Review of Computer Engineering Research
2016 Vol.3, No.2, pp.35-40
ISSN(e): 2410-9142
ISSN(p): 2412-4281
DOI: 10.18488/journal.76/2016.3.2/76.2.35.40
© 2016 Conscientia Beam. All Rights Reserved.

LOSSLESS IMAGE COMPRESSION AND DECOMPRESSION TO IMPROVE THE PSNR AND MSE VALUES USING ARCHITECTURE

S.Kannadhasan[1]† --- B.Naveen Lingaesh[2] --- R.Alagumanikandan[3]
[1]*Assistant Professor, Department of Electronics and Communication Engineering, Raja College of Engineering and Technology, Madurai, Tamilnadu, India*
[2,3]*U.G Student, Department of Electronics and Communication Engineering, Raja College of Engineering and Technology, Madurai, Tamilnadu, India*

ABSTRACT
An adaptive algorithm for compressing the color images is proposed. This technique uses a combination of simple and computationally easy operations. The two main steps consist of decomposition of data and data compression. The result is a practical scheme that achieves good compression while providing fast decompression. The approach has performance comparable to and often better than, existing architecture. This paper gives the overview of an adaptive lossless compression scheme. This scheme uses a new technique to predict a pixel by matching neighboring pixel, an adaptive color difference estimation scheme to remove the color spectral redundancy while handling red and blue samples and an adaptive codeword generation technique to encode the prediction residues. The technique lossless image compression plays an important role in image transmission and storage for high quality. At present, both the compression ratio and processing speed should be considered in a real time multimedia system. Lossless compression algorithm is used for this technique. A low Complexity predictive model is proposed using the correlation of pixels and color components. Also a color space transform is used and good decoration is obtained in our algorithm. The compared experimental results have shown that our algorithm has a noticeably better performance than traditional algorithms.

Keywords: Preprocessor, DDPCM, Golombrice encoder, Decoder.

Received: 8 September 2016/ **Revised:** 20 September 2016/ **Accepted:** 4 October 2016/ **Published:** 19 October 2016

1. INTRODUCTION

In real time image processing on massive multimedia, jobs, high performance communication between processor and memory is required. In addition, increasing performance gaps between embedded processors and external memories have resulted in greater bandwidth requirement in the communication networks or data buses. There still exists a critical limitation on the physical size of a bus due to restricted operating speed from signal integrity issues and the increased hardware area. Compressed memory architecture can be good alternatives to the above problems since it can significantly reduce requirements for increasing bus bandwidth. Digital Image Processing is used for convert video pictures into a digital form, and then applies various algorithms to extract the information from the picture . It is also used for the computerized technique for image analysis. Image acquisition technique is the process of capturing a scene and digitizing it into a pixel image.It has been widely used in several disciplines such as medicine, biology, material science and in industry, astronomic monitoring, medical diagnosis, product quality control, weather forecasting, etc..,.

† Corresponding author
© 2016 Conscientia Beam. All Rights Reserved.

2. RELATED WORK

High performance lossless color image compression and decompression architecture to reduce both memory requirement and bandwidth is proposed [1]. The Proposed architecture consists of differential-differential pulse coded modulation (DDPCM) and Golomb Rice coding. The original frame is organized as m by n sub window arrays, to which DDPCM is applied to produce one seed and m*n-1 pieces of differential data. Then the differential data are encoded using the Golomb-Rice algorithm to produce lossless compressed data [2]. According to the experimental results on benchmark images, the proposed architecture can guarantee a high enough compression rate and throughput to perform real time lossless CODEC operations with a reasonable hardware area.

3. PROPOSED SYSTEM

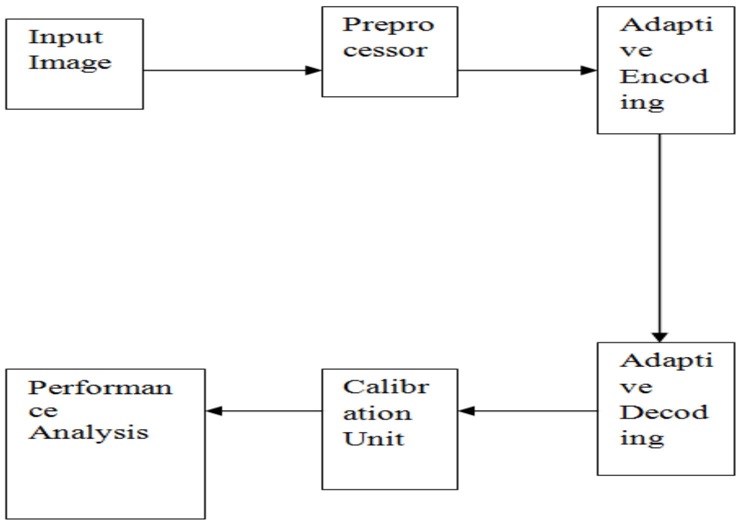


Figure-1. Block Diagram of Proposed Architecture

Source: Figure 1 shows the basic architecture model of digital image processing. This source has been taken from the digital image processing Gonzalez book.

Image Processing technique have concerned with manipulation and analysis of images by computer includes wide range of techniques for manipulating and modifying images in various ways.Image acquisition is used to acquire a digital image to improve the image in ways that increase the chances for success of the other processes. Image segmentation technique is used to partitions an input image into its constituent parts or objects. Image representation is used to convert the input data to a form suitable for computer processing. Image description technique is used to extract features that result in some quantitative information of interest or features that are basic for differentiating one class of objects from another. Image recognition technique is used to assign a label to an object based on the information provided by its descriptors. Image interpretation is used to assign meaning to an ensemble of recognized objects. Compression technique is used for reducing the storage required to save an image or transmit it. It also deals with tools for extracting image components that are useful in the representation and description of shape of the images. Image compression is the art and science of representing information in a compact form. Still image data, that is a collection of 2-D arrays (one for each color plane) of values representing intensity (color) of the point in corresponding spatial location (pixel).

3.1. Proposed Compression Architecture

The compression step is divided into two pipeline stages as shown in figure 1, in order to reduce the critical path delay. In the first stage, DDPCM-based preprocessing operations are performed, one piece of seed, 15 pieces

© 2016 Conscientia Beam. All Rights Reserved.

of differential data are latched and delivered to the second stage. In the second stage, 15 pieces of differential data are compressed by Golomb-Rice encoder and the encoded data are packed according to the segment data format. Using this information about whether the data are changing by large or small amounts in the vertical or horizontal direction in the neighborhood of the data being encoded provides a good initial prediction.

3.2. Preprocessor

It consists of two stages like sampling and DDPCM. Before processing, image by DDPCM image is sampled.

3.3. DDPCM

DDPCM is applied to 4*4 block data (the sub-window size is assumed to be 4*4 throughout this paper unless otherwise described) to produce one piece of seed and 15 pieces of differential data. In the DDPCM preprocessing step, 8-bit input data is processed by two successive subtractions so that the length of the DDPCM result becomes 10 bits.

3.4.Golomb-Rice Encoder

The Golomb rice encoder receives 15 pieces of 10-bit differential data from the DDPCM module and generates four outputs which are sign values (15 bits), 15 remainder data (15*2 bits), variable length unary-encoded quotient data (15-68 bits), and unary code length (7 bits). In the DDPCM preprocessing step, 8-bit input data is processed by two successive subtractions so that the length of the DDPCM result data becomes 10 bits, four outputs which are sign values (15 bits), 15 remainder data(15*2 bits), variable length unary-encoded quotient data (15-68 bits) and Unary code length (7 bits).

3.5. Data Packing

Once the Golomb-Rice encoding operation is finished, the final compression data are produced by packing the encoded codeword and information data according to the format as shown in figure 2.

SEED (8)	L (7)	SIGN (15)	REMAINDER (30)	UNARY (15-68)

Figure-2. Data Format for Proposed Architecture

Source: Figure 2 shows the basic architecture model of datapacking techique. This source has been taken from data compression Khalid Sayood book

The length of the encoded data can be greater than the length of the uncompressed data. A piece of seed data, a unary code length, the sign information of the DDPCM results, the binary coded remainders and the unary encoded quotient data are packed together. The remaining data (75-128 bit enclosed code when compressed and 128 bit original data when uncompressed) are stored in the memory.

3.6.Golomb Rice Decoder

The quotient unary code word should be decoded to reconstruct the original quotient data. Once 15 pieces of quotient data will be recovered by putting together remainders and quotients as shown in figure 3. Then the 15 pieces of unsigned differential data are converted into the original signed data accordingly to the information. Finally 8 bit seed and 15 pieces of 10 bit differential data construct the DDPCM data which will be processed to reconstruct the original 4*4 sub window image by the inverse DDPCM operation.

© 2016 Conscientia Beam. All Rights Reserved.

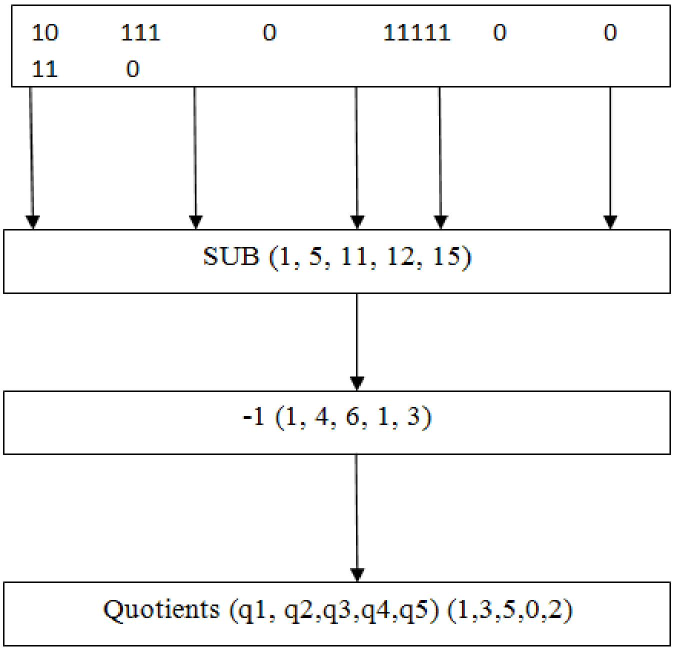


Figure-3. Remainder and Quotients

Source: Figure3 shows the Golomb Rice Decoder technique. This source has been taken from data compression Khalid Sayood book

4. RESULTS AND DISCUSSION

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - x_i)^2 \text{------------------------------}(1)$$

Equation 1 shows the Mean Square Error

$$SNR = 10 \cdot \log_{10}\left[\sigma^2 / MSE\right] \text{-----------------}(2)$$

Equation 2shows the Signal to Noise Ratio

Figure-4. Gray Scale Image

© 2016 Conscientia Beam. All Rights Reserved.

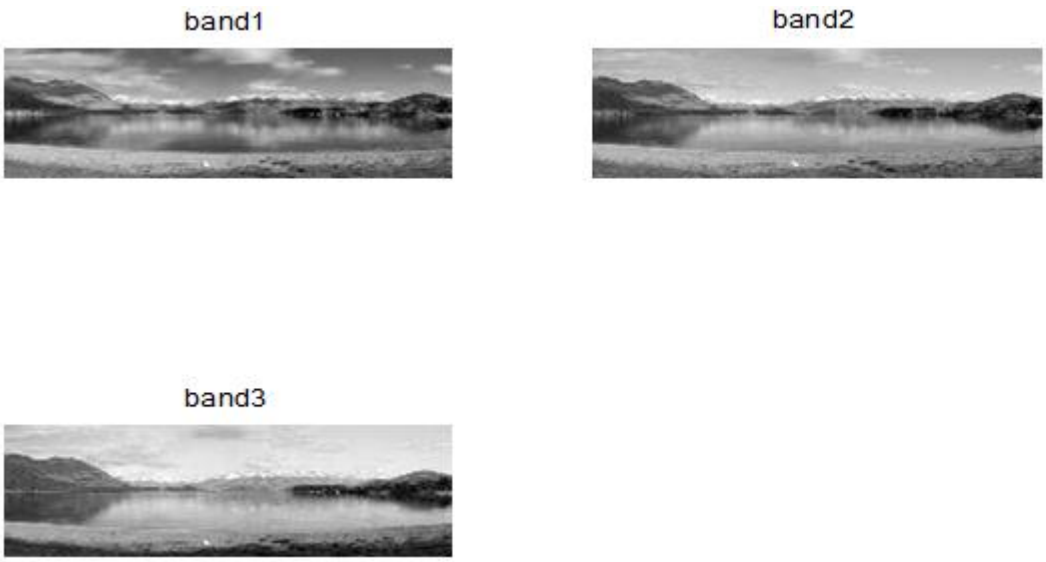
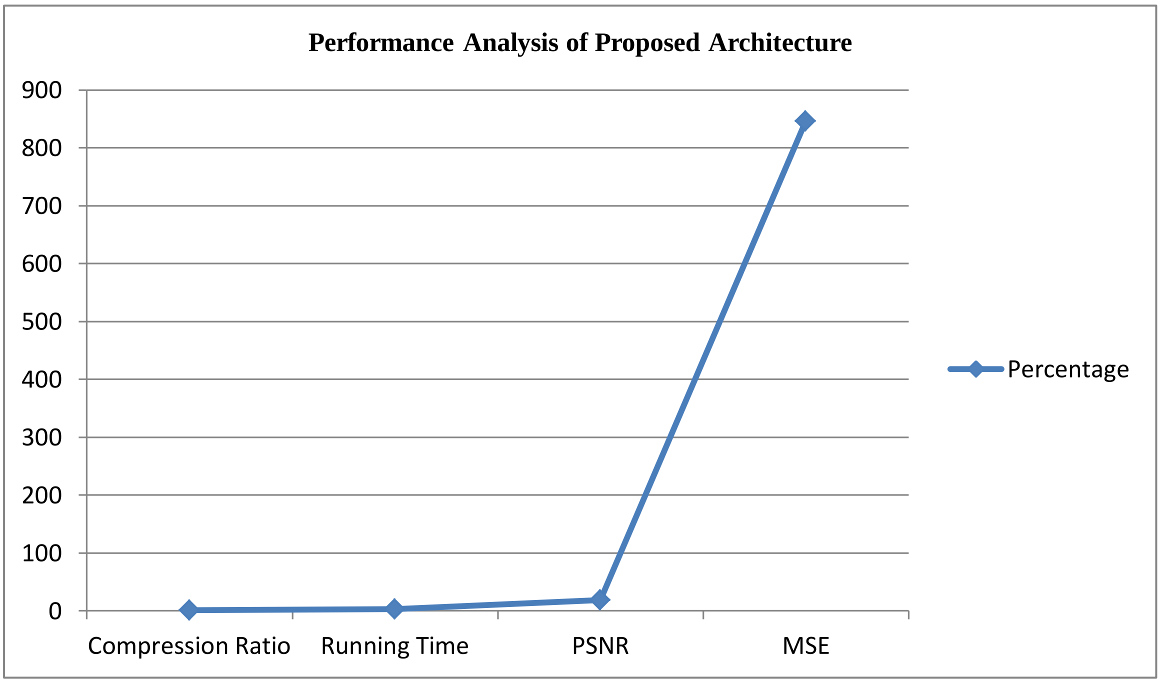


Figure-4. Architecture Image

Table-1. Performance Analysis of Proposed Architecture

Sl.No	Parameters	Percentage (%)
1	Compression Ratio	0.90
2	Running Time	2.84
3	PSNR	18.850
4	MSE	846.4138

Source: Table 1 shows the basic formula in digital processing techniques.



5. CONCLUSION

The unary decoder recovers the original 15 piece between 15 bits and 68 bits. Each piece of decoded quotient data has a fixed length of 8 bits. Since the unary code word contains 15 pieces of compressed 0s within the variable length unary code word. The proposed unary decoding hardware determines the position of termination 0s and

© 2016 Conscientia Beam. All Rights Reserved.

then reconstructs the original quotient data by subtracting the successive two position values. In order to improve the throughput of the whole system, all of 15 positions are calculated in parallel.

Funding: This study received no specific financial support.

Competing Interests: The authors declare that they have no competing interests.

Contributors/Acknowledgement: All authors contributed equally to the conception and design of the study.

REFERENCES

[1] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli, "Image denoising using a scale mixture of Gaussians in the wavelet domain," *IEEE Transactions on Image Processing*, vol. 12, pp. 1338–1351, 2003.

[2] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Transactions on Image Processing*, vol. 15, pp. 3736–3745, 2006.

BIBLIOGRAPHY

[1] A. Buades, B. Coll, and J. Morel, "A review of image denoising methods, with a new one," *Multiscale Model Simul.*, vol. 4, pp. 490–530, 2005.

[2] W. Dong, L. Zhang, G. Shi, and X. Wu, "Image deblurring and super resolution by adaptive sparse domain selection and adaptive regularization," *IEEE Trans. Image Process*, vol. 20, pp. 1838–1857, 2011.

[3] R. C. Gonzalez and E. W. Richard, "Digital image processing," 2008.

[4] B. Chanda and D. M. Dwijesh, *Digital image processing and analysis*: PHI Learning Pvt. Ltd, 2004.

[5] H. Zhengting, "Video compression and data flow for video surveillance," *Texas Instruments*, *SPRY104*, 2007.

Views and opinions expressed in this article are the views and opinions of the author(s), Review of Computer Engineering Research shall not be responsible or answerable for any loss, damage or liability etc. caused in relation to/arising out of the use of the content.

© 2016 Conscientia Beam. All Rights Reserved.