

組合語言期末報告

-簡單的圖像處理

組員:

B96902128 廖國平

B96902130 張軍裕



前言:

由於我們花了一半的時間在研究寫視窗，所以我們就只實作出了三個功能，而且只能支援 24 位元的 bitmap 格式

Bitmap 點陣圖格式介紹:

由四個部分組成：Bitmap file header, Bitmap Info header, color

Table(Palette) 及 Bitmap array

0000h~0001h: identifier(ID)

0002h~0005h: 點陣圖檔案的大小

0006h~0009h: 這是保留欄位

000Ah~000Dh: bitmap array 在檔案中的開始位置

000Eh~0011h: bitmap info header 長度(通常為 28h)

00012h~0015h: 點陣圖的寬度, 以像素為單位

0016h~0019h: 點陣圖的高度, 以像素為單位

001Ah~001Bh: 點陣圖為元圖層數(通常是 1)

001Ch~001Dh: 每個像素的位元數

001Eh~0021h: 壓縮方式

0022h~0025h: 點陣圖資料大小, 以 byte 為單位

0036h~ : 調色盤資料, 調色盤後就是點陣圖資料了, 而 24 位元的點陣圖是沒有調色盤資料的, 0036h 開始就是點陣圖的資料

實作部分

黑白:

就是讓每一個 pixel 的 $r=g=b$, 首先將原檔案一個 pixel 的 r, g, b 拿出來乘上各自的比例加起來, 然後就讓 r, g, b 都等於這個值, 放進目的檔案

R, G, B 的比例是 30:59:11, 所以就是

$$R*0.30 + G*0.59 + B*0.11$$

因為運算上的方便, 所以我是將他們的比例先乘上 256, 算完後才除回 256, 所以實際上的運算是

$$(R*77 + G*151 + B*28)/256$$



縮小:

我們用的是一個比較簡單直接的做法, 就是只取奇數列, 然後只取每一列第偶數位的像素, 所以這種做法會使得出來的圖檔會是原圖的 1/4 大小

@縮小一次



@縮小兩次



轉 90 度:

我們是把原檔案的第一列放去目的檔案的最後一行, 第二列放去最後第二行, 以此類推, 因為點陣圖的資料是反著來放的, 所以這種處理的方法會使圖向左轉 90 度, 而不是向右轉 90 度



因為點陣圖資料每一列的位元數一定要是四的倍數, 當資料不是四的倍數時, 就要在每一列的後面補上 1~3 個 byte, 所以在讀取檔案時都要略過後面那些多餘的資料

操作方法:

在 command prompt 上做操作,

在命令列上輸入 <執行檔> <源圖檔> <option>

<option> 有三個選項:

s 代表縮小

t 代表轉 90 度

b 代表黑白

結語:

這次的報告可以說是挫折重重, 原本要用 C 加 Intel 寫個拳王遊戲, 但是卻因為寫視窗失敗而作罷, 用僅剩的一天半做好這個簡單的圖像處理。所以我覺得還蠻可惜沒辦法做出比較多的功能及支援更多的圖檔格式, 也沒有時間研究用 SIMD 的方式來寫。

參考資料：

<http://crazycat1130.pixnet.net/blog/post/1345538>

<http://sunh.hosp.ncku.edu.tw/~cww/htmap45.htm>