A Novel Algorithm of Super-Spatial Structure Prediction for RGB Colourspace

C.S.Rawat, Seema G Bhateja, Dr. Sukadev Meher

Abstract— In image compression the key challenge is to efficiently encode and represent high frequency image structural components such as patterns, edges and textures. In this work, we develop an efficient image compression scheme based on super-spatial prediction of structural units. This so-called similar structure block prediction is motivated by motion prediction in video coding, attempting to find an optimal prediction of structure components within previously encoded image regions.

Index Terms— Bit Rate , Compressed Image, Context-based adaptive lossless image coding (CALIC), lossless image compression, RGB Colorspace, structure components, super spatial structure prediction.

1 INTRODUCTION

THE key in efficient image compression is to explore source correlation so as to find a compact representation of image data. Existing lossless image compression [1], [2] schemes attempt to predict image data using their spatial neighborhood [1]. A natural image often contains a large number of structure components, such as edges, contours, and textures. These structure components may repeat themselves at various locations and scales. Therefore, there is a need to develop a more efficient image prediction scheme to exploit this type of image correlation.

The idea of improving image prediction and coding efficiency by relaxing the neighborhood constraint can be traced back to sequential data compression [4] and vector quantization for image compression. In sequential data compression, a substring of text is represented by a displacement/length reference to a substring previously seen in the text. Storer extended the sequential data compression to lossless image compression. However, the algorithm is not competitive with the state-of-the-art such as context-based adaptive lossless image coding (CALIC)[1] in terms of coding efficiency. During vector quantization (VQ) for lossless image compression, the input image is processed as vectors of image pixels. The encoder takes in a vector and finds the best match from its stored codebook. The address of the best match, the residual between the original vector and its best match are then transmitted to the decoder. The decoder uses the address to access an identical codebook, and obtains the reconstructed vector. Recently, researchers have extended the VQ method to visual pattern image coding (VPIC) and visual pattern vector quantization (VPVQ). The encoding performance of VQ-based methods largely depends on the codebook design. These methods still suffer from lower coding efficiency, when compared with the state-of-the-art image coding schemes.

In the intra prediction scheme proposed by Nokia, there are ten possible prediction methods: DC prediction, directional extrapolations, and block matching. DC and directional prediction methods are very similar with those of H.264 intra prediction [3]. The block matching tries to find the best match of the current block by searching within a certain range of its neighboring blocks. This neighborhood constraint will limit the image compression efficiency since image structure components may repeat themselves at various locations.

In fractal image compression [4], the self-similarity between different parts of an image is used for image compression based on contractive mapping fixed point theorem. However, the fractal image compression focuses on contractive transform design, which makes it usually not suitable for lossless image compression. Moreover, it is extremely computationally expensive due to the search of optimum transformations. Even with high complexity, most fractal-based schemes are not competitive with the current state of the art [1].

An efficient image compression scheme based on super-spatial structure prediction of structure units is presented here. A natural image can be often separated into two types of image regions: structure and non-structure regions. Nonstructure regions, such as smooth image areas, can be efficiently represented with conventional spatial transforms, such as KLT (Karhunen Lòeve transform), DCT (discrete cosine transform) and DWT (discrete wavelet transform). However, structure regions, which consist of high frequency structural components and curvilinear features in images, such as edges, contours, and texture regions, cannot be efficiently represented by these linear spatial transforms. They are often hard to compress and consume a majority of the total encoding bit rate.

Super-spatial structure prediction breaks the neighborhood constraint, attempting to find an optimal prediction of structure components [5], [6] within the previously encoded image regions. It borrows the idea of motion prediction from video coding, which predicts a block in the current frame using its previous encoded frames. In order to "enjoy the best of both worlds", a classification scheme is used to partition an image into two types of regions: *structure regions (SRs)* and *nonstructure regions (NSRs)*. Structure regions are encoded with super-spatial prediction while NSRs can be efficiently encoded with conventional image compression methods, such as CALIC. It

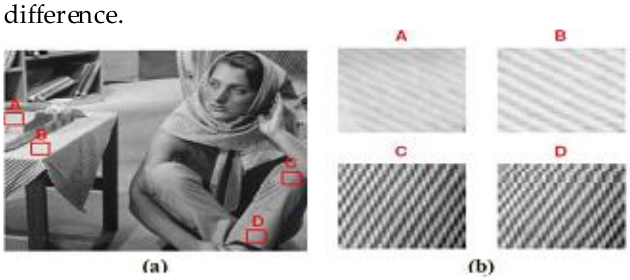
- *Seema G Bhateja is currently pursuing masters degree in electronics and-telecommuniation engineering in VESIT, Chembur, Mumbai University, India,. E-mail: seemagbhateja@gmail.com*
- *C.S.Rawat is pursuing Phd in ECE Dept., NIT Rourkela. E-mail:chandansrawat@gmail.com*
- *Dr. Sukadev Meher is currently working as Prof. and Head of ECE Dept, NIT Rourkela.*

is also important to point out that no codebook is required in this compression scheme, since the best matches of structure components are simply searched within encoded image regions.
This paper is organized as follows:
Section 2 expalins the algorithm used to predict the image using direct prediction method wherein an optimal prediction of structure components is done within the previously encoded image regions. Also gives an explanation for different modes that can be used for prediction of image blocks. Section 3 explains the residue encoding scheme used which helps in retrieving the lossless image at the decoder. Section 4 gives detail about compressing the nonstructural areas using CALIC. The block diagram of the complete algorithm is given in next section and at the end simulation results in RGB Colorspace is given, where the algorithm was tested.

2 SUPER-SPATIAL STRUCTURE PREDICTION

A real world scene often consists of various physical objects, such as buildings, trees, grassland, etc. Each physical object is constructed from a large number of structure components based upon some predetermined object characteristics. These structure components may repeat themselves at various locations and scales Fig. 1. Therefore, it is important to exploit this type of data similarity and redundancy for efficient image coding.

The Super Spatial Structure Prediction borrows its idea from motion prediction [3] Fig.2. In motion prediction Fig. 2(b), we search an area in the reference frame to find the best match of the current block, based on some distortion metric. The chosen reference block becomes the predictor of the current block. The prediction residual and the motion vector are then encoded and sent to the decoder. In similar structure block prediction Fig.2(a), we search within the previously encoded image region to find the prediction of an image block. The reference block that results in the minimum block difference is selected as the optimal prediction. For simplicity, we use the sum of absolute difference (SAD) to measure the block difference.



(a) Barbara image. (b) Four image blocks extracted from Barbara
Fig. 1 Example for Super-Spatial Structure Block Redundancies

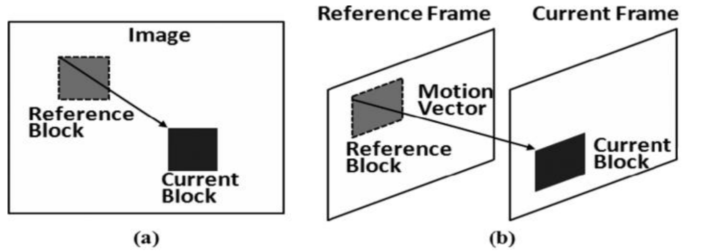


Fig.2. (a) Super-Spatial structure prediction. (b) Motion prediction in video coding.

As in video coding [3], we need to encode the position information of the best matching reference block. To this end, we simply encode the horizontal and vertical offsets, between the coordinates of the current block and the reference block using context-adaptive arithmetic encoder. The size of the prediction unit is an important parameter in the similar structure block prediction. When the unit size is small, the amount of prediction and coding overhead will become very large. However, if we use a larger prediction unit, the overall prediction efficiency will decrease. In this work, we attempt to find a good tradeoff between these two and propose to perform spatial image prediction on block basis.

2.1 Image Block Classification

A block-based image classification scheme is used here. The image is partitioned into blocks of 8x8. We then classify these blocks into two categories: structure and nonstructure blocks. Structure blocks are encoded with super-spatial prediction. Nonstructure blocks are encoded with conventional lossless image compression methods, such as CALIC.

2.2 Estimation of Threshold

The threshold is required while comparing the current block with the previous encoded region. This threshold value should be so decided that it will give best compression performance.

2.3 Prediction Modes

In this scheme using 4 × 4 blocks, nine modes of prediction are supported. A 4 × 4 block of pixels labeled "a" through "p" are predicted from a row of eight pixels labeled "A" through "H" above the current block and a column of four pixels labeled "I" through "L" to the left of the current block as well as a corner pixel labeled "M," as shown in Fig 3. The nine modes of 4 × 4 blocks are mode 0 (vertical prediction), mode 1 (horizontal prediction), mode 2 (DC prediction), mode 3 (diagonal down/left prediction), mode 4 (diagonal down/right prediction), mode 5 (vertical-right prediction), mode 6 (horizontal-down prediction), mode 7 (vertical-left prediction), and mode 8 (horizontal-up prediction). Out of the nine modes the mode that results in minimum SAD is the chosen block.

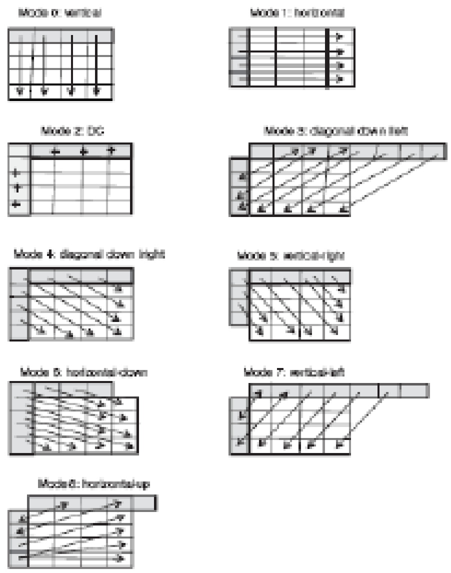


Fig.3 Nine modes of Prediction used

2. 4 CALIC

The Context Adaptive Lossless Image Codec (CALIC) scheme, uses both context and prediction of the pixel values. CALIC employs a two-step (prediction/residual) approach. In the prediction step, CALIC [1] employs a simple new gradient based non-linear prediction scheme called GAP (gradient-adjusted predictor) which adjusts prediction coefficients based on estimates of local gradients. Predictions then made context-sensitive and adaptive by modeling of prediction errors and feedback of the expected error conditioned on properly chosen modeling contexts. The modeling context is a combination of quantized local gradient and texture pattern, two features that are indicative of the error behavior. The net effect is a non-linear, context-based, adaptive prediction scheme that can correct itself by learning from its own past mistakes under different contexts.

CALIC encodes and decodes images in raster scan order with a single pass through the image. The coding process uses prediction templates that involve only the previous two scan lines of coded pixels. Consequently, the encoding and decoding algorithms require a simple double buffer that holds two rows of pixels that immediately precede the current pixel, hence facilitating sequential build-up of the image.

In the continuous-tone mode of CALIC, the system has four major integrated
components: -

- Prediction
- Context selection and quantization
- Context modeling of prediction errors
- Entropy coding of prediction errors.

CALIC is a spatial prediction based scheme, in which GAP is used for adaptive image prediction [1].

Here GAP prediction is performed on the original image and the prediction error for each block is computed. If the prediction error is larger than a given threshold, then it is considered as a structure block. Otherwise, it is classified as a nonstructure block.

3 RESIDUE ENCODING

The implemented image compression scheme is purely lossless, the residues need to be transmitted along with the image. But this will increase the payload size and thus the compression will not be achieved successfully. The residues are encountered in two places: - The CALIC Algorithm and the SAD residues. Arithmetic coding [7], [8] schemes are to be used to transmit the residues to further reduce the size of the overhead data per block.

Arithmetic coding is especially useful when dealing with sources with small alphabets, such as binary sources, and alphabets with highly skewed probabilities. It is also a very useful approach when, for various reasons, the modeling and coding aspects of lossless compression are to be kept separate. In arithmetic coding a unique identifier or tag is generated for the sequence to be encoded. This tag corresponds to a binary fraction, which becomes the binary code for the sequence.

In order to distinguish a sequence of symbols from another sequence of it has to be tagged with a unique identifier. One possible set of tags for representing sequences of symbols are the numbers in the unit interval (0, 1). Because the number of numbers in the unit interval is infinite, it should be possible to assign a unique tag to each distinct sequence of symbols. In order to do this we need a function that will map sequences of symbols into the unit interval. A function that maps random variables, and sequences of random variables, into the unit interval is the cumulative distribution function *(cdf)* of the random variable associated with the source. This is the function to be used in developing the arithmetic code.

4 THE COMPLETE ALGORITHM

The complete algorithm used for this lossless image compression scheme can be categorized into two main parts as listed below.

4.1 Proposed Encoder

The original image is subjected to Similar Structure Block Prediction Algorithm. This produces a Lossy Compressed Image and a set of residues. The residues are then encoded using Arithmetic Coding. The Lossy Compressed Image along with the encoded residues forms the compressed data as shown in Fig. 4.

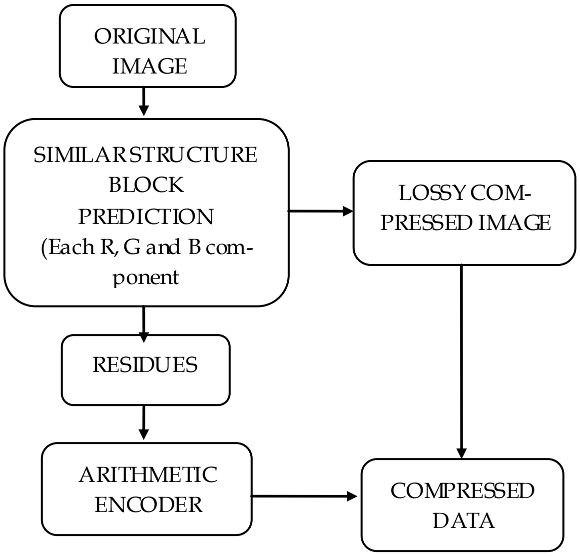


Fig. 4 Proposed Encoder

4.2 Proposed Decoder

The compressed data consisting of Lossy Compressed Image and encoded residues is then given as inputs to the decoder. The encoded residues are given to the Arithmetic Decoder to obtain the original set of residues which is then added to the Lossy Compressed Image to reconstruct the Final Image as shown in Fig. 5.

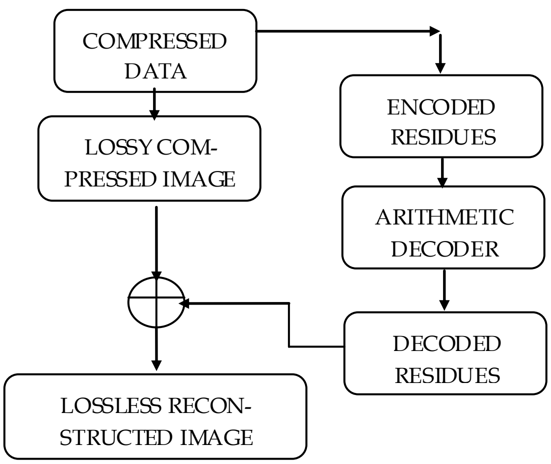


Fig. 5 Proposed Decoder

5 Simulation Results

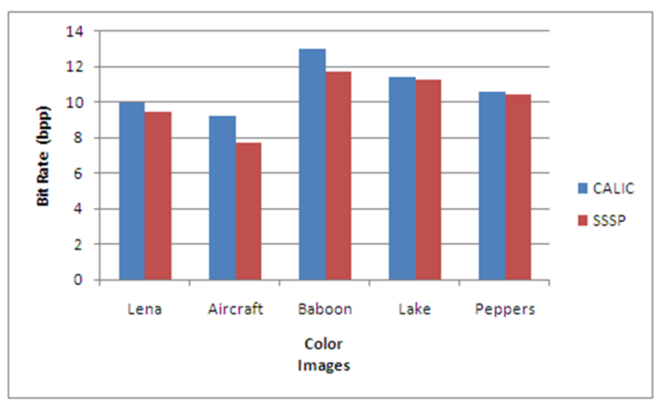
All the simulations were done using MATLAB 7.11 (R2010b) on RGB Colorspaces on standard Images (Fig 6) size of 512x512 pixels like Lena, Aircraft, Baboon, Lake and Peppers.

Fig.6 Standard RGB Colorsapce Test Images used

The RGB Colorspace images are first divided into each R, G and B component and individually the algorithm is applied into each component. The bit rate for the compressed color image is calculated and as shown in graph 1.

In graph 1, Bit rate of proposed algorithm is compared with CALIC for each test images. From graph we can observe that bit rate saving is more for baboon which has more structural regions.

The graph 2 shows the variation of bit rate for different percentage of structural regions for all test images. The result shown in graph 1 is the best case result when comparing with CALIC which has been obtained while changing the percentage of structure regions as shown in graph 2.

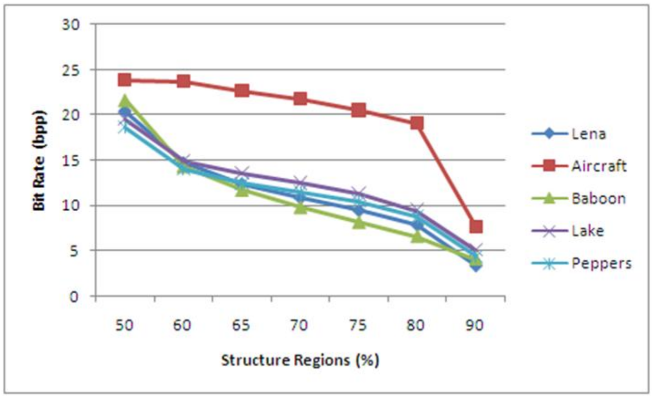


Graph 1 Compression Performance comparison of SSSP with CALIC

Graph 2 Variation of Bit Rate with the percentage of Structure Regions

6 Conclusion

In this endeavor a simple yet efficient lossless image compression algorithm based on structure prediction has been successfully designed and tested for RGB Colorspace. It is motivated by motion prediction in video coding, attempting to find an



optimal prediction of a structure components within previously encoded image regions. Taking CALIC as the base code, the image was classified into various regions and they were encoded accordingly. The extensive experimental results demonstrate that the proposed hybrid scheme is very efficient in lossless image compression, especially for images with significant structure components.

REFERENCES

[1] X. Wu and N. Memon, "Context-based, adaptive, lossless image coding," IEEE Trans. Commun., vol. 45, no. 4, pp. 437–444, Apr. 1997.

[2] M. J. Weinberger, G. Seroussi, and G. Sapiro, "The LOCO-I lossless image compression algorithm: Principles and standardization into JPEG-LS," IEEE Trans. Image Process., vol. 9, no. 8, pp. 1309–1324, Aug. 2000.

[3] T. Wiegand, G. J. Sullivan, G. Bjntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," IEEE Trans. Circuits Syst. Video Technol., vol. 13, no. 7, Jun. 2003.

[4] B. Wohlberg and G. de Jager, "A review of the fractal image coding literature," IEEE Trans. Image Process., vol. 8, no. 12, pp. 1716–1729, Dec. 1999.

[5] Xiwen Zhao and Zhihai He," Local Structure Learning and Prediction for Efficient Lossless Image Compression", 2010 IEEE, pp. 1286-1289.

[6] Xiwen OwenZhao, and Zhihai HenryHe," Lossless Image Compression Using Super-Spatial Structure Prediction", IEEE SIGNAL PROCESSING LETTERS, VOL. 17, NO. 4, APRIL 2010, pp. 383-386.

[7] David Salomon, *"Data Compression: A Complete Reference"*,3rd Edition, Springer, 2007.

[8] Khalid Sayood, *"Introduction to Data compression"*, 3rd Edition, Elseiver Publications, 2006.

[9] X. Wu, *"Lossless compression of continuous-tone images via context selection, quantization, and modeling"*, IEEE Trans. Image Processing,Vol.6, No.5 (pp.656-664) 1997.

[10] Still Image and Video Compression with Matlab, by K.S. Thyagarajan, a John Wiley & Sons, Inc, Publication.

[11] Hao Hu, "A Study of CALIC", A paper submitted to the Computer Science & Electrical Engineering Department at University of Maryland Baltimore County, December 2004.

[12] Grzegorz Ulacha, and Ryszard Stasiński, "Effective Context Lossless Image Coding Approach Based on Adaptive Prediction", World Academy of Science, Engineering and Technology, 2009.

[13] F. Wu and X. Sun, "Image compression by visual pattern vector quantization (VPVQ)," in *Proc. IEEE Data Compression Conf. (DCC'2008)*, pp. 123–131.

[14] Mauro Barni, "Document and Image Compression", Taylor and Francis, 2006.

[15] Rafael Gonzalez & Richard Woods, "Digital Image Processing using MATLAB", Tata McGraw Hill, 2010.