

DCC 期末專題

資工所 R93922039 李沛倫

資工所 R93922016 沈允中

題目: **Secret image communication scheme based on vector quantisation**

作者: **S.D. Lin and S.-C. Shie**

出處: **ELECTRONICS LETTERS 8TH July 2004 Vol.40 No.14**

這篇 paper 提出了一個傳送機密影像的方法。主要的概念是將數張機密影像利用一個掩飾的資料媒介(cover media)傳送出去來達到保密的目的。使用到的現有技術包括: Vector Quantisation 和 DES 加密技術。關鍵處在於作者用了 adaptive least-significant-bits modification 的技巧將 VQ 壓縮過的影像資料(即 codeword 的 index)放進 VQ codebook, 而此修改過的 codebook 在經過 DES 加密後即成為最後的 cover media。

通常在隱藏影像資料的時候, 會採用的方法是在不太影響影像品質的情況下盡可能的把要隱藏的影像放進掩飾的媒介裡。因此, 容量和品質之間的取舍就成了一個考量。到目前為止, Hu, Y.C. 提出的 'Grey-level image hiding scheme based on vector quantisation' 在隱藏資訊的容量方面是最好的, 他的方法是將數張影像嵌入另一張掩飾的影像中, 但是在還原的影像品質方面就有所不足。而我們所報告的這篇 paper 所提出的方法, 就是想達到高容量且高品質的目標。為了達到到較好的影像品質, 作者選擇用 VQ codebook 來當作 cover media。

以下將詳細說明這篇 paper 所用的方法。首先說明會用到的參數: t 是原始影像的數目, 大小同為 $w \times h$ pixels, VQ codebook 大小為 N_c , 每個 codeword 大小為 $m \times n$ pixels, 因此經過壓縮的資料大小就變成 $l = t \times (w/m) \times (h/n) \times (\log_2 N_c)$ 。

經過 VQ training 後, 為了更進一步得到更小的資料, 作者提出了一個方法把參數 t 、 w 、 h 、 m 、 n 、 N_c 和壓縮過的資料嵌入 codebook, 使最後加密的檔案大小等於 codebook 大小, 而且不置於產生肉眼能察覺的失真。

嵌入的方法如下:

1. 首先要將 codebook 中的 codeword 根據使用的頻率作排序, 由小到大, 也就是把最少被用到、最不重要的 codeword 放在最前面。
2. 根據壓縮資料的大小 l , 從排序過的 codeword 中選擇前面 $p+k$ 個出來, 前 p 個用來嵌入相關的參數(t 、 w 、 h 、 m 、 n 、 N_c), 而接下來 k 個 codeword 則用來嵌入壓縮過的資料。
3. 將參數及壓縮資料嵌入 codeword 的方法是將要嵌入的資料拆成一個個 bit, 然後直接填入 codeword 中的 least significant bits (LSBs)。

其中嵌入的參數為固定個數, 而壓縮資料每一筆只需要 $\log_2 N_c$ 個 bit。由於 codebook 被破壞的部分是被引用到最少的 codeword, 而且修改的部分又是那些 codeword 的 LSB, 所以實際上對影像造成的破壞非常的小。

此一經過修改的 `codebook` 最後在用 DES 加密之後，就成為最後送出的資料串流。

而要將影像還原回來也非常的簡單，首先用 DES 解密後得到 `codebook`，再從此 `codebook` 中的 LSB 擷取出相關的參數，然後就可以依據這些參數繼續將壓縮的資料也傑取出來，最後再查詢 `codebook` 就可以重建原來的影像了。

實做：

我們在 `unix` 平台上用 C 語言實做了這套系統，包括 VQ 和 LSB 修改的部分。DES 加密/解密由於並不會影響到影像的內容，因此我們並沒有使用。我們將程式用在一張 512*512 的灰階(256 levels)影像(`lena`)上，VQ 的 `codeword` 大小選為 8*8，結果如下：

下圖為原始的影像：



下圖爲用 VQ 壓縮的影像，codebook 沒有嵌入任何資訊





上圖為嵌入參數及壓縮資料後的 codebook 所還原的影像

由結果的影像可看出用這篇 paper 的方法來嵌入資訊到 codebook 中並不會造成影像的二度失真。事實上，經過我們的計算，有無嵌入資訊的影像彼此間的方均根差(RMS error)為 0.125(單位為灰階值)，可說是幾乎沒有差別，而且資料大小又可以大幅縮小，因此是一個不錯的方法。