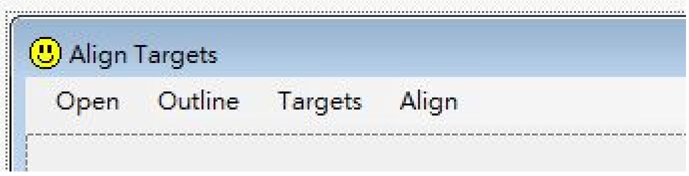
第 5 章如何找出字元目標群組

5-1 準備工作與介紹



請複製前一章的程式專案，改名為 AlignTargets，主功能表按鍵增刪為：Open, Outline, Targets 與 Align。接下來我們會一一隨著辨識程序實作這些功能按鍵，刪減的按鍵其實就是代表逐步變成一些副程式，整合到整體辨識流程中了，不是消失了哦！譬如本章就跳過灰階與二值化的顯示，但實際上要產生輪廓線，這些處理程序都還是必要的！

上一章我們已經介紹到可以建立有很多屬性的目標物件 TgInfo 了！但是實務上，很少影像辨識的目的只是要辨識一個**單一目標**物件的！譬如車牌辨識就至少要辨識 4 到 7 個字元的組合，如果是貨櫃碼就更長了！甚至一個中文字就可能是好幾個「**影像目標**」的集合，譬如「**川**」字就很明顯，一定會變成三個影像目標，但我們必須將它們從零散的目標堆中集合起來一起解析他們的意義。

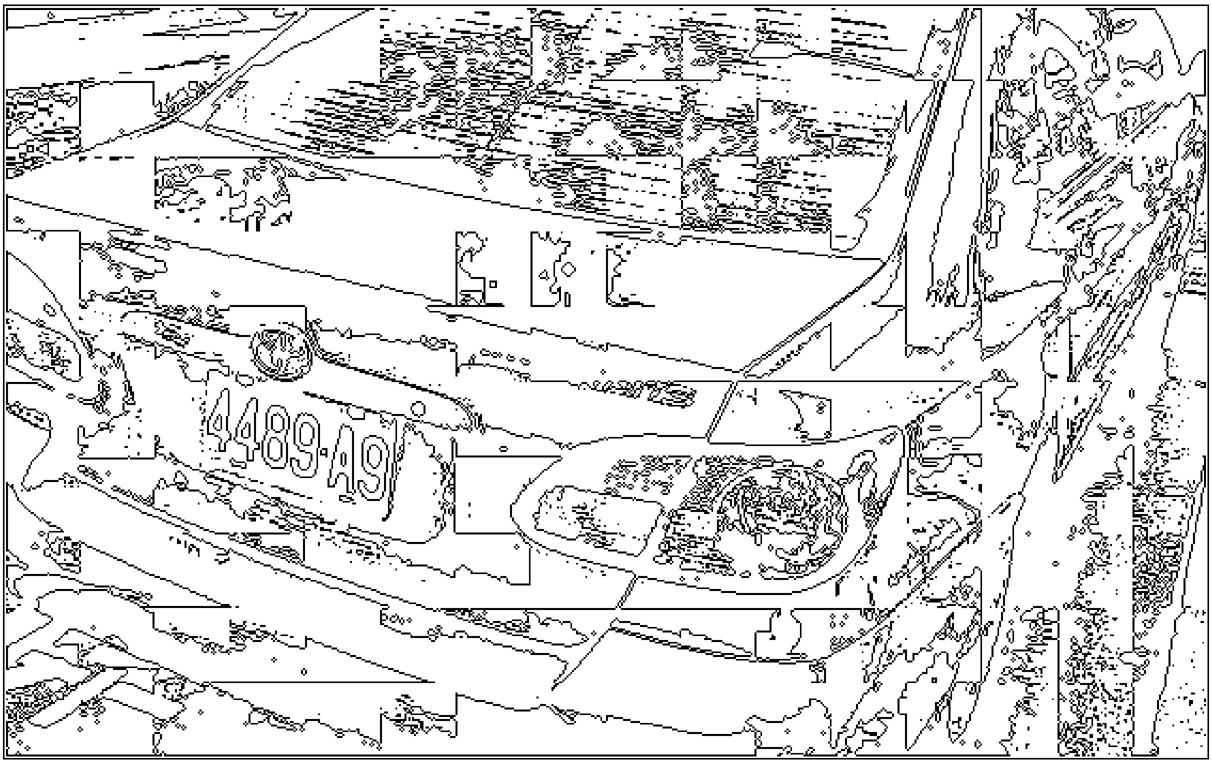
所以如何依照最終的辨識目的，找出合理的目標「**群組**」，是大多數**實務影像辨識必經的過程**，但是多數影像辨識書籍都不會詳細介紹這類實作技巧，所以建議大家一定要深入理解並熟練本章的介紹內容。在數學概念上就是將靠得很近，且有共同特性的目標找出來定義為一個集合，實作過程其實不難，但卻是人工智慧化我們視覺邏輯的一個

有趣過程。

5-2 整合後的 Outline 功能

在此我們將二值化處理變成一個獨立副程式，包含在 Outline 的顯示按鍵之中，ThresholdBuild 與 Ouline 副程式與之前一樣就不列出了！程式碼與執行結果如下：

```
'輪廓線
Private Sub OutlineToolStripMenuItem_Click(ByVal sender As Object, ByVal e As EventArgs) _
    Handles OutlineToolStripMenuItem.Click
    Z = DoBinary(B) '二值化
    Q = Outline(Z) '建立輪廓點陣列
    PictureBox1.Image = BWImg(Q) '建立輪廓圖
End Sub
'二值化
Private Function DoBinary(ByVal b(,) As Byte) As Byte(,)
    Th = ThresholdBuild(b) '建立二值化使用之門檻值陣列
    Dim Z(nx - 1, ny - 1) As Byte '建立二值化陣列
    For i As Integer = 1 To nx - 2
        Dim x As Integer = i \ Gdim 'x 座標換算
        For j As Integer = 1 To ny - 2
            Dim y As Integer = j \ Gdim 'y 座標換算
            If b(i, j) < Th(x, y) Then
                Z(i, j) = 1 '低於亮度門檻設為目標點
            End If
        Next
    Next
    Return Z
End Function
```



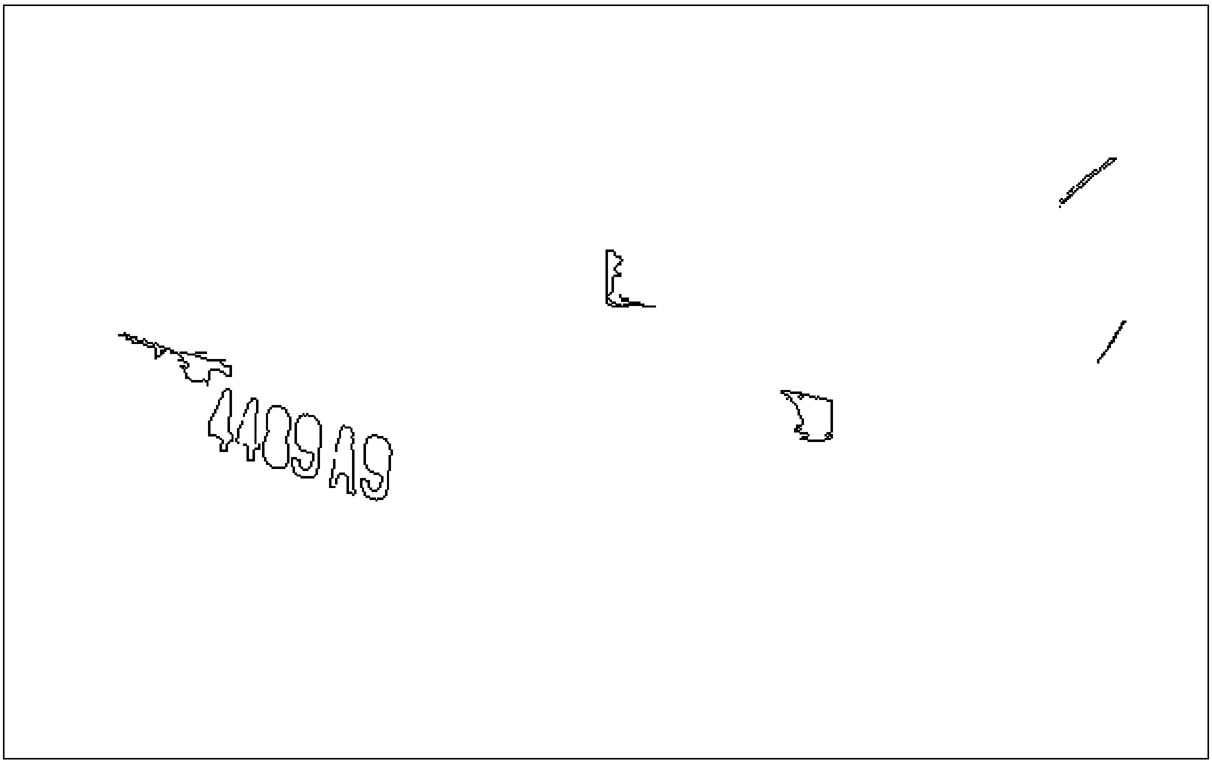
5-3 整合後的 Targets 功能

前一章為了讓讀者能清楚理解建立與篩選目標的過程，所以有獨立的 Filter 與 Sort 程序，在此我們就將篩選過程融入到目標建立的過程，也就是在建立過程中就直接將目標篩選與排序，所以 getTargets 副程式所輸出的目標就已經過篩選排序了！還加上一個 Tgmax 的變數，最多只會輸出我們指定的目標個數。因為我們要辨識的多半是較明顯的目標，擇優選取部分較明顯目標即可，合格目標太多不會影響最後結果，但會增加資料處理量拖慢辨識速度。完整程式碼如下：

```
'以輪廓點建立目標陣列，排除負目標
Dim minHeight As Integer = 10, maxHeight As Integer = 80 '有效目標高度範圍
Dim minwidth As Integer = 2, maxWidth As Integer = 80 '有效目標寬度範圍
Dim Tgmax As Integer = 20 '進入決選範圍的最明顯目標上限
Function getTargets(ByVal q(,) As Byte) As ArrayList
    Dim A As New ArrayList
    Dim b(,) As Byte = q.Clone '建立輪廓點陣列副本
    For i As Integer = 1 To nx - 2
        For j As Integer = 1 To ny - 2
            If b(i, j) = 0 Then Continue For
            Dim G As New TgInfo
            G.xmn = i : G.xmx = i : G.ymn = j : G.ymx = j : G.P = New ArrayList
            Dim nc As New ArrayList '每一輪搜尋的起點集合
            nc.Add(New Point(i, j)) '輸入之搜尋起點
            G.P.Add(New Point(i, j))
            b(i, j) = 0 '清除此起點之輪廓點標記
            Do
            Dim nb As ArrayList = nc.Clone '複製此輪之搜尋起點集合
            nc = New ArrayList '清除準備蒐集下一輪搜尋起點之集合
            For m As Integer = 0 To nb.Count - 1
                Dim p As Point = nb(m) '搜尋起點
                '在此點周邊 3X3 區域內找輪廓點
                For ii As Integer = p.X - 1 To p.X + 1
                    For jj As Integer = p.Y - 1 To p.Y + 1
                        If b(ii, jj) = 0 Then Continue For '非輪廓點忽略
                        Dim k As New Point(ii, jj) '建立點物件
                        nc.Add(k) '本輪搜尋新增的輪廓點
                        G.P.Add(k) '點集合
                        If ii < G.xmn Then G.xmn = ii
                        If ii > G.xmx Then G.xmx = ii
                        If jj < G.ymn Then G.ymn = jj
                        If jj > G.ymx Then G.ymx = jj
                        b(ii, jj) = 0 '清除輪廓點點標記
                    Next
                Next
            Next
        Loop While nc.Count > 0 '此輪搜尋有新發現輪廓點時繼續搜尋
        If Z(i - 1, j) = 1 Then Continue For '排除白色區塊的負目標，起點左邊是黑點
        G.width = G.xmx - G.xmn + 1 '寬度計算
        G.height = G.ymx - G.ymn + 1 '高度計算
        '以寬高大小篩選目標
        If G.height < minHeight Then Continue For
        If G.height > maxHeight Then Continue For
        If G.width < minwidth Then Continue For
        If G.width > maxWidth Then Continue For
        G.cx = (G.xmn + G.xmx) / 2 : G.cy = (G.ymn + G.ymx) / 2 '中心點
        G.np = G.P.Count
        '計算目標的對比度
        For m As Integer = 0 To G.P.Count - 1
            Dim pm As Integer = PointPm(G.P(m))
            If pm > G.pm Then G.pm = pm '最高對比度的輪廓點
        Next
        A.Add(G) '加入有效目標集合
    Next
Next
'以對比度排序
For i As Integer = 0 To A.Count - 2
    For j As Integer = i + 1 To A.Count - 1
        Dim T As TgInfo = A(i), G As TgInfo = A(j)
        If T.pm < G.pm Then A(i) = G : A(j) = T '互換位置，高對比目標在前
    Next
Next
'取得 Tgmax 個最明顯的目標輸出
```

```
    Dim C As New ArrayList
    For i As Integer = 0 To Tgmax - 1
        If i > A.Count - 1 Then Exit For '超過總目標數
        Dim T As TgInfo = A(i) : T.ID = i '建立以對比度排序的序號
        C.Add(T)
    Next
    Return C '回傳目標物件集合
End Function
```

與前一章不同之處還有我們加入了目標中心點的座標 cx 與 cy 屬性，就是目標左右與上下極值的中點了！還有將目標依據對比度排序之後，我們循序定義了 ID 的屬性，亦即 ID=0 的目標是影像中最明顯的目標，依此類推。至於顯示目標輪廓線的按鍵功能程式內容與之前一樣，也不再重複列出了！執行結果如下：



5-4 搜尋最佳目標群組

接下來就是本章的重頭戲，如何找出最可能是車牌字元目標的組合？我們依循的概略原則是：第一、車牌字元目標應該是影像中對比較明顯的目標，所以我們搜尋的優先次序都是按照它們的 ID 屬性。第二、車牌字元是群聚在一起，還概略水平的，所以相隔太遠的目標當然不會被視為同組。按照這兩個原則，我們可以建立一個很簡單的搜尋副程式 AlignTgs 如下，此副程式饋入的參數是所有合格目標的集合 C。

```
Private Function AlignTgs(ByVal C As ArrayList) As ArrayList
    Dim R As New ArrayList, pmx As Integer = 0 '最佳目標組合與最佳度比度
    For i As Integer = 0 To C.Count - 1
        Dim T As TgInfo = C(i) '核心目標
```

```
            Dim D As New ArrayList, Dm As Integer = 0 '此輪搜尋的目標集合
            D.Add(T) : Dm = T.pm '加入搜尋起點目標
            '搜尋 X 範圍
            Dim x1 As Integer = T.cx - T.height * 2.5, x2 As Integer = T.cx + T.height * 2.5
            '搜尋 Y 範圍
            Dim y1 As Integer = T.cy - T.height * 1.25, y2 As Integer = T.cy + T.height * 1.25
            For j As Integer = 0 To C.Count - 1
                If i = j Then Continue For '與起點重複略過
                Dim G As TgInfo = C(j)
                If G.cx < x1 Then Continue For
                If G.cx > x2 Then Continue For
                If G.cy < y1 Then Continue For
                If G.cy > y2 Then Continue For
                D.Add(G) : Dm += G.pm '合格目標加入集合
                If D.Count >= 7 Then Exit For '目標蒐集個數已滿跳離迴圈
            Next
            If Dm > pmx Then '對比度高於之前的目標集合
                pmx = Dm : R = D
                Rec = New Rectangle(x1, y1, x2 - x1 + 1, y2 - y1 + 1) '搜尋範圍
            End If
        Next
        Return R
    End Function
```

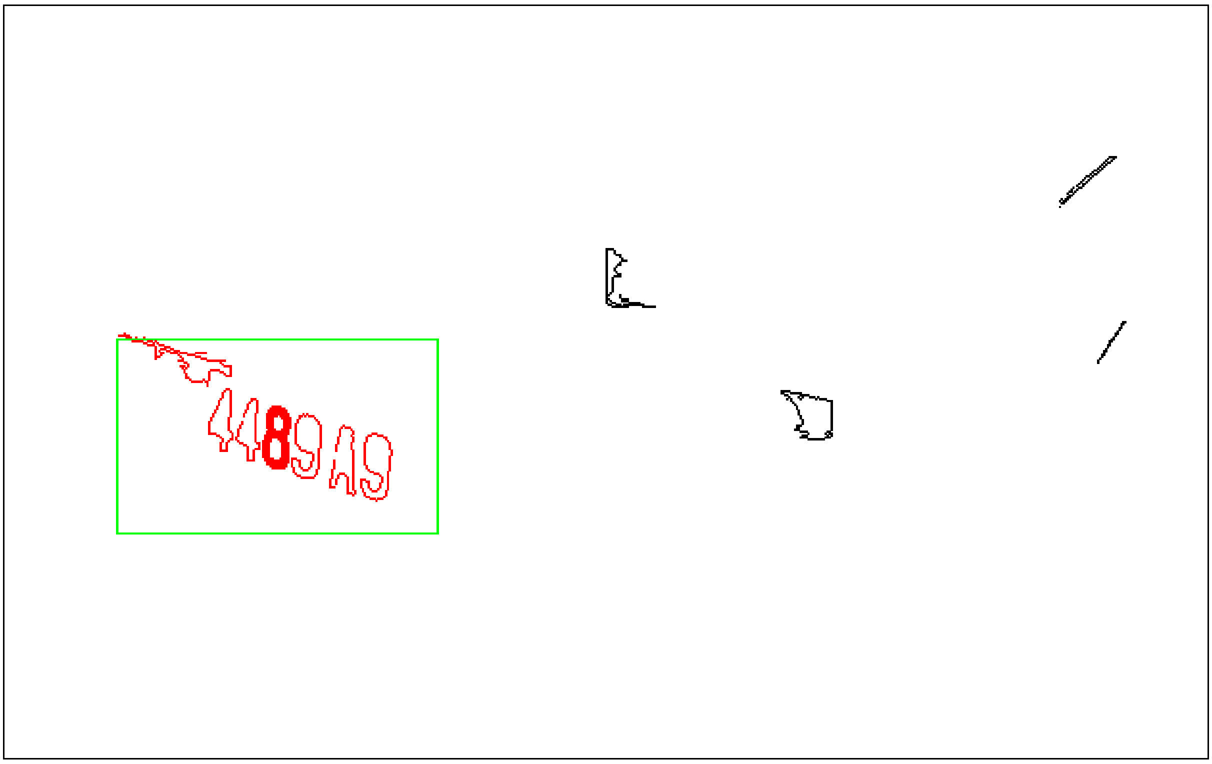
此副程式會依序從最明顯的目標開始，以該目標為核心，假定它是車牌字元之一，在此目標周圍搜尋可能的目標，因為車牌是左右寬上下窄的，所以預設搜尋寬度是從目標中心點左右延伸此目標高度的 2.5 倍，上下則是只延伸 1.25 倍。凡是中心點在此範圍之內的目標就會被收進一個暫時的集合物件，當然還是按照目標對比度的順序，比較明顯的目標會先被收錄。

因為車牌字數是有上限的，在台灣是 7 個字，所以這個搜尋收編的過程，如果到達 7 個就應該立即停止(跳出迴圈)，再繼續找更多較不明顯的目標沒有意義。事實上如果碰到六碼車牌，可能搜完全圖也找不到七碼的組合。不論有沒有達到七碼，通常目標組合的對比度總和最大，就表示這個組合最明顯，也最可能是車牌候選人！所以每一個組合的對比度總和就是比較哪個組合最好的重要依據。

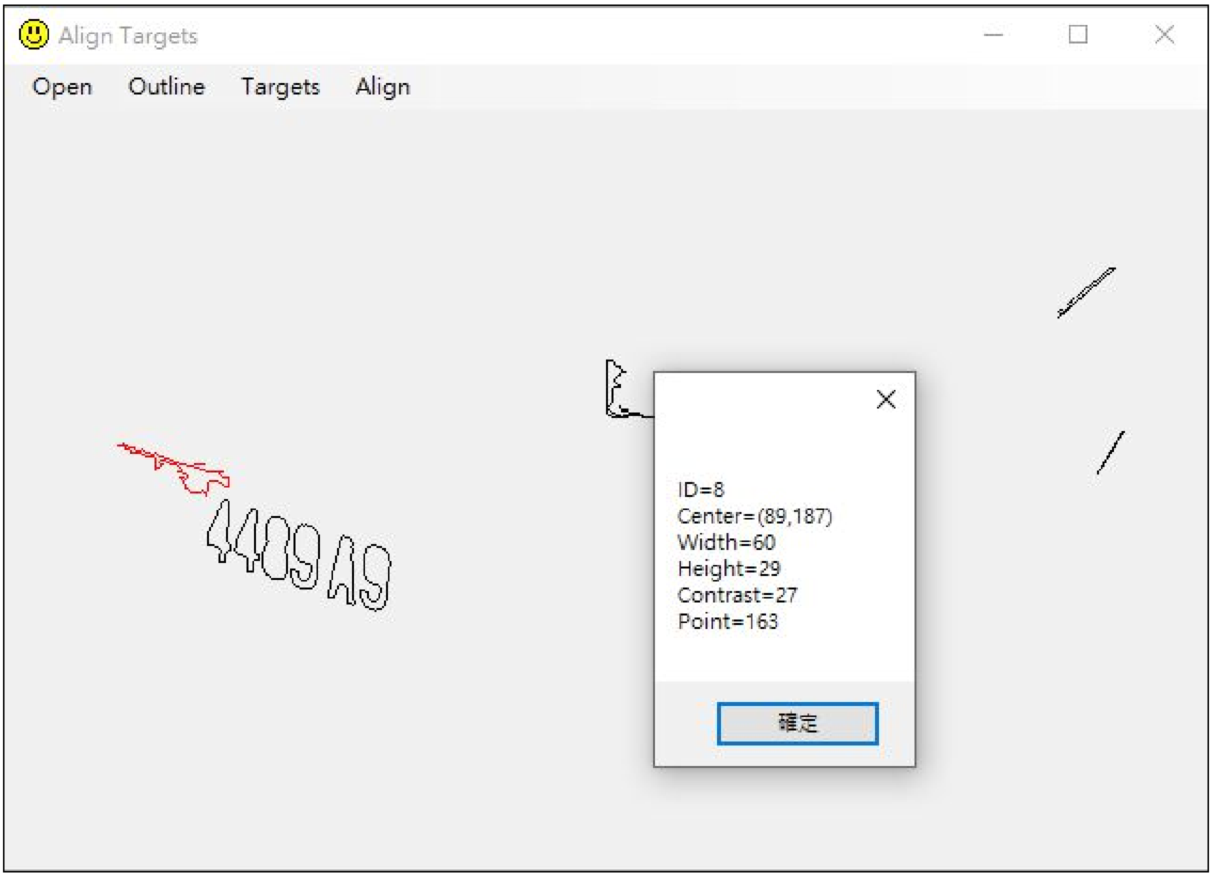
下面是 Align 功能按鍵的程式碼，除了執行 AlignTgs 之外就是繪製結果了！最佳的輸出組合目標會被畫成紅色，中心目標則畫成實心的，搜尋區範圍以綠色方框顯示。

```
    '找車牌字元目標群組
    Private Sub AlignToolStripMenuItem_Click(ByVal sender As Object, ByVal e As EventArgs) _
        Handles AlignToolStripMenuItem.Click
        Dim R As ArrayList = AlignTgs(C) '找到最多七個的字元目標
        Dim bmp As Bitmap = Mb.Clone
        For k As Integer = 0 To R.Count - 1
```

```
            Dim T As TgInfo = R(k)
            For m As Integer = 0 To T.P.Count - 1
                Dim p As Point = T.P(m)
                If k = 0 Then '搜尋的中心目標畫成實心
                    For i As Integer = T.xmn To T.xmx
                        For j As Integer = T.ymn To T.ymx
                            If Z(i, j) Then bmp.SetPixel(i, j, Color.Red)
                        Next
                    Next
                Else '畫輪廓
                    bmp.SetPixel(p.X, p.Y, Color.Red)
                End If
            Next
        Next
        Dim Gr As Graphics = Graphics.FromImage(bmp) '繪製搜尋區
        Gr.DrawRectangle(Pens.Lime, Rec)
        PictureBox1.Image = bmp
    End Sub
```



上圖就是程式執行的結果，唯一的錯誤是左上角的目標多了一個！因為我們無法事先知道會看到幾碼的車牌，所以當然是先收集到最多可能的 7 個目標，但是在此例中實際上只有 6 碼，剛好某些光影造成的假目標也在搜尋範圍之內，那怎麼辦？先點選一下這個目標，看看錯誤目標的屬性，你就會有靈感了！

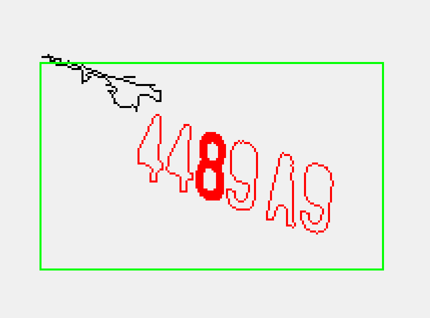


ID=8
Center=(89,187)
Width=60
Height=29
Contrast=27
Point=163

很明顯的，它的寬高形狀與可能的字元差很多，對比度其實也比真的字元低，一般人「一看」就知道不是車牌字元，但我們要怎麼把這個辨識能力變成程式呢？只要在搜尋目標組合時加上以下兩行程式即可：

```
If G.width > T.height Then Continue For '目標寬度太大略過
If G.height > T.height * 1.5 Then Continue For '目標高度太大略過
```

就是如果某目標的寬度比核心字元目標的高度還大時，就不可能是正常字元，應予排除。高度如果大於核心目標的 1.5 倍高，也是明顯不符規格，要排除。寫這些這些寬高排除條件時一定要記得目標文字中可能有「1」或「I」這種很窄的字元，所以一般來說**不要隨便用目標寬度做比較的基準**，而是要用比較可靠的**字元高度**為基準，因為每一個字元的高度不會差很多，寬度就會！加入上述條件之後的辨識結果就會變正確了：



當然，在實際環境影像中甚麼怪事都可能發生！一些奇奇怪怪的雜訊目標都可能混進你的決選名單，所以我賣的商用軟體中，針對排除這些意外的程式不是只有**兩行**，而是有**好幾十行**！就是當你發現這類偷渡客時，就要看它們和合理的字元之間有何差異？如果找到明確的差異特徵，就可以據以寫條件式來把關了！你的程式可以排除越多這種偷渡客，辨識錯誤就會越少，軟體看起來就變聰明了！

完整專案

```
Public Class Form1
    Dim B(,) As Byte '灰階陣列
    Dim Z(,) As Byte '全圖二值化陣列
    Dim Q(,) As Byte '輪廓線陣列
    Dim Gdim As Integer = 40 '計算區域亮度區塊的寬與高
    Dim Th(,) As Integer '每一區塊的平均亮度，二值化門檻值
    Dim C As ArrayList '目標物件集合
    Dim Mb As Bitmap '顯示目標的影像
    '目標物件結構
    Public Structure TgInfo
        Dim np As Integer '目標點數
        Dim P As ArrayList '目標點的集合
        Dim xmn As Short, xmx As Short, ymn As Short, ymx As Short '四面座標極值
        Dim cx As Integer, cy As Integer '目標中心點座標
        Dim width As Integer, height As Integer '寬與高
        Dim pm As Integer '目標與背景的對比強度
        Dim ID As Integer '目標依據對比度的排序
    End Structure
    '開啟檔案
    Private Sub OpenToolStripMenuItem_Click(ByVal sender As Object, ByVal e As System.EventArgs) _
        Handles OpenToolStripMenuItem.Click
        If OpenFileDialog1.ShowDialog = Windows.Forms.DialogResult.OK Then
            Dim bmp As New Bitmap(OpenFileDialog1.FileName)
            Bmp2RGB(bmp) '擷取影像資訊
            B = Gv.Clone '以綠光為灰階
            PictureBox1.Image = bmp '顯示
        End If
    End Sub
    '輪廓線
    Private Sub OutlineToolStripMenuItem_Click(ByVal sender As Object, ByVal e As EventArgs) _
        Handles OutlineToolStripMenuItem.Click
        Z = DoBinary(B) '二值化
        Q = Outline(Z) '建立輪廓點陣列
        PictureBox1.Image = BWImg(Q) '建立輪廓圖
    End Sub
    '二值化
    Private Function DoBinary(ByVal b(,) As Byte) As Byte(,)
        Th = ThresholdBuild(b) '建立二值化使用之門檻值陣列
        Dim Z(nx - 1, ny - 1) As Byte '建立二值化陣列
        For i As Integer = 1 To nx - 2
            Dim x As Integer = i \ Gdim 'x 座標換算
            For j As Integer = 1 To ny - 2
                Dim y As Integer = j \ Gdim 'y 座標換算
                If b(i, j) < Th(x, y) Then
                    Z(i, j) = 1 '低於亮度門檻設為目標點
                End If
            Next
        Next
        Return Z
    End Function
    '門檻值陣列建立
    Private Function ThresholdBuild(ByVal b(,) As Byte) As Integer(,)
        Dim kx As Integer = nx \ Gdim, ky As Integer = ny \ Gdim
        Dim T(kx, ky) As Integer
        '累計各區塊亮度值總和
        For i As Integer = 0 To nx - 1
            Dim x As Integer = i \ Gdim
            For j As Integer = 0 To ny - 1
                Dim y As Integer = j \ Gdim
                T(x, y) += b(i, j) '亮度值累加
            Next
        Next
        '區塊亮度平均值計算
        For i As Integer = 0 To kx - 1
            For j As Integer = 0 To ky - 1
                T(i, j) /= Gdim * Gdim
            Next
        Next
        Return T
    End Function
    '建立輪廓點陣列
    Private Function Outline(ByVal b(,) As Byte) As Byte(,)
        Dim Q(nx - 1, ny - 1) As Byte '輪廓點陣列
        For i As Integer = 1 To nx - 2
            For j As Integer = 1 + 1 To ny - 2
                If b(i, j) = 0 Then Continue For '非輪廓點忽略
                If b(i, j - 1) = 0 Then Q(i, j) = 1 : Continue For '確認為輪廓點
                If b(i - 1, j) = 0 Then Q(i, j) = 1 : Continue For '確認為輪廓點
                If b(i + 1, j) = 0 Then Q(i, j) = 1 : Continue For '確認為輪廓點
                If b(i, j + 1) = 0 Then Q(i, j) = 1 '確認為輪廓點
            Next
        Next
        Return Q
    End Function
    '建立目標物件
    Private Sub TargetsToolStripMenuItem_Click(ByVal sender As Object, ByVal e As System.EventArgs) _
        Handles TargetsToolStripMenuItem.Click
        C = getTargets(Q) '建立目標物件集合
        '繪製目標輪廓點
        Dim bmp As New Bitmap(nx, ny)
        For k As Integer = 0 To C.Count - 1
            Dim T As TgInfo = C(k)
            For m As Integer = 0 To T.P.Count - 1
                Dim p As Point = T.P(m)
                bmp.SetPixel(p.X, p.Y, Color.Black)
            Next
        Next
        PictureBox1.Image = bmp '顯示目標輪廓
        Mb = bmp.Clone
    End Sub
    '以輪廓點建立目標陣列，排除負目標
    Dim minHeight As Integer = 10, maxHeight As Integer = 80 '有效目標高度範圍
    Dim minwidth As Integer = 2, maxWidth As Integer = 80 '有效目標寬度範圍
    Dim Tgmax As Integer = 20 '進入決選範圍的最明顯目標上限
    Function getTargets(ByVal q(,) As Byte) As ArrayList
        Dim A As New ArrayList
        Dim b(,) As Byte = q.Clone '建立輪廓點陣列副本
        For i As Integer = 1 To nx - 2
            For j As Integer = 1 To ny - 2
                If b(i, j) = 0 Then Continue For
                Dim G As New TgInfo
                G.xmn = i : G.xmx = i : G.ymn = j : G.ymx = j : G.P = New ArrayList
                Dim nc As New ArrayList '每一輪搜尋的起點集合
                nc.Add(New Point(i, j)) '輸入之搜尋起點
                G.P.Add(New Point(i, j))
                b(i, j) = 0 '清除此起點之輪廓點標記
                Do
                    Dim nb As ArrayList = nc.Clone '複製此輪之搜尋起點集合
                    nc = New ArrayList '清除準備蒐集下一輪搜尋起點之集合
                    For m As Integer = 0 To nb.Count - 1
                        Dim p As Point = nb(m) '搜尋起點
                        '在此點周邊 3X3 區域內找輪廓點
                        For ii As Integer = p.X - 1 To p.X + 1
                            For jj As Integer = p.Y - 1 To p.Y + 1
                                If b(ii, jj) = 0 Then Continue For '非輪廓點忽略
                                Dim k As New Point(ii, jj) '建立點物件
                                nc.Add(k) '本輪搜尋新增的輪廓點
                                G.P.Add(k) '點集合
                                If ii < G.xmn Then G.xmn = ii
                                If ii > G.xmx Then G.xmx = ii
                                If jj < G.ymn Then G.ymn = jj
                                If jj > G.ymx Then G.ymx = jj
                                b(ii, jj) = 0 '清除輪廓點點標記
                            Next
                        Next
                    Next
                Loop While nc.Count > 0 '此輪搜尋有新發現輪廓點時繼續搜尋
                If Z(i - 1, j) = 1 Then Continue For '排除白色區塊的負目標，起點左邊是黑點
                G.width = G.xmx - G.xmn + 1 '寬度計算
                G.height = G.ymx - G.ymn + 1 '高度計算
                '以寬高大小篩選目標
                If G.height < minHeight Then Continue For
                If G.height > maxHeight Then Continue For
                If G.width < minwidth Then Continue For
                If G.width > maxWidth Then Continue For
                G.cx = (G.xmn + G.xmx) / 2 : G.cy = (G.ymn + G.ymx) / 2 '中心點
                G.np = G.P.Count
                '計算目標的對比度
                For m As Integer = 0 To G.P.Count - 1
                    Dim pm As Integer = PointPm(G.P(m))
                    If pm > G.pm Then G.pm = pm '最高對比度的輪廓點
                Next
                A.Add(G) '加入有效目標集合
            Next
        Next
        '以對比度排序
        For i As Integer = 0 To A.Count - 2
            For j As Integer = i + 1 To A.Count - 1
                Dim T As TgInfo = A(i), G As TgInfo = A(j)
                If T.pm < G.pm Then A(i) = G : A(j) = T '互換位置，高對比目標在前
            Next
        Next
        '取得 Tgmax 個最明顯的目標輸出
        Dim C As New ArrayList
        For i As Integer = 0 To Tgmax - 1
            If i > A.Count - 1 Then Exit For '超過總目標數
            Dim T As TgInfo = A(i) : T.ID = i '建立以對比度排序的序號
            C.Add(T)
        Next
        Return C '回傳目標物件集合
    End Function
```

```
    '輪廓點與背景的對比度
    Private Function PointPm(ByVal p As Point) As Integer
        Dim x As Integer = p.X, y As Integer = p.Y
        Dim mx As Integer = 0 '周邊最亮點，依據灰階陣列 B
        If mx < B(x - 1, y) Then mx = B(x - 1, y)
        If mx < B(x + 1, y) Then mx = B(x + 1, y)
        If mx < B(x, y + 1) Then mx = B(x, y + 1)
        If mx < B(x, y - 1) Then mx = B(x, y - 1)
        Return mx - B(x, y) '最亮點與輪廓點的差值
    End Function
    '儲存目前影像
    Private Sub SaveImageToolStripMenuItem_Click(ByVal sender As Object, ByVal e As System.EventArgs)
_
        Handles SaveImageToolStripMenuItem.Click
        If SaveFileDialog1.ShowDialog = Windows.Forms.DialogResult.OK Then
            PictureBox1.Image.Save(SaveFileDialog1.FileName)
        End If
    End Sub
    '點選目標顯示位置與屬性
    Private Sub PictureBox1_MouseDown(ByVal sender As Object, ByVal e As
Windows.Forms.MouseEventArgs) _
        Handles PictureBox1.MouseDown
        If IsNothing(Mb) Then Exit Sub
        If e.Button = MouseButtons.Left Then
            Dim m As Integer = -1
            For k As Integer = 0 To C.Count - 1
                Dim T As TgInfo = C(k)
                If e.X < T.xmn Then Continue For
                If e.X > T.xmx Then Continue For
                If e.Y < T.ymn Then Continue For
                If e.Y > T.ymx Then Continue For
                m = k '被點選目標
                Exit For
            Next
            If m >= 0 Then '有被選目標時
                Dim bmp As Bitmap = Mb.Clone
                Dim T As TgInfo = C(m)
                For k As Integer = 0 To T.P.Count - 1
                    Dim p As Point = T.P(k)
                    bmp.SetPixel(p.X, p.Y, Color.Red)
                Next
                PictureBox1.Image = bmp
                '指定目標的資訊
                Dim S As String = "ID=" + T.ID.ToString
                S += vbNewLine + "Center=(" + T.cx.ToString + "," + T.cy.ToString + ")"
                S += vbNewLine + "Width=" + T.width.ToString
                S += vbNewLine + "Height=" + T.height.ToString
                S += vbNewLine + "Contrast=" + T.pm.ToString
                S += vbNewLine + "Points=" + T.np.ToString
                MsgBox(S)
            End If
        End If
    End Sub
    '找車牌字元目標群組
    Private Sub AlignToolStripMenuItem_Click(ByVal sender As Object, ByVal e As EventArgs) _
        Handles AlignToolStripMenuItem.Click
        Dim R As ArrayList = AlignTgs(C) '找到最多七個的字元目標
        Dim bmp As Bitmap = Mb.Clone
        For k As Integer = 0 To R.Count - 1
            Dim T As TgInfo = R(k)
            For m As Integer = 0 To T.P.Count - 1
                Dim p As Point = T.P(m)
                If k = 0 Then '搜尋的中心目標畫成實心
                    For i As Integer = T.xmn To T.xmx
                        For j As Integer = T.ymn To T.ymx
                            If Z(i, j) Then bmp.SetPixel(i, j, Color.Red)
                        Next
                    Next
                Else '畫輪廓
                    bmp.SetPixel(p.X, p.Y, Color.Red)
                End If
            Next
        Next
        Dim Gr As Graphics = Graphics.FromImage(bmp) '繪製搜尋區
        Gr.DrawRectangle(Pens.Lime, Rec)
        PictureBox1.Image = bmp
    End Sub
    '找車牌字元目標群組
    Dim Rec As Rectangle
    Private Function AlignTgs(ByVal C As ArrayList) As ArrayList
        Dim R As New ArrayList, pmx As Integer = 0 '最佳目標組合與最佳度比度
        For i As Integer = 0 To C.Count - 1
            Dim T As TgInfo = C(i) '核心目標
            Dim D As New ArrayList, Dm As Integer = 0 '此輪搜尋的目標集合
            D.Add(T) : Dm = T.pm '加入搜尋起點目標
            Dim x1 As Integer = T.cx - T.height * 2.5, x2 As Integer = T.cx + T.height * 2.5 '搜尋 X 範圍
            Dim y1 As Integer = T.cy - T.height * 1.25, y2 As Integer = T.cy + T.height * 1.25 '搜尋 Y 範圍
            For j As Integer = 0 To C.Count - 1
                If i = j Then Continue For '與起點重複略過
                Dim G As TgInfo = C(j)
                If G.cx < x1 Then Continue For
                If G.cx > x2 Then Continue For
                If G.cy < y1 Then Continue For
                If G.cy > y2 Then Continue For
                If G.width > T.height Then Continue For '目標寬度太大略過
                If G.height > T.height * 1.5 Then Continue For '目標高度太大略過
                D.Add(G) : Dm += G.pm '合格目標加入集合
                If D.Count >= 7 Then Exit For '目標蒐集個數已滿跳離迴圈
            Next
            If Dm > pmx Then '對比度高於之前的目標集合
                pmx = Dm : R = D
                Rec = New Rectangle(x1, y1, x2 - x1 + 1, y2 - y1 + 1) '搜尋範圍
            End If
        Next
        Return R
    End Function
End Class
```