國立中央大學資訊工程學系

資料壓縮期末報告

7z format所使用的LZMA

梁中瀚

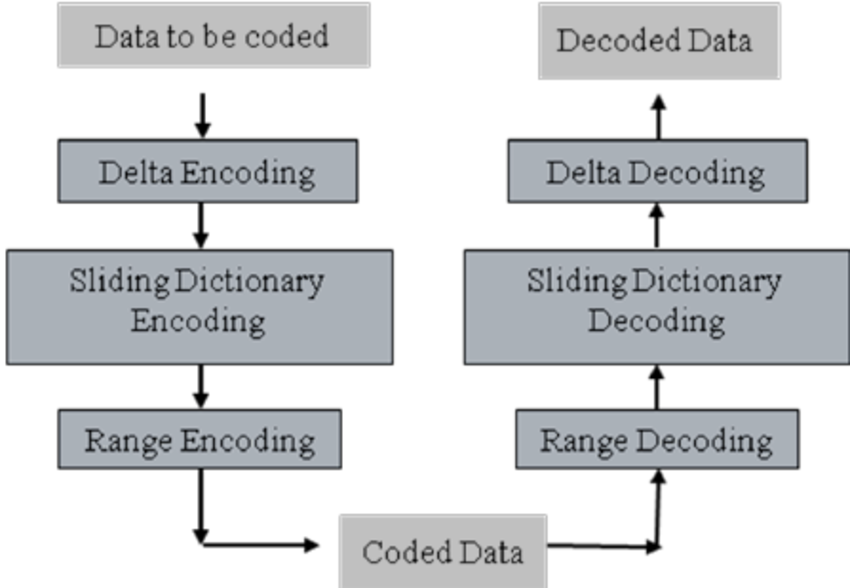
資工三

一、前言

Lzma為7z壓縮格式所支援的壓縮演算法，而7z又是近期大多數人會使用到的壓縮工具，並且大多數人都是直接將檔案放到壓縮檔裡面壓縮，實際測試先將圖片壓縮一次後，再放到lzma壓縮，與只用一次圖片壓縮的方式比較，觀察其蜂鳴值的大小。

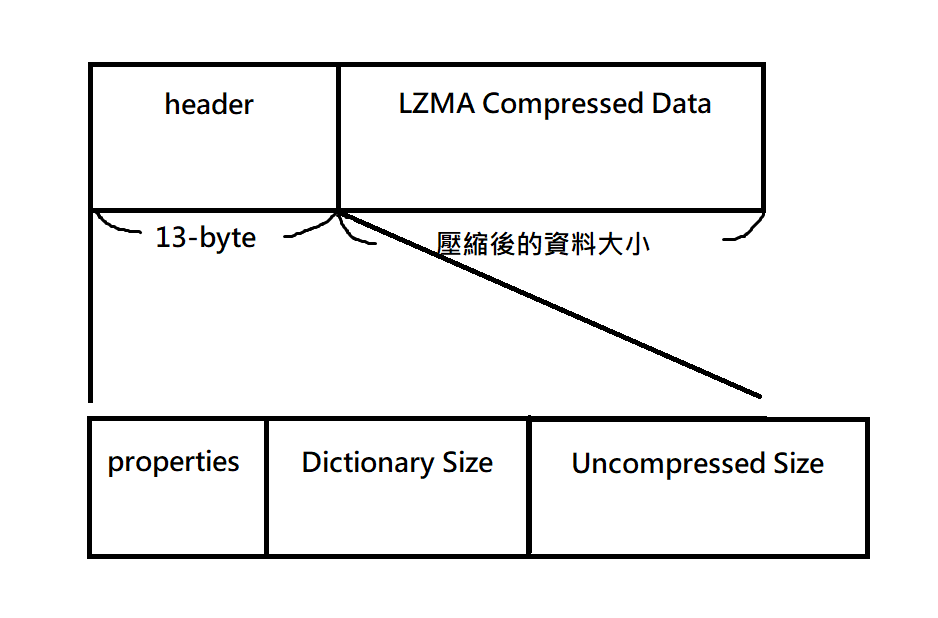
Lzma是基於lz77的壓縮演算法，並且結合range coding(類似arithmetic encoding，但是可以用非二進位來壓縮)。

壓縮流程:



二、Lzma的壓縮後格式:

.lzma檔案就是被lzma壓縮後的資料格式，它包含了兩個部分，header以及LZMA Compressed Data，如下圖。



(一)、header內部構造:

1. Properties:

Properties內部由三個變數所組成:

1. lc: literal context的bits數量，範圍是[0, 8]

之前的byte當中有多少是1的bit會拿來當作context(前後文)。

1. lp: literal position的bits數量，範圍是[0, 4]
2. pb: postion的bits數量，範圍是[0, 4]

最後用這個式子把以上三個變數encode:

Properties = (pb \* 5 + lp) \* 9 + lc

Decode就會用以下的sudo code來達成:

pb = properties / (9 \* 5);

properties -= pb \* 9 \* 5;

lp = properties / 9;

lc = properties - lp \* 9;

2. Dictionary Size:

用unsigned 32-bit little endian integer來儲存字典大小，所以總共是4個byte。

但是為了提供好的移植性(不同硬體)，應該只用2^n或者是2^n+2^(n-1)這兩種size。

3. Uncompressed Size:

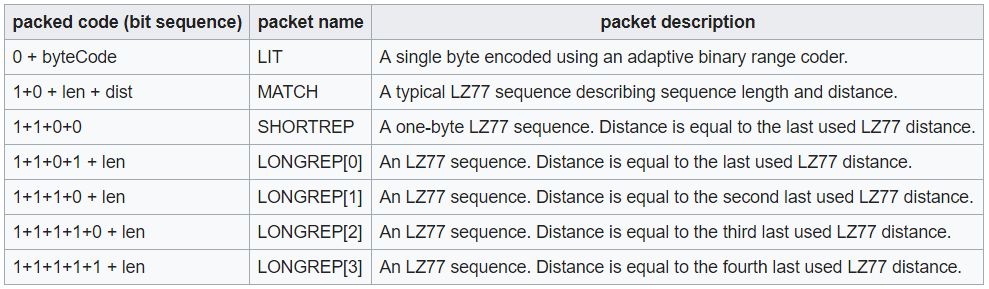
未壓縮大小是用unsigned 64-bit little endian integer來儲存的。

比較特別的是，如果64個bit都是1的話，代表未壓縮大小是未知的。

(二)、LZMA Compressed Data

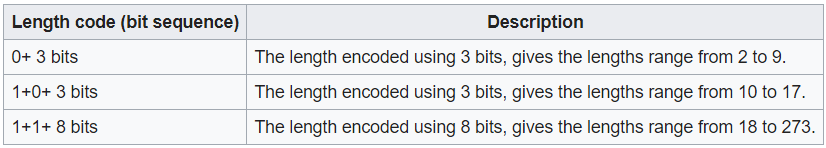
經由LZMA壓縮過後的資料會存放在這個區塊。

壓縮後的形式為bit stream，並且用 adaptive binary range coder做encode，而bit stream會被劃分成很多packets，packets的形式如下:

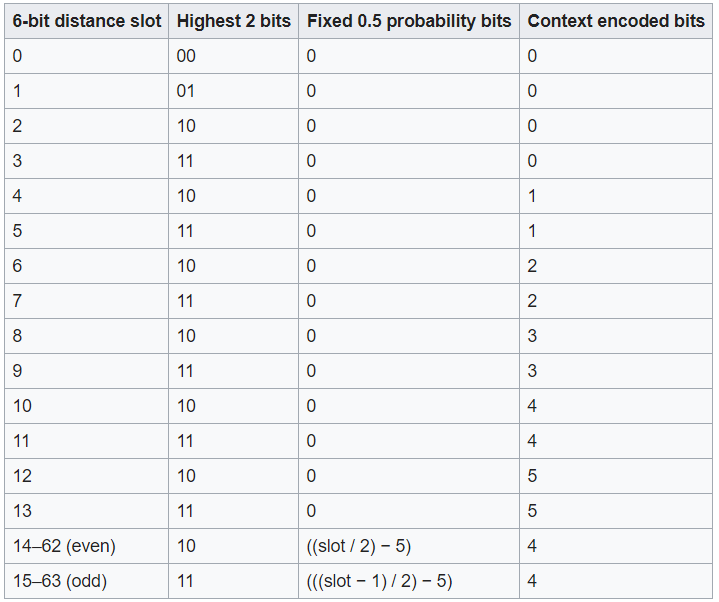


1. LIT: literal，代表單獨的byte並且用adaptive binary range coder encode。
2. MATCH: 標準的LZ77格式，有長度(在window找到look ahead buffer的code的長度)還有距離(在window找到look ahead buffer的code出現在window的哪裡)
3. SHORTREP: 一個byte的LZ77序列，距離等同於上一個LZ77使用的距離
4. LONGREP[n]: 一個LZ77的序列，並且距離等同於上n個LZ77的距離

長度的encode用以下的表格:



Distance用以下表格編碼:



每一個distance都會用6-bit distance slot作為起始， 6-bit distance slot就是後面會再接多少bit的意思。

三、lzma、jpeg與先壓縮成jpeg再放到lzma壓縮Lena.raw的比較

參考:

[1]: <https://dev.twsiyuan.com/2018/06/how-to-compress-and-decompress-gamesaves-in-unity.html>

[2]: <http://html.rhhz.net/BJHKHTDXXBZRB/20150302.htm#R-4>

[3]: <https://gautiersblog.blogspot.com/2016/08/lzma-compression.html>