International Journal of Engineering Science Invention (IJESI)
ISSN (Online): 2319 – 6734, ISSN (Print): 2319 – 6726
www.ijesi.org || PP. 65-69

An Improved Fast Color Halftone Image Data Compression Algorithm

R.Jothi.,[1] Dr.C.Bhuvaneswari[2]
[1]*(Assistant Professor and head, Department of Computer Application, Krishnasamy College of Science, Arts & Management for Women, Cuddalore-9, TamilNadu, India)*
[2]*(Assistant Professor and head, Department of computer science, Thiruvalluvar university college of arts and science, Thiruvennainallur, TamilNadu, India)*

Abstract: *Halftoning is a technique that converts tone image into halftone image that is image with 8 bit plane into 1 bit plane. Halftoning is widely employed in the printing and display of digital images. The need for halftoning encoding arises either because the physical processes involved are binary in nature or the processes have been restricted to binary operation for reasons of cost, speed, memory or stability in the presence of process fluctuations. Here the field is presented, starting with analog methods of halftone image rendering and proceeding to the digital techniques of template dot halftones, noise encoding, ordered dither, and error diffusion. Here, comparisons of image data halftone and vector quantization techniques are used. To achieve higher compression ratio combination of halftone with improved Kekre's Fast codebook generation vector quantization algorithm is used. Improved KFCG algorithm contains Line pattern extraction with least square fitting methods. It is used to reduce time and memory space to achieve compression and decompression of image data. In this system the analysis of pixel average intensity of each plane and mean square error is calculated.*
Keywords: *Code book, Inverse, Plane intensity, Vector.*

I. Introduction

The main objective is to reduce the encoding time in halftone image compression by reducing the computational complexities. The proposed image compression based on an Kekre's Fast Codebook Generation algorithm and it has two advantages: it is used to drastically reduces time and computational complexity. Here Finite Impulse Response and Fast Inverse Half Toning algorithms are used. The system analysis the pixel average intensity of each plane and mean square error is calculated. It contains the line pattern extraction with least square fitting methods which gives lossless compression and reduced time computation.
Modern digital technology has made it possible to manipulate multi-dimensional signals with systems that range from simple digital circuits to advanced parallel computers. The goal of this manipulation can be divided into three categories:
* Image Processing *image in -> image out*
* Image Analysis *image in -> measurements out*
* Image Understanding *image in -> high-level description out*

Further, we will restrict ourselves to two-dimensional (2D) image processing although most of the concepts and techniques that are to be described can be extended easily to three or more dimensions. An image defined in the "real world" is considered to be a function of two real variables, for example, $a(x, y)$ with a as the amplitude (e.g. brightness) of the image at the *real* coordinate position (x, y). An image may be considered to contain sub-images sometimes referred to as *regions-of-interest*, *ROIs*, or simply *regions*. This concept reflects the fact that images frequently contain collections of objects each of which can be the basis for a region. In a sophisticated image processing system it should be possible to apply specific image processing operations to selected regions. Thus one part of an image (region) might be processed to suppress motion blur while another part might be processed to improve color rendition.

The amplitudes of a given image will almost always be either real numbers or integer numbers. The latter is usually a result of a quantization process that converts a continuous range, between 0 and 100% to a discrete number of levels. In certain image-forming processes, however, the signal may involve photon counting which implies that the amplitude would be inherently quantized. In other image forming procedures, such as magnetic resonance imaging, the direct physical measurement yields a complex number in the form of a real magnitude and a real phase.

II. Working Principle

2.1 Digital Image

A digital image $a[m, n]$ described in a 2D discrete space is derived from an analog image $a(x, y)$ in a 2D continuous space through a *sampling* process that is frequently referred to as digitization. For now we will look at some basic definitions associated with the digital image. The effect of digitization is shown in Figure 2.1. The 2D continuous image $a(x, y)$ is divided into *N rows* and *M columns*. The intersection of a row and a column is termed a *pixel*. The value assigned to the integer coordinates $[m, n]$ with $\{m=0,1,2,...,M-1\}$ and $\{n=0,1,2,...,N-1\}$ is $a[m, n]$. In fact, in most cases $a(x, y)$--which we might consider to be the physical signal that impinges on the face of a 2D sensor--is actually a function of many variables including depth (z), color (λ) and time (t).

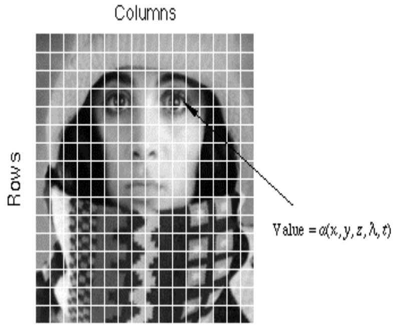

Fig 2.1: Digitization of a continuous image. The pixel at coordinates $[m=10, n=3]$ has the integer brightness value 110.

The image shown in Figure 2.1 has been divided into $N = 16$ rows and $M = 16$ columns. The value assigned to every pixel is the average brightness in the pixel rounded to the nearest integer value. The process of representing the amplitude of the 2D signal at a given coordinate as an integer value with L different gray levels is usually referred to as amplitude quantization or simply *quantization*.

2.2 Image Enhancement

It can be defined as conversion of the image quality to a better and more understandable level for feature extraction or image interpretation, while radiometric correction is to reconstruct the physically calibrated value from the observed data. On the other hand, feature extraction can be defined as the operation to quantify the image quality through various parameters or functions, which are applied to the original image. These processes can be considered as conversion of the image data. Image enhancement is applied mainly for image interpretation in the form of an image output, while feature extraction is normally used for automated classification or analysis in a quantitative form. Typical image enhancement techniques include gray scale conversion, histogram conversion, color composition, color conversion between RGB and HSI, etc., which are usually applied to the image output for image interpretation.
Feature Extraction features involved in an image are classified as follows.
(1) Spectral features - special color or tone, gradient, spectral parameter etc.
(2) Geometric features - edge, lineament, shape, size, etc.
(3) Textural features - pattern, spatial frequency, homogeneity, etc.

2.3 Gray Scale Conversions and Linear Conversion

Gray scale conversion is one of the simplest image enhancement techniques. Gray scale conversion can be performed using the following function.
y = f (x), where x: original input data
y: converted output data
Linear conversion
y = ax + b
a : gain , b : offset
Statistical procedures can be also applied in two ways as follows.
(1) Conversion of average and standard deviation

$$y = \frac{S_y}{S_x}(x - x_m) + y_m$$

where xm : average of input image, Sx : standard deviation of input image, ym : average of output image, Sy : standard deviation of output image

(2)Regression: In such cases as multi-date images for producing a mosaic or radiometric adjustment, a selected image can be related to other images using regression technique. Using the regression technique between different detectors can eliminate line noise due to different detectors, for example Landsat MSS.

2.4 Histogram Conversion

Histogram conversion is the conversion of the histogram of original image to another histogram. Histogram conversion can be said to be a type of gray scale conversion. There are two typical histogram conversion techniques.

2.4.1. Histogram equalization

Histogram equalization is to convert the histogram of an original image to equalized histogram. As a first step, an accumulated histogram should be made. Then the accumulated histogram should be divided into a number of equal regions. Thirdly, the corresponding gray scale in each region should be assigned to a converted gray scale. The effect of histogram equalization is that parts of the image with more frequency variation will be more enhanced, while parts of an image with less frequency will be neglected.

2.4.2. Histogram normalization

Generally a normal distribution of the density in an image would create an image that is natural for a human observation. In this sense the histogram of the original image may be sometimes converted to the normalized histogram. However in this conversion, pixels with same gray scale should be reallocated to other pixels with different gray scales, in order to form a normalized histogram. Histogram normalization may be applied, for example, to an unfocused image of a planet with a low dynamic range, though it is not be very much popular for ordinary remote sensing data.

2.5Arithmetic Operations

Arithmetic operations can be thought of as manipulation of pixel values using additions, subtractions, multiplications, and divisions. They are also called linear point operations, since the output image value is a linear function of the input image.

$D_{OUT} = aD_{IN} + b$ where D_{OUT} is the output image value and D_{IN} is the input pixel value.

Adding or subtracting a constant to and from a pixel value changes the brightness of an image. That is, the output image appears brighter when b > 0, and the image appears darker when b < 0. Figure 2.1 shows the effect of adding 50 to and subtracting 50 from every pixel value of the image BIRDGIRL. Multiplying or dividing the pixel values by a constant has the effect of increasing or decreasing the image's contrast. Figure 2.2 shows the effect of multiplying pixel values by 1.5 and 0.8. Arithmetic operations do create problems when the resulting pixel values are above 255 or below 0. Thus, clamping is usually used to set negative values to 0 and values above 255 to 255. The arithmetic operation function is implemented using table-lookup. Table 2.1 shows the operation of a "$D_{OUT} = 3*D_{IN}$" arithmetic operation using table-lookup. Note that the pixel value 255 is also converted to 255 because of clamping.

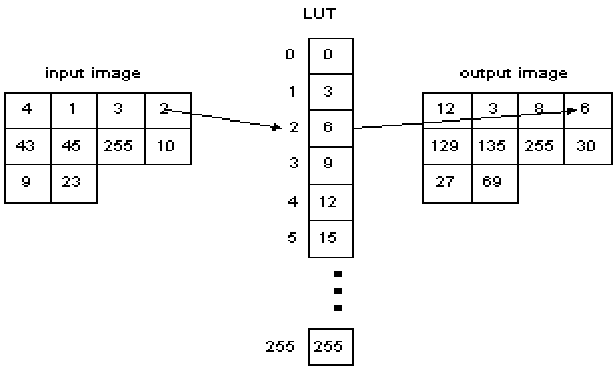

Table 2.1 An example of arithmetic operation using table-lookup

III. Digital Halftoning

Halftoning or analog halftoning is a process that simulates shades of gray by varying the size of tiny black dots arranged in a regular pattern. This technique is used in printers, as well as the publishing industry. If you inspect a photograph in a newspaper, you will notice that the picture is composed of black dots even though it appears to be composed of grays. This is possible because of the spatial integration performed by our eyes. Our eyes blend fine details and record the overall intensity [1]. Digital halftoning is similar to halftoning in which an image is decomposed into a grid of halftone cells. Elements or dots that halftoning uses in simulates shades of grays of an image are simulated by filling the appropriate halftone cells. The number of black dots in a halftone cell, the darker the cell appears. For example, in Figure 3.1, a tiny dot located at the center is simulated in digital halftoning by filling the center halftone cell; likewise, filling the four cells at the top-left corner simulates a medium size dot located at the top-left corner. Filling all halftone cells simulates the large dot covering most of the area in the third image.

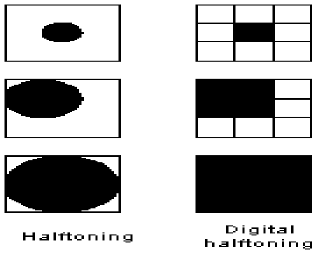

Fig 3.1 A sample of digital halftoning

Three common methods for generating digital halftoning images are patterning , dithering and error diffusion .

3.1 Halftone Compression

Half Tone is an application of image processing widely used in printing process. With the evolution of latest printing concepts the field of halftoning evolved in digital halftoning. All newspapers, magazines and books are printed with digital halftoning. The grayscale digital image consists of 256 gray levels but normal printers have only colored ink. Patterning and dithering methods are used and reproducing the image with high quality. The images can be compressed using two methods. It is similar to the context based arithmetic coding algorithm which adaptively positions the template pixels in order to obtain correlations between the adjacent pixels. Second is descreening is performed on the halftone image so that the image is converted back to gray scale.

3.2 Error Diffusion

Error diffusion is another technique used for generating digital half toned images. It is often called spatial dithering. Error diffusion sequentially traverses each pixel of the source image. Each pixel is compared to a threshold. If the pixel value is higher than the threshold, a 255 is outputted; otherwise, a 0 is outputted. The error - the difference between the input pixel value and the output value - is dispersed to nearby neighbors. Error diffusion is a neighborhood operation since it operates not only on the input pixel, but also its neighbors. Generally, neighborhood operations produce higher quality results than point operations. Error diffusion, when compared to dithering, does not generate those artifacts introduced by fix thresholding matrices. However, since error diffusion requires neighborhood operations, it is very computationally intensive. Error diffusion generates a digital halftone image using error diffusion. It reads an input image, compares each pixel with the input threshold, and sets the output to 0 or 255. The quantization error is then computed and dispersed to input pixels to the right and below the current pixel with different weights. The weights used in this implementation were first specified by Floyd and Steinberg (1975). A word of caution: since "error diffusion" requires excessively intensive computations, images of size less than 70x70 are recommended. The value that determines the output to be 0 or 255. It is usually set to 128.

IV. Vector Quantization

Vector Quantization plays an important role to generate a codebook such that distortion between the original image and the reconstructed image is the minimum. An image is 2-D analog signal which is converted to digital form for processing, so the need for image compression arises for resourceful storage and transmission. Compression is achieved by the removal of one or more of the basic data redundancies such as coding inter pixel and psycho visual redundancies. VQ is most popularly used in lossy data compression. The codebook is generated for each image which is representation of entire image which contains a definite pixel pattern calculated X subset of R_k by set of reference vectors CB ={C1,C2…C_N} in R_k. CB is a codebook which has a set of reproduced code words. During encoding form the original image is divided into several K dimensions vectors and each vector is encoded by index of codeword by a table look-up method. When decoding the receivers uses the same codebook to translate the index back to its corresponding codeword.

KFCG algorithm requires less number of computations, reduces the processing time. The halftone image is treated as input to KFCG for further image data compression. Divide the full image into NxN into 2x2 non-overlapping blocks. From 12 byte input vector as a row in matrix so as to get V1, V2, ….,Vm number of input vectors. This matrix is called as training set or cluster1. Centroid or code vector is calculated, then comparing first byte of Centroid with each co-efficient of input vector and dividing cluster1 as two clusters one and two. Codebook is generated repeatedly. The total number of code vectors generated is 2n, where n is the number of bits used as code vector index. Each pixel of line pattern is accumulated on vector called cumulative matrix. The cumulative matrix is employed to describe the line characteristic of the image. Image is decoded from index by taking corresponding code vector from codebook.

V. Conclusion

Digital halftoning is the process of rendering an image with greater amplitude resolution to one with lesser amplitude resolution. This has been practiced for over one hundred years in the printing industry: indeed a printer can only produce black dots on a sheet of paper, and one has to optimally distribute them across the space to render a visually pleasing pattern, giving the illusion of the original tonal quality of the image. This is not the only application of halftoning, though. Image rendering with a significant reduction of physically distinct colours is also a problem that arises for low level displays. Compression techniques reduce the amount of data needed to represent an image so that images can be economically transmitted and stored. A VGA screen is only driven by two bits for each colour gun (RGB), which gives 64 different colours, instead of the 2^{24} possibilities in a bitmap image. This is an immediate generalization of our work, known as multilevel dithering. Halftoning is a method for creating the illusion of continuous tone output with a binary device. Effective digital halftoning can substantially improve the quality of rendered images at minimal cost. Thus vector quantization technique KFCG is a lossy technique which improves the color tone and line pattern extraction method. In future other types of method can be used to reduce error calculation instead of binary sequence.

References

Journal Papers:

[1]. N. Damera-Venkata and B.L. Evans, "Design and analysis of vector color error diffusion halftoning systems," *IEEE Trans. Image Processing*, vol. 10, pg. 1552-1565, Oct. 2001.

[2]. S. Ardalan and J. Paulos, "An analysis of nonlinear behavior in delta-sigma modulators," *IEEE Trans. Circuits Syst.*, vol. CAS-34, pg. 593-603, June 1987.

[3]. T.D. Kite, B.L. Evans, and A.C. Bovik, "Modeling and quality assessment of halftoning by error diffusion," *IEEE Trans. Image Processing*, vol. 9. pg. 909-922, May 2000.

[4]. B. Kolpatzik and C. Bouman, "Optimized error diffusion for high quality image display," *J. Electron. Imaging*, vol. 1, pg. 277-292, Jan. 1992.

[5]. L. Akarun, Y. Yardimici, and A.E. Cetin, "Adaptive methods for dithering color images," *IEEE Trans. Image Processing*, vol. 6, pg. 950-955, July 1997.

[6]. C. Sharma and H.J. Trussell, "Digital color imaging," *IEEE Trans. Image Processing*, vol. 6, pg. 901-932, July 1997.

Books:

[7]. H. Staik and J.W. Woods, *Probability, Random Processes and Estimation Theory for Engineers*(Englewood Cliffs, NJ: Prentice-Hall, 1986)

[8]. T.K. Moon and W.C. Stirling, *Mathematical Methods and Algorithms for Signal Processing*(Englewood Cliffs, NJ: Prentice-Hall, 2000)

[9]. R. Ulichney, *Digital Halftoning*(Cambridge, MA: MIT Press, 1987)

Proceedings Papers:

[10]. T.J. Flohr. B.W. Kolpatzik, R. Balasubramanian, D.A. Carrara, C.A. Bouman, and J.P. Allebach, "Model based color image quantization," in *Proc. SPIE Human Vision, Visual Proc. and Digital Display IV*, 1993, vol. 1913, pg. 270-281

[11]. H. Haneishi, T. Suzuki, N. Shimonyama, and Y. Miyake, "Color digital halftoning taking colorimetric color reproduction into account," *J. Electron.Imaging*, vol. 5, pg. 97-106, Jan. 1996.

[12]. Z. Fan and S. Barrington, "Improved quantization methods in color error diffusion," *J. Electron. Imaging*, vol. 8, pg. 430-437, Oct. 1999.

[13]. C. Kim, I. Kweon, and Y. Seo, "Color and printer models for color halftoning," *J. Electron. Imaging*, vol. 6, pg. 166-180, June 1997.

[14]. R. Klassen, R. Eschbach, and K. Bharat, "Vector error diffusion in a distorted colour space," in *Proc. IS&T Conf.*, 1994 [Online]. Available: http://www.imaging.org

[15]. D.L. Lau, G.R. Arce, and N.C. Gallagher, "Digital color halftoning with generalized error diffusion and multichannel green-noise masks," *IEEE Trans. Image Processing*, vol. 9, pg. 923-935. May 2000.

[16]. D. Shaked, N. Arad, A. Fitzhugh, and I. Sobel, "Color diffusion: Error-diffusion for color halftones," HP Labs, Haifa, Israel, Tech. Rep. HPL-96-128R1, 1996.

[17]. R. Eschbach. "Pseudo-vector error diffusion using cross separation threshold imprints," in *Proc. IS&T Symp. Electronic Imaging*, 1999, pg. 321