Employing hybrid methods for compression color images Run Length Encoding ضغط الصور الملونة باستخدام خوارزمية K-Means وخوارزمية

Assist. Prof. Dr.Baheja K.Shukur
Baheja2003@yahoo.com
Iraq/Babylon University
Computer Technology /software department

Abstract

The purpose of compressing images to represent the image less to provide cost data storage and transmission time, however, the effectiveness of pressure accomplished by bringing the original image (instead of the exact loss).

Therefore clustering problem defined as a collection of objects which are “similar” between them and are “dissimilar” to the objects belonging to other clusters; so, as every other problem of this kind, it deals with finding a structure in a collection of unlabeled data that providing a novel solution to the color image compression by exploiting the ability to generating groups of data by using k-mean algorithm.

In this study true color images converted to YIQ color space then k-mean algorithm applied on the Y component to determine the number of cluster which used to constructing the clustered images. Run length encoding (RLE) algorithm applied on the resulting YIQ clustering images. In decompression stage RLE decompression algorithm

used to reconstruct the RGB color images. Number of quality measurement computed like (peak signal to noise ratio (PSNR), (mean square error) MSE and signal to noise ratio (SNR) to measure the amount of distortion in this processes also the compression ratio calculated.

Keywords: Image clustering, Image Compression, RGB, YIQ, K-means Algorithm, RLE compression.

الخلاصة

الغرض من ضغط الصور لتمثيل الصورة بأقل بيانات لتوفير كلفة الخزن ووقت الإرسال, ومع ذلك فعالية الضغط تنجز بتقريب للصورة الأصلية(بدلا من ضياعها بالضبط),ولهذا السبب مشكلة الغنقدة والتي تعرف هي تجميع للبيانات المتشابهه مع بعضها والتي تختلف عن البيانات التي تنتمي الى عناقيد اخرى حيث تم استغلال هذة الخاصية لتوفير طريقة لضغط الصور الملونة. في هذا البحث تم استخدام صورة ملونة ومن ثم تم تحويلها من صيغة الــ RGB إلى صيغة YIQ ومن ثم تم استخدام خوارزمية K-meanعلى مركبة Y بتحديد عدد العناقيد ولتقليل عملية الحساب, وبالنهاية استخدم خوارزمية RLE على المكونات الناتجة YIQ ومن جانب أخر استخدمنا خوارزمية إعادة فك الضغط لخوارزمية RLE للحصول على YIQ وبناء صورة ملونة بمساحة لونية RGB وبعد ذلك حساب MSR PSNR, لقياس كمية التشويه وإيجاد نسبة الضغط.

1. Introduction

In an image there is usually a likelihood of high correlation between pixels. Such correlations between pixels or a block

Of pixels are exploited to achieve image compression [1].

Finding the optimal way to compress data with respect to resource constraints remains one of the most challenging problems in the field of data compression. There are two data compression strategies: lossy and lossless [2].

There exist many, quite different, compression techniques, corresponding to different ways of detecting and using departures from equiprobability in input strings, such as Run-Length Encoding, Huffman, Delta, LZW, GOLOMB, CS&Q, etc. [4].

Clustering is a process that groups similar objects into the same cluster. Various clustering algorithms have been designed to fit various requirements and constraints of application [5], which assigns a large number of data points to a smaller number of groups such that data points in the same group share the same properties while, in different groups, they are dissimilar[6]. Clustering has many applications, including part family formation for group technology, image segmentation, information retrieval, web pages grouping, market segmentation, scientific, compression and engineering analysis [7].

Many clustering methods have been proposed and they can be broadly classified into four categories [8]: partitioning methods, hierarchical methods, density-based methods and grid based methods. Other clustering techniques that do not fir in these categories have been developed. They are fuzzy clustering, artificial neural networks and generic algorithms [9].

In this study, an image clustering and compression method based on K-means and Run Length Encode was introduced for color images together with conversion the RGB to YIQ color space and image segmentation based on K-means then applying the RLE to compression stage. This method was applied to different sample images and high compression ratios and good validity measures were observed.

2. Color Spaces

A color space is a mathematical representation of a set of colors. The three most popular color models are RGB; YIQ, YUV, YCbCr; and CMYK. All of the color spaces can be derived from the RGB information supplied by devices such as cameras and scanners. The red, green, and blue (RGB) color space is widely used throughout computer graphics. Red, green, and blue are three primary additive colors (individual components are added together to form a desired color) and represented by a three-dimensional Cartesian coordinate system. The RGB color space is the most prevalent choice for computer graphics because color displays use red, green, and blue to create the desired color. Therefore, the choice of the RGB color space simplifies the architecture and design of the system [10].

In the RGB model, each color appears in its primary spectral components of red, green, and blue [11]. The reason, is red(R), green (G), and blue (B) are described as the primary colors of the additive color system. All colors can be obtained in this way; the technique is a powerful one, nonetheless. It would seem to suggest that a color image can be formed by making three measurements of scene brightness at each pixel, using the red, green and blue components of the detected light. We can do this by using a color camera in which the sensor is able to measure radiation at red, green, and blue wavelengths for all points in the image, or by using a monochrome camera in conjunction with three special filters that block all but a narrow band of wavelengths centered on red, green, and blue [12].

In color image conforming to the RGB model, the value of f(x, y) is a vector with three components, corresponding to R, G, and B, the components each vary between 0.0. And1.0. R, G and B can be regarded defining a three- dimensional color space. The value of f (x.y) is a point in this ‘color cube’.

The primary colors red, green and blue are at the corners; black is at the origin, white is at the corner furthest from the origin, see fig.2. The three components red, green, and blue are quantized using 8 bits; an image made up of these components is commonly described as 24-bit color image. Because each primary color is represented to a precision of 1 part in 256, we can specify an arbitrary color to a precision of about 1 part in 16 million; that is around 16 million colors are available in a 24-bit image [12].

Figure1: The RGB Space

2.1 YIQ Space

YIQ is used in the U.S. television standard, NTSC (National Television System Committee). It is similar to YUV, except that its color space is rotated 33 degrees clockwise, so that I is the orange-blue axis, and Q is the purple-green axis. The equations to find this conversion shown in below equations :

$$Y = 0.299R + 0.587G + 0.114B \tag{1}$$

$$I = 0.74(R - Y) - 0.27(B - Y) = 0.596R - 0.275G - 0.321B \tag{2}$$

$$Q = 0.48(R - Y) + 0.41(B - Y) = 0.212R - 0.523G + 0.311B \tag{3}$$

3. RLE Compression

Run length encoding, sometimes called recurrence coding, is one of the simplest data compression algorithms. It is effective for data sets that are comprised of long sequences of a single repeated character. For instance, text files with large runs of spaces or tabs may compress well with this algorithm. RLE finds runs of repeated characters in the input stream and replaces them with a three-byte code. The code consists of a flag character, a count byte, and the repeated characters. For instance, the string "AAAAAABBBBCCCCC" could be more efficiently represented as "A6B4C5". That saves us six bytes. Of course, since it does not make sense to represent runs less than three characters in length with a code, none is used. Thus "AAAAAABBCCCDDDD" might be represented as "A6B2C3D4". The flag byte is called a sentinel byte [13].

Run-length encoding can be used on only one of the characters (as with the zero below), several of the characters, or all of the characters.

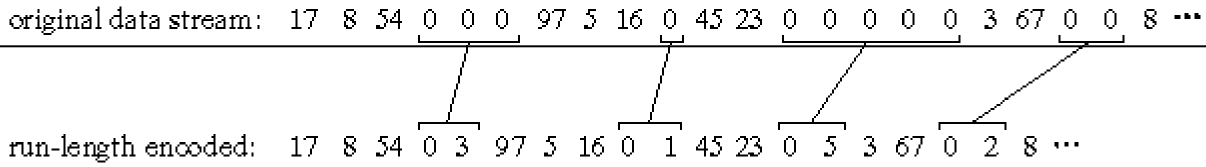

Figure 2: show Run Length Encoding Example

Example of run-length encoding, each run of zeros is replaced by two characters in the compressed file: a zero to indicate that compression is occurring, followed by the number of zeros in the run.

4. The structure of the suggested System

The input color image converted from RGB to YIQ color space, Then applied k-mean algorithm to determine the number of the cluster centers, in addition to that RLE algorithm used to compress the three color space YIQ each one separately then decompress the compression image and compute the PSNR and MSE, SNR to measure the quality of this process, as shown in figure 3

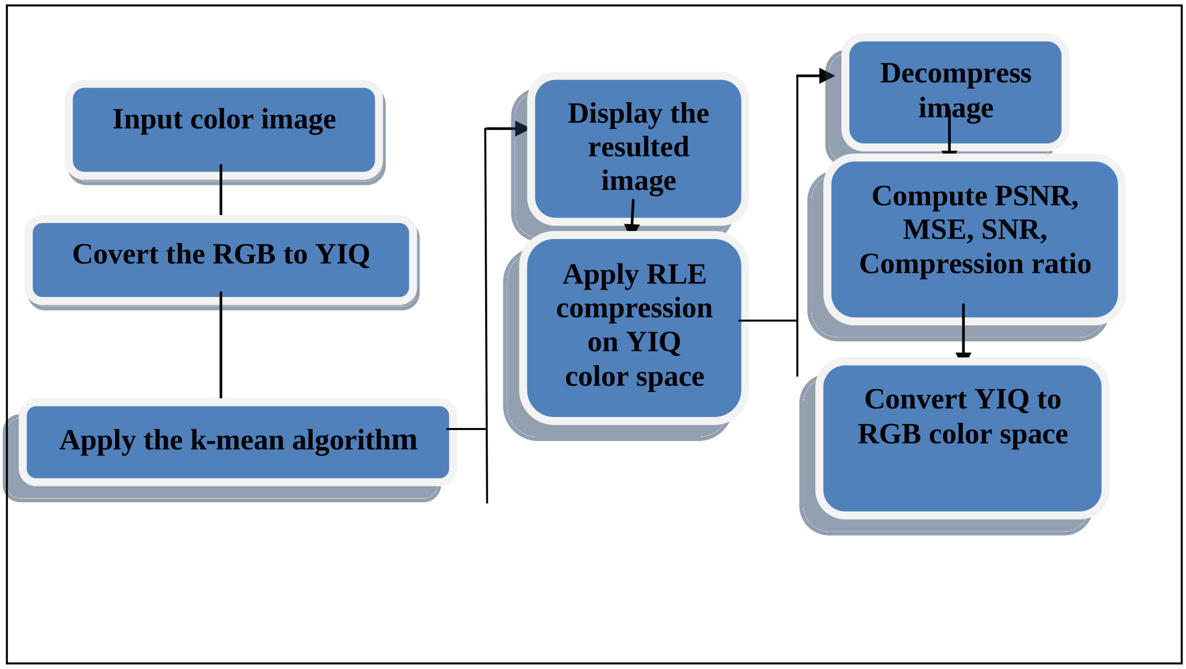

Figure 3: the structure of the suggested system

5. K-mean algorithm

Given the data set X, choose the number of clusters 1 < c < N. Initialize with random cluster centers chosen from the data set.

Repeat for l = 1; 2; …

Step 1: Compute the distances

$$D_{ik}^{2} = (x_k - v_i)^T (x_k - v_i), 1 \le i \prec c, 1 \le k \le N \qquad (4)$$

Step 2: Select the points for a cluster with the minimal distances, they belong to that cluster.

Step 3: Calculate cluster centers

$$v_i^l = \frac{\sum_{j=1}^{N_l} x_i}{N_i} \qquad (5)$$

Until $\Pi_{k=1}^{n} \max \left| v^{(l)} - v^{(l-1)} \right| \neq 0$

Ending Calculate the partition matrix

Now applying the RLE algorithm to each component Y, U, and V. explains this algorithm by used test numeric text.

Example:100,20,20,20,20,30,40,255,40,40,40,40,40,40,10,10,10

Compressed text: 100, # 20 4, 30, 40, 255, # 40 6,10,10,10

6. The Decompression Stage

1- Retrieve the parameters of original image height of the image, number of the clusters, and value of each cluster.

2- For each components Y, I, Q

* Read compressed files with RLE Algorithm then apply decompressed algorithm to produce Y, I, and Q components.

7. Quality measurements

A natural way to determine the quality of the recovered image is to find the difference between the original and reconstructed values. The most often used average measure is the average of squared error. This measure is called the Root mean squared error (RMSE) and it often given by:

$$e_{RMS} = \sqrt{\frac{1}{N^2}\sum_{r=0}^{N-1}\sum_{c=0}^{N-1}[I^{\wedge}(r,c) - I(r,c)]^2} \qquad (6)$$

Where:

$I^{\wedge}(r,c)$: Reconstructed Image.

$I\ (r,c)$: Original Image.

N: Samples number.

The smaller value of the error metrics, the better the compressed image represents the original image. Alternately, with the signal-to-noise (SNR) metrics, a larger number implies a better image. The SNR metrics consider the decompressed image I(r, c) to the "signal" and the error to be "noise". We can define the root-mean-square signal-to-noise ratio as:

$$SNR_{RMS} = \sqrt{\frac{\sum_{r=0}^{N-1}\sum_{c=0}^{N-1}[I^{\wedge}(r,c)]^2}{\sum_{r=0}^{N-1}\sum_{c=0}^{N-1}[I^{\wedge}(r,c) - I(r,c)]^2}} \qquad (7)$$

Another related metric, the peak signal-to nose ratio, is defined as:

$$PSNR(db) = 10\log_{10}\frac{(L-1)^2}{\frac{1}{N}\sum_{r=0}^{N-1}\sum_{c=0}^{N-1}[I^{\wedge}(r,c) - I(r,c)]^2} \qquad (8)$$

Where L= the number of gray levels (e.g., for 8 bits L=256)

PSNR is the most commonly used value to evaluate the objective image compression quality [14].

8. Conversion YIQ color space to RGB color space

After the process of decompress the result image, Convert the YIQ color space to RGB color space by applying the equation for the inverse converting process to display the result image.

9. The Results and Discussion

In this section the results of this work presented, where read each true color images entered to this system converted to the YIQ color space. This system applied to several color images. In figure (4) show the original true color images and figure (5) show

(a) (b) (c) (d)

Figure 4: shows the original color images

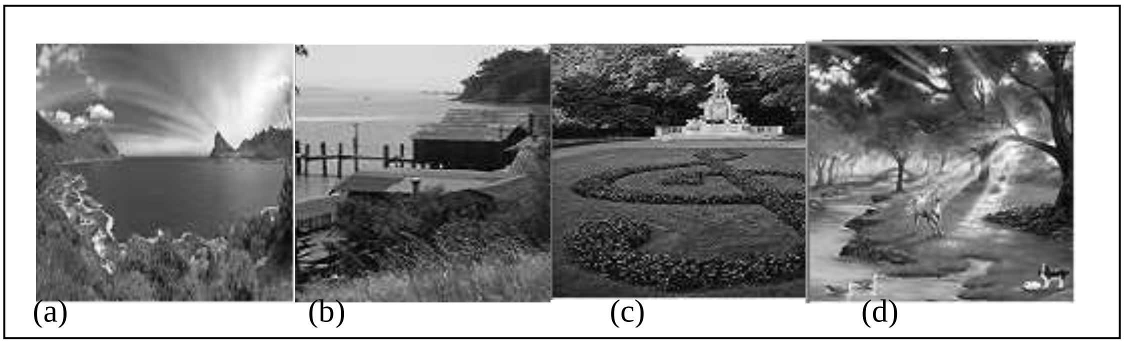

Figure 5: shows the Y Component for the previous images

the converting of RGB to YIQ color space and taken only Y component. Then k-mean algorithm applied with different cluster number, and at each implementation the compression ratio and quality measurements calculated.
The behavior of application k-mean algorithm using cluster number c=3, 4 for the different color images used in case study can be shown in figure 6.

Figure 6: shows the results of the K-mean for cluster number c=3, 4

Also the k-mean algorithm applying on different color images for cluster number C=8, 10, 12, 14 as shown in figure 7.

Figure 7::shows the results of the k-mean for cluster number c=8, 10, 12, 14

In the compression stage of the proposed system the RLE algorithm applied on the Y, I, Q components then decompression process applied and RGB color space for each image calculated. Tables (1, 2, 3, 4) contain the size of each components after applying RLE algorithm on these components, the compression ratio calculated to show the goodness of this algorithm in compression process, mean square error between the original and the compression color images, also the peak to signal ratio and signal to noise ratio computed to show the amount of the noise in the resulted images.

Table 1: show the size of Y, I, and Q after applying the RLE and PSNR, compression ratio, SNR, Compression Ratio for natural view image

	Number of cluster centers	Size of YUV components after RLE			RMS	Compression Ratio	SNR	PSNR
		Y	U	V				
	4	8138	16384	16384	1.19127159383089E-05	90.39%	97.37	12.34
	6	9995	16384	16384	1.02600017470098E-05	88.89%	98.02	14.11
	7	10055	16384	16384	1.28003735067712E-05	95.04%	97.06	18.26

Table 2: show the size of Y, I, and Q after applying the RLE and PSNR, compression ratio, SNR, Compression Ratio for beach image

	Number of cluster	Size of YUV components after RLE			PSNR	Compression Ratio	SNR	MSE
		Y	U	V				
	2	4015	14163	14163	3.6	78%	94.2	2.43483315067204E-05
	4	7181	14163	13163	6.1655	84.6%	95.96	1.64822461603538E-05
	8	9299	14163	14163	29.12	92.026%	98.87	8.42982533940118E-06

Table 3: show the size of Y, I, and Q after applying the RLE and PSNR, compression ratio, SNR, Compression Ratio for garden image

	Numberr	Size of YUV components after RLE			PSNR	Compression Ratio	SNR	MSE
		Y	U	V				
	4	7455	13448	13448	1.667	72.4%	92.84	3.37681622537387E-05
	8	11293	13448	13448	14.111	95.61%	96.76	1.37122958948283E-05
	12	11925	13448	13448	31.37	97.98%	98.006	1.02927018067209E-05

Table 4: show the size of Y, I, and Q after applying the RLE and PSNR, compression ratio, SNR, Compression Ratio for sea image

	Cluster Number	Size of YUV components after RLE			PSNR	Compression Ratio	SNR	MSE
		Y	U	V				
	2	703	16170	16170	3.55	73.13%	95.50	1.83450336377026E-05
	3	6424	16170	16170	9.33	86.46%	96.91	1.32364333174222E-05
	5	8102	16170	16170	13.45	89.39%	98.44	9.31001119966742E-06

10. Conclusions

- The application of k-mean algorithm provided a large amount of resolution and compression ratio.
- There is relation between cluster number and compression ratio.
- In RLE algorithm, applying it on the Y, I, Q components to reduce size them obtain on the size of Y reducing very largely, because the Y contains of the similar pixels depending on the number of clusters.
- The higher values of SNR in this work reflect the appearing and less amount of distortion in the resulted images.
- The using of YIQ from RGB color space reduce large amount of processes, because concentrating on one component instead of three components in the compression process as shown in the tables (1),(2),(3) and (4).
- The size of the Component Y related descending with the cluster number.

11. References

[1] S.Dutta, A.Abhinav, P.Dutta, P. Kumar, and A. Halder, “An Efficient Image Compression Algorithm Based on Histogram Based Block Optimization and Arithmetic Coding”, International Journal of Computer Theory and Engineering, Vol. 4, No. 6, December 2012.

[2] R. He, "Indexing Compressed Text”, A thesis, Waterloo, Ontario, Canada, 2003.

[3] R. C. Gonzalez and others, “Digital Image Processing", University of Tennessee, 1992.

[4]S. W. Smith, A sample chapter from: "The Scientist and Engineer's Guide to Digital Signal Processing”, 1997.

[5] V.B. Nikam, V.J. Kadam, B. B. Meshram ,“Image Compression Using Partitioning Around Melodies Clustering Algorithm". IJCS, Vol. 8, Issue 6, No 1, November 2011.

[6] Sherin M. Youssef , Mohamed Rizk , Mohamed El-Sherif, “Enhanced swarm-like agents for dynamically adaptive data clustering”, Proceedings of the 2nd WSEAS International Conference on Computer Engineering and Applications, p.213-219, January 25-27, Acapulco, Mexico International Journal of Database Management Systems (IJDMS) Vol.3, No.4, November 2011.

[7] Marcel Brun , Chao Sima , Jianping Hua , James Lowey , Brent Carroll , Edward Suh , Edward R. Dougherty, Model-based evaluation of clustering validation measures, Pattern Recognition, v.40 n.3, March p.807-824, 2007 .

[8] Nargess Memarsadeghi, Dianne P. O'Leary, “Classified Information: The Data Clustering Problem”, Computing in Science and Engineering, v.5 n.5, September p.54-60, 2003.

[9] Yifan Li , Jiawei Han , Jiong Yang, “Clustering moving objects”, Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, August 22-25, Seattle, WA, USA,2004.

[10] Plataniotis K. and Venetsanopoulos, “Color Image Processing and Application”, Springer, 2000.

[11] Rafael C. Gonzalez, Richard E. Woods, “ Digital Image Processing “,Addison-Wesley publishing company, Inc., 1992.

[12] Nick Efford,”Digital Image Processing”: A practical Introduction Using Java, Person Education. USA, 2000.

[13] Panrony X., “Image Compression by Wavelet Transform”, East Tennessee State University, Master Thesis, 2001.

[14] Scott V.,"Computer Vision and Image Processing a Practical Using CVIP tools”, Prentice Hall, 1998.