Proposed Lossy Compression Method Using RLE and Segmentation

Tawfiq A. Abbas Al-Asadi **Safaa O. Mahdi**

Department of Computer Science, University of Babylon.

1. Abstract

The idea from this research is to combine the segmentation techniques with lossy RLE to produce a lossy compression algorithm. The first step is to apply the segmentation techniques on the gray image (256 colors) to reduce the gray levels of the image and approximate the values of neighbor's pixels, the next one is applying the proposed lossy RLE on the segmented image to achieve high compression ratio.

الخلاصة

يهدف هذا البحث لدمج تقنيات تقطيع الصورة مع خوارزمية RLE المقترحة ذات الفقدان بالبيانات للحصول على خوارزمية ضغط للصور مع فقدان البيانات. الخطوة الأولى هي تطبيق تقنيات التقطيع على صورة رمادية (256 لون) لتقليل التدرجات الرمادية للصورة وجعل القيم المتجاورة متقاربة أو متشابهة. أما الخطوة الثانية فهي تطبيق خوارزمية RLE المقترحة ذات الفقدان بالبيانات على الصورة المقطعة للحصول على أعلى نسبة ضغط.

2. Introduction

Data compression is achieved by reducing redundancy in data but this may make the data less reliable and more prone to errors. Image compression techniques can be broadly categorized into types, lossless (or reversible) compression, and lossy (or irreversible) compression. Lossless schemes allow exact recovery of the original image from the compressed data with compression of up 2:1 like JPEG, LZW and EZW method (Joonmi, 1999).

Alternatively, lossy schemes provide an approximation to the original image, but it can achieve many higher compression ratios. In general, large compression ratios are obtained at the expense of image degradation.

To determine exactly what information is important and to be able to measure image fidelity, it is needed to define an image fidelity criterion. Note that the information required is application specific, and that, with lossless schemes, there is no need for a fidelity criterion. One of the fidelity criterions is signal-to-noise ratio and defined as follow:

$$SNR_{Peak} = 10 \log_{10} \frac{(L-1)^2}{\frac{1}{N^2}\sum_{r=0}^{N-1}\sum_{c=0}^{N-1}\left[g(r,c)-I(r,c)\right]^2}$$

3. RLE

The RLE technique can be used for lossy image compression by reducing the number of gray levels and then applying standard RLE techniques. For this method to be effective, the reduced image data need to be mapped to a gray code.

The idea behind this approach to data compression is: if a data item ***D*** occurs ***N*** consecutive times in the input stream, replace the ***N*** occurrences with the single pair

ND. ***N*** consecutive occurrences of a data item are called a run length of ***D***, and this approach to data compression is called run length encoding or RLE (Rhys, 2002).

The block-based coding scheme of RLE can be coupled with the Discrete Cosine Transform (DCT) (Pennebaker, 1992).The beauty of this coding scheme is its simplicity, low computational complexity, low memory requirement, and flexibility on a block-by-block basis (Chengjie, 2001).

4. Segmentation

The goal of image segmentation is to find regions that represent objects or meaningful parts of objects. Division of the image into regions corresponding to objects of interest is necessary before any processing can be done at a level higher than that of the pixel. Identifying real objects, pseudo-objects, and shadows or actually finding anything of interest within the image requires some form of segmentation.

Image segmentation methods will look for objects that either have some measure of homogeneity within them or have some measure of contrast with the objects on their border.

The major problems are a result of noise in the image and digitization of a continuous image. We can divide image segmentation techniques into three main categories: region growing and shrinking, clustering methods and boundary detection. Optimal image segmentation is likely to be achieved by focusing on the application and on how the different methods can be used, singly or in combination, to achieve the desired results (Scott,1998).

5. Scan Methods

Compressing an image using RLE is based on observation that if we select a pixel in the image at random, there is a good chance that its neighbors will have the same color. The compressor thus scans the bitmap, row by row, looking for runs of pixels of the same color.

The output of RLE can sometimes be bigger than pixel-by-pixel storage for complex pictures. Imagine an image with many vertical lines. When it is scanned horizontally, it produces very short runs, resulting in very bad compression, or even expansion. A good, practical RLE image compressor should be able to scan the bitmap by rows, columns, or in zigzag and it may automatically try all three ways on every bitmap compressed to achieve the best compression (David, 1998).

6. The Proposed Method

Each compression system model consists of two parts: the compression stage and the decompression stage. And they are stated as follows.

The Compression Stage

It is possible to get even better compression ratios if short runs are enlarged. Such a method loses information when compressing an image but, sometimes, this is acceptable to the user. Figure (1) shows how the proposed compression system should work:

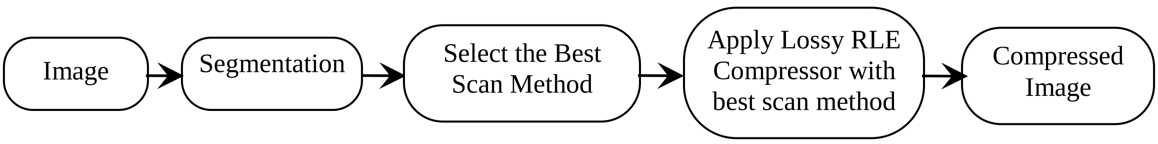

Figure (1) The Compression Stage

The first step of the proposed compression system is to apply segmentation on the input image to reduce the gray levels of the image and approximate the values of neighbors pixels.

The next step is selecting the best scan method to achieve the best compression ratios. After that, we apply the proposed lossy RLE with best scan method. The proposed Lossy RLE relaxes the criterion of the runs being the same value and allows for the runs to fall within a gray-level range. The user specify this range, the bigger the range the good compression ratio.

Example:

The format of the encoded data is (Data, Occurrence). Assume the following:

- Data is (5, 8, 7, 5, 3, 4, 1, 33).
- Tolerance is 6 (The range value).

The first value is 5, so when we find the second value is 8, we compute MAX and MIN:

$$MAX = 5 + (Tolerance/2) = 5 +(6 / 2) = 8$$
$$MIN = 5 - (Tolerance/2) = 5 -(6 / 2) = 2$$

We consider the values in the range [2,8] as a single run. So we ask:

$$MIN <= 8 <= MAX$$

And the result is true, so we consider the second value (8) as 5 and increment the count. Else, we assume there is no run and put the previous value (5) in the out stream and consider (8) as the new value. This process will be continues until we reach the end of data. Then the out stream is (5, 6, 1, 1, 33, 1).

The Decompression Stage

Figure (2) explains the main steps of the decompression system:

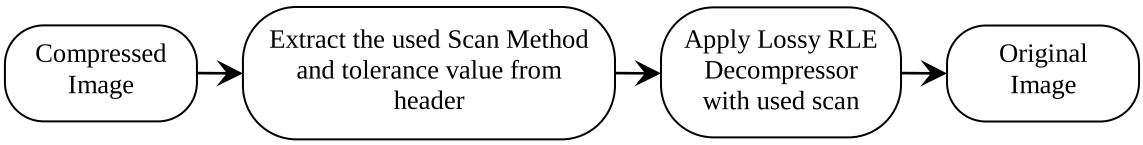

Figure (2) The Decompression Stage

At the beginning, the decompressor extracts the tolerance value (range value) and the used scan method from the header of compressed image. After that, we apply the proposed lossy RLE and obtain the original Image.

In the example above, if we apply the standard decompressor of RLE, we obtain the stream (5, 5, 5, 5, 5, 5, 1, 33).

7. Results and Discussion

It can be noted the following points from our research:

1. The segmentation techniques are suitable for RLE method by approximating the values of neighbor's pixels.
2. The use of best scan method will produce high compression ratio.

The images in Figure (3) and Table1 show the decompressed image and compression results, respectively, for different Tolerance values:

Segmented Image

Image With Tolerance=5

Image With Tolerance=10

Image With Tolerance=15

Image With Tolerance=20

Image With Tolerance=25

Image With Tolerance=30

Image With Tolerance=35

Image With Tolerance=40

Figure (3) Decompressed images for different tolerance values

Tolerance	Compression Rate	RMSE	SNR Peak
5	69.251	0.645	29.509
10	70.041	1.365	26.254
15	70.345	1.567	25.652
20	70.418	1.685	25.339
25	70.722	2.381	23.837
30	71.208	3.056	22.753
35	75.109	7.822	18.671
40	75.365	7.963	18.594

Table (1) Compression results for different tolerance values

8. References

Chengjie T., Jie L. & Trac D., "**Adaptive Run-Length Coding**", IEEE,2001.

David S., "**Data Compression The Complete Reference**", Springer-Verlag New York, 1998.

Joonmi O., "**A Review on Medical Image Compression and Storage Options**", University of Birmingham, 1999.

Pennebaker W., & Mitchell J., "**JPEG Still Image Data Compression**", Van Norstand Reinhold, New York, 1992.

Rhys H., "**Digital Stereo Video: Display Compression and Transmission**", Master thesis, The Australian National University, 2002.

Scott V., "**Computer Vision and Image Processing a Practical Approach Using CVIP Tools**", Prentice Hall, 1998.