52	7	4	42	47	17	41	64
Alpha difference	MSE	best 10	39	44	54	25	62	38	12	14	49	13
Final Alpha			39	44	54	25	62	38	12	49	14	23
Alpha difference	MSE	worst 10	11	52	43	4	47	7	45	64	37	58
Final Alpha			11	52	4	47	43	7	45	64	37	17

表 4-4 Alpha difference 與最終 Alpha 的關係 - CostMin

圖片種類	評價方法	圖片來源	1	2	3	4	5	6	7	8	9	10
Alpha difference	Alpha 相似度	best 10	45	63	37	24	64	66	4	47	65	29
Final Alpha			63	45	66	24	4	37	65	69	46	64
Alpha difference	Alpha 相似度	worst 10	42	54	49	39	9	13	38	28	26	23
Final Alpha			42	9	54	49	28	26	13	23	39	62
Alpha difference	SAD	best 10	65	44	23	25	39	66	69	12	21	24
Final Alpha			65	44	39	25	23	12	69	66	21	22
Alpha difference	SAD	worst 10	42	11	6	49	28	3	7	8	52	51
Final Alpha			42	11	6	28	49	8	52	3	7	51
Alpha difference	MSE	best 10	44	39	23	65	25	12	14	13	21	62
Final Alpha			44	39	23	65	25	12	13	10	21	69
Alpha difference	MSE	worst 10	42	11	52	6	7	47	4	51	8	28
Final Alpha			42	11	6	52	7	47	4	28	51	8

表 4-5 Alpha difference 與最終 Alpha 的關係 - CostSum

明顯可以看出 Alpha difference 與 Final Alpha 具有正向關係，但在 CostAverage 與 CostMin 中的 Alpha 相似度 best 10，正向關係似乎就不那麼明顯。推測是因為 Alpha difference 尚未去除眼睛的關係，因為 Final Alpha 非常相似於 ground truth，因此排序結果就受到眼睛大小所影響。

另外，我們觀察了 Alpha difference 的品質是否一定比 Final Alpha 差，透過算式：評價函數(alpha difference) - 評價函數(Final Alpha)，觀察是否皆大於 0，結果是 68 個大於 0，2 個小於 0。不過小於 0 的 case 都是接近 worst top 10 那邊，所以大方向來說有一定品質以上的圖都滿足評價函數(alpha difference) - 評價函數(Final Alpha) > 0。

4.3. DB outlier 的參數比較：

評價函數	統計值	CostAverage	CostMin	CostSum
MSE	Max	86392	86842	119088
	Min	8843	8844	18053
	Avg	42807	43482	54997
SAD	Max	219	305	478
	Min	21	21	56
	Avg	104	108	155
Alpha 相似度	Max	0.804	0.841	0.957
	Min	0.024	0.024	0.057
	Avg	0.138	0.149	0.349

表 4-6 DB outlier 參數 $r=0.7, n=6$ 得到的 final Alpha 統計

評價函數	統計值	CostAverage	CostMin	CostSum
MSE	Max	87982	88168	119088
	Min	9480	11137	23121
	Avg	43390	44293	58085
SAD	Max	220	305	478
	Min	23	25	56
	Avg	106	111	168
Alpha 相似度	Max	0.804	0.841	0.957
	Min	0.024	0.025	0.057
	Avg	0.151	0.169	0.388

表 4-7 DB outlier 參數 $r=0.5, n=6$ 得到的 final Alpha 統計

評價函數	統計值	CostAverage	CostMin	CostSum
MSE	Max	1590	1326	0
	Min	637	2293	5068
	Avg	583	811	3088
SAD	Max	1	0	0
	Min	2	4	0
	Avg	2	3	13
Alpha 相似度	Max	0	0	0
	Min	0	0.001	0
	Avg	0.013	0.02	0.039

表 4-8 表格相減： $r=0.5$ 的表格減掉 $r=0.7$ 的表格

統計結果比較發現 $r=0.5$ 的結果比 $r=0.7$ 差，仔細觀察每張圖片的品質後發

現不是所有的圖都是 $r=0.7$ 有較好的品質。不過 $r=0.5$ 比 $r=0.7$ 好的圖並沒有明顯差異，而 $r=0.7$ 比 $r=0.5$ 好的圖卻有明顯差異。圖片編號 49 可以看出明顯差異。



圖 4-19 圖片編號 49 的人臉偵測結果



圖 4-20 圖片編號 49 使用 DB outlier 參數 $r=0.7$ 的最終 alpha



圖 4-21 圖片編號 49 使用 DB outlier 參數 $r=0.5$ 的最終 alpha

我們將 DB outlier 的參數 n 值設定為 6，是根據圖片共有 14 張。假設 H_1 到 H_{14} 中正確落在頭髮上的數量會大於半數(7)，因為當 H_1 到 H_{14} 不是落在頭髮上

時，也會有群聚現象，因此 n 值若設太小則無法判定出頭髮。我們對相似度矩陣作沉澱圖，觀察 DB outlier 的計算方式。沉澱圖的計算過程是 1：每個 row 自己遞減排序，2：以第 n 個 col 作遞減排序。

1	0.98	0.96	0.96	0.96	0.93	0.93	0.69	0.32	0.02	0.02	0	0	0
1	0.98	0.97	0.96	0.95	0.94	0.92	0.7	0.32	0.02	0.02	0	0	0
1	0.97	0.97	0.96	0.95	0.92	0.92	0.71	0.33	0.02	0.02	0	0	0
1	0.96	0.96	0.95	0.94	0.93	0.9	0.71	0.32	0.01	0.01	0	0	0
1	0.96	0.94	0.94	0.92	0.92	0.89	0.68	0.3	0.01	0.01	0	0	0
1	0.97	0.95	0.93	0.93	0.89	0.89	0.72	0.33	0.02	0.02	0	0	0
1	0.93	0.92	0.92	0.92	0.9	0.89	0.71	0.34	0.03	0.02	0	0	0
1	0.72	0.71	0.71	0.71	0.7	0.69	0.68	0.46	0.03	0.03	0	0	0
1	0.46	0.34	0.33	0.33	0.32	0.32	0.32	0.3	0.07	0.06	0	0	0
1	0.71	0.07	0.03	0.03	0.02	0.02	0.02	0.02	0.01	0.01	0	0	0
1	0.71	0.06	0.03	0.02	0.02	0.02	0.02	0.02	0.01	0.01	0	0	0
1	1	1	0	0	0	0	0	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0	0	0	0	0

表 4-9 圖片編號 46 相似度矩陣沉澱圖

因此考慮一個極端的情況：當資料只有兩種值，兩種值之間的相似度是 0，

第一種值有 8 個，第二種值有 6 個時，N 太小或太大都無法分類出結果

1	1	1	1	1	1	1	1	0	0	0	0	0	0
1	1	1	1	1	1	1	1	0	0	0	0	0	0
1	1	1	1	1	1	1	1	0	0	0	0	0	0
1	1	1	1	1	1	1	1	0	0	0	0	0	0
1	1	1	1	1	1	1	1	0	0	0	0	0	0
1	1	1	1	1	1	1	1	0	0	0	0	0	0
1	1	1	1	1	1	1	1	0	0	0	0	0	0
1	1	1	1	1	1	1	1	0	0	0	0	0	0
1	1	1	1	1	1	1	1	0	0	0	0	0	0
1	1	1	1	1	1	0	0	0	0	0	0	0	0
1	1	1	1	1	1	0	0	0	0	0	0	0	0
1	1	1	1	1	1	0	0	0	0	0	0	0	0
1	1	1	1	1	1	0	0	0	0	0	0	0	0
1	1	1	1	1	1	0	0	0	0	0	0	0	0

表 4-10 極端的例子相似度矩陣沉澱圖

4.4. 與其他 paper 比較：

我們從其他 paper 的 pdf 中取得圖片，並將圖片等比例縮放到長寬較大的值為 200 後，執行我們的演算法。

Paper1 名稱：frequential and color analysis for hair mask segmentation[14]：



圖 4-22 與 paper1 比較-我們的臉部去背 trimap



圖 4-23 與 paper1 比較-我們的最終 trimap



圖 4-24 與 paper1 比較-我們的最終結果



圖 4-25 與 paper1 比較-我們的最終 alpha



圖 4-26 與 paper1 比較-paper1 的最終 alpha

我們與 paper1 的結果(圖 4-26)比較並不是很理想，我們在右邊的部分有一個大區域的誤判，因此我們將原圖的頭部切下後再跑一次我們的演算法。此處提到的頭部是 paper1 對於頭部的定義。



圖 4-27 與 paper1 比較-原圖裁剪後我們的臉部去背 trimap



圖 4-28 與 paper1 比較-原圖裁剪後我們的最終 trimap



圖 4-29 與 paper1 比較-原圖裁剪後我們的最終結果



圖 4-30 與 paper1 比較-原圖裁剪後我們的最終 alpha

由於我們的頭髮線是由多個 closed-form matting 的結果組成，而 closed-form matting 又有模糊的特性，因此在頭髮附近的背景若出現顏色和距離皆與頭髮接近的情形，就會有不好的結果產生。

Paper2 名稱：Using integrated color and texture features for automatic hair detection[13]：



圖 4-31 與 paper2 比較-臉部去背 trimap



圖 4-32 與 paper2 比較-最終 trimap



圖 4-33 與 paper2 比較-最終結果



圖 4-34 與 paper2 比較-最終 alpha



圖 4-35 與 paper2 比較-paper2 的 hair mask

由於 paper2 並沒有附上任何一張 alpha map，所以只能跟他的 hair mask 作比較。我們的頭髮線畫的偏高了，所以品質就稍微差了點。

4.5. 演算法時間效能分析：

軟體：

使用的程式語言是 C++，closed-form matting 是採用原作的 matlab code，透過 C++ matlab engine 來作溝通。

硬體：

CPU：Intel® Core™2 Quad CPU Q9505 @ 2.83GHz

RAM：4G

OS：Windows XP

Matlab 版本：7.0

時間：

人臉偵測：小於 1 秒

所需 Matting 次數：31 次(頭髮線 14*2 次、頭部臉部各 1 次、最終 1 次)

花在 Matting 上的時間：約 8 分鐘

動態規劃：1~5 秒不等

平均一張圖約 8 分鐘，最主要的時間是花在 Matting 上。

4.6. 系統限制

我們的系統限制是輸入圖片的頭髮不能觸碰到圖片邊框，以及初始頭髮點必須有一半以上落在頭髮上，而通常會作失敗的情形是發生在人臉偵測失敗或 Closed-form matting 的 alpha 產生破碎的現象，而結果品質差的情形是發生在背景上有小色塊與頭髮相連，背景顏色與皮膚相近，或者皮膚顏色與頭髮相近等情形。

人臉偵測失敗：



圖 4-36 人臉偵測失敗

Closed-form matting alpha 破碎：



圖 4-37 Closed-form matting alpha 破碎 case1-頭髮去背 trimap



圖 4-38 Closed-form matting alpha 破碎 case1-頭髮去背 alpha



圖 4-39 Closed-form matting alpha 破碎 case2-頭部去背 trimap



圖 4-40 Closed-form matting alpha 破碎 case2-頭部去背 alpha
背景有小色塊與頭髮相連：



圖 4-41 背景有小色塊與頭髮相連-頭髮去背 trimap



圖 4-42 背景有小色塊與頭髮相連-頭髮去背 alpha

5. 結論與未來展望

本篇論文有三大貢獻，其一是我們在自動頭髮去背這方面提出了一個完整的架構：定義頭髮、臉部、和邊界三個區塊，接著作頭部和臉部去背，作出 trimap，最後得到去背結果。其二是我們也提出新的評價函數:alpha 相似度，可用來評價去背演算法。其三是我們另外提出了 alpha 空間的概念，拿兩張 alpha 當作基底向量張出一個向量空間，使我們可以在此空間中尋找品質較好的 alpha。

但就結果上來看，仍有進步的空間。在複雜背景的情況下，我們還無法得到一個商業用品質的頭髮，但是在單純背景的情況下系統運作還算順利。若想要在品質上得到進步，就應該找一個更適合此架構的去背演算法。比方說搭配材質分割(texture segmentation)的評量標準(metric)，或許就能夠支援複雜材質的背景。另外在時間方面，我們幾乎花了所有的時間在頭髮線繪製上，若能找到更有效率的頭髮線繪製方法，就能大幅減少演算法時間。事實上 Automatic Hair Detection in the Wild[15]也定義了頭髮上半部的模型，叫作 Upper Hair Shape Model(UHSM)，看起來效果不錯，但是在他們的 paper 裡面並沒有提到模型配適(model fitting)會花多少時間。

我們也可以往其他方向發展，例如將此架構套用到非頭髮的物件上，例如衣服或者全身之類的自動去背。另外，修改我們的動態規劃演算法，作成 match 物件外型的效果，應該也是個不錯的發展方向。

我們的圖片有限制頭髮不能觸碰到圖片邊界，所以我們可以從圖片中裁剪出一個最小的矩形滿足限制條件，就像與 frequential and color analysis for hair mask segmentation[14]作比較時一樣。理論上我們先對輸入圖片對最小矩形作裁切，拿得到的新圖當作是輸入圖片再去執行演算法，就可以得到更高的品質。如果結合一些偵測圖片中重點物件的演算法，應該可以作出以框選的方式選取不規則外形物件的選取工具。

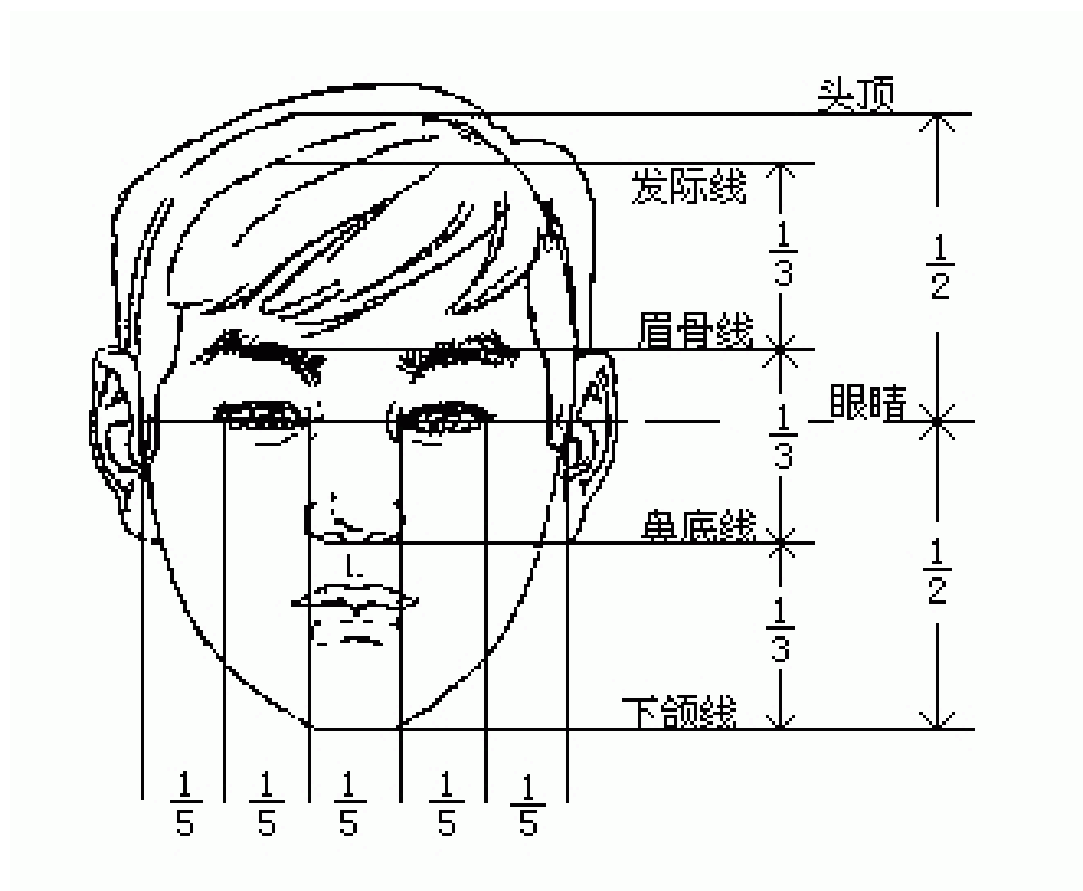
參考文獻

- [1] Paul Viola, Michael Jones. Rapid Object Detection using a Boosted Cascade of Simple Features. *Computer Vision and Pattern Recognition*, 2001.
- [2] Timothy F. Cootes, Christopher J. Taylor, D. H. Cooper, J. Graham. Active Shape Models-Their Training and Application. In *Computer Vision and Image Understanding*, volume 61(1), pages 38-59, 1995.
- [3] Timothy F. Cootes, Gareth J. Edwards, Christopher J. Taylor. Active Appearance Models. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 23(6), 2001.
- [4] Irad Ben-Gal. Outlier Detection. In *Data Mining and Knowledge Discovery Handbook*, 2005.
- [5] Edwin M. Knorr, Raymond T. Ng, Vladimir Tucakov. Distance-based outliers: algorithms and applications. *The VLDB Journal — The International Journal on Very Large Data Bases*, volume 8, 2000.
- [6] Jue Wang, Michael F. Cohen. Image and video matting: a survey. *Foundations and TrendsR in Computer Graphics and Vision*, 2007.
- [7] Y.Y. Chuang, B. Curless, D. Salesin, R. Szeliski. A Bayesian Approach to Digital Matting. *Computer Vision and Pattern Recognition*, 2001.
- [8] J. Sun, J. Jia, C.K. Tang, H.Y. Shum. Poisson Matting. *SIGGRAPH*, 2004.
- [9] A. Levin, D. Lischinski, Y. Weiss. A Closed Form Solution to Natural Image Matting. *Computer Vision and Pattern Recognition*, 2006.
- [10] J. Wang, M. Cohen. Optimized Color Sampling for Robust Matting. *Computer Vision and Pattern Recognition*, 2007.
- [11] Xue Bai, Guillermo Sapiro. A geodesic framework for fast interactive image and video segmentation and matting. *International Conference on Computer Vision*, 2007.

- [12] Christoph Rhemann, Carsten Rother, Jue Wang, Margrit Gelautz, Pushmeet Kohli, Pamela Rott. A Perceptually Motivated Online Benchmark for Image Matting. *Computer Vision and Pattern Recognition*, 2009.
- [13] U. Lipowezky, O. Mamo, A. Cohen. Using integrated color and texture features for automatic hair detection. In *Convention of Electrical and Electronics Engineers in Israel*, 2008.
- [14] C. Rousset, P. Y. Coulon. Frequential and color analysis for hair mask segmentation. In *International Conference on Image Processing*, 2008.
- [15] P. Julian and C. Dehais and Lauze and V. Charvillat and A. Bartoli and A. Choukroun. Automatic Hair Detection in the Wild. *International Conference on Pattern Recognition*, 2010.



附錄



引用自 <http://www.hh75.com/article/fzdp/2010-11-05/972.html>, 怎樣畫時裝畫, 三庭五眼圖