

2010

A new predictive image compression scheme using histogram analysis and pattern matching

F. Keissarian

University of Wollongong in Dubai, farhadk@uow.edu.au

Follow this and additional works at: <https://ro.uow.edu.au/dubaipapers>

Recommended Citation

Keissarian, F.: A new predictive image compression scheme using histogram analysis and pattern matching 2010.
<https://ro.uow.edu.au/dubaipapers/253>

A New Predictive Image Compression Scheme using Histogram Analysis and Pattern Matching

Farhad Keissarian

University of Wollongong in Dubai

farhadkeissarian@uowdubai.ac.ae

Abstract—In this paper, we introduce a new predictive image compression scheme that compresses an image by a set of parameters computed for individual blocks of different types. These parameters include the average and difference of the representative intensities of an image block, together with the index of a pattern associated with the block visual activity. The block representative gray values are computed through a histogram analysis of the block residuals and a pattern matching technique is employed to find the best match for the block bit-pattern from a pre-defined pattern book. To further reduce the bit rate, a predictive technique selectively predicts the parameters based on the corresponding values in the neighboring blocks. The simulation results confirm that the proposed technique can provide a high compression ratio with acceptable image quality of the compressed images.

Keywords—component; image compression; block patterns; block histograms, predictive coding.

I. INTRODUCTION

Efficient image compression solutions are becoming more critical with the recent growth of data intensive, multimedia-based web applications. There are numerous methods for compressing digital image data and each has its own advantages and disadvantages [1].

Vector quantization (VQ) [2], block truncation coding (BTC) [3]-[4], and transform coding [5] are three commonly used schemes for image compression. VQ is widely used because it has a simple decoding structure and requires a low bit rate. However, a lot of computational costs are needed in the codebook design procedure and the image coding procedure. In general, BTC has the advantages of requiring very low computational cost and providing good reconstructed image quality. However, a high bit rate is needed in BTC. Predictive coding scheme is exploited due to existing the similarity among neighboring image blocks and is adopted to image compression techniques.

The study in [6] presents a predictive image compression scheme that combines the advantages of vector quantization and moment preserving block truncation coding. The prediction technique employs the squared Euclidean distance between two blocks to find the similarity of the current block with a neighboring block. If a similar compressed image block can be found in the neighborhood of current processing block, it is taken to encode this block. Otherwise, this image block is encoded either by vector quantization or moment preserving block truncation coding. A hybrid image

coding scheme that combines the advantages of AMBTC, predictive coding scheme, VQ, and DCT coding algorithms is presented in [7]. In this method, the predictive coding scheme combined with VQ is used in encoding the bit-map blocks generated by AMBTC. In this study, Hamming distance is used to measure the difference rather than Euclidean distance. Hamming distances of the current block with the neighboring blocks are compared with a pre-defined threshold, and the current block is replaced with the neighboring block with the minimum Hamming distance.

In this paper, we present a modified scheme of the image compression technique that we reported earlier in [8], by incorporating a predictive technique. The proposed technique compresses an image block by transmitting a set of parameters computed through a histogram analysis of residual blocks and a pattern matching scheme. Moreover, for exploiting the correlation between the adjacent image blocks, these values are selectively predicted based on the corresponding values in the neighboring blocks. The rest of the paper is organized into four sections. The concept of the proposed compression algorithm is introduced in Section 2. In section 3, the predictive coding scheme is presented. Experimental results are given in Section 4.

II. IMAGE COMPRESSION ALGORITHM

In the proposed image compression algorithm in [8], an image is block coded according to the type of individual blocks. A novel classifier, which is designed based on the histogram analysis of residual blocks, is employed to classify the blocks according to their level of visual activity. The classifier places each block into one of the two categories of uniform or edge block. A uniform block is coded by the block mean, whereas an edge block is coded by a set of parameters associated with the pattern appearing inside the block. Like the original BTC algorithm [3], our method encodes an edge block by initially computing two gray values and constructing a bit-map. However, in the proposed method the computation of the gray values, namely the low and high representative intensities are carried out through analysis of the block residuals' histogram. Moreover, instead of transmitting the two gray values, their average and difference will be sent to decoder. Finally, instead of transmitting the whole bit-map for the processed edge block, an optimum bit-pattern is selected from a set of pre-defined patterns, and its index will be transmitted. The use of these parameters at the receiver reduces the cost of reconstruction

significantly and exploits the efficiency of the proposed technique. A brief description of the block classifier and the coding scheme are given in the next two sub-sections.

A. Block Classifier

A novel histogram-based classification scheme has been developed for classifying the image blocks [6]. The method operates based on the distribution of the block residuals and classifies block either as a low-detail (uniform) or as a high-detail (edge) block. The classifier employs the block residuals and classifies the block according to their histogram. The classification is carried out through a peak detection method on the histogram. A brief description of the classifier is as follows:

Each block of 4x4 pixels is converted into a residual block by subtracting the sample mean from the original pixels. The residual samples are less correlated than the original samples within a block. Here, two of the most important local characteristics of the image block are considered: *central tendency*, represented by the mean value and the *dispersion* of the block samples about the mean, which is represented by the residual values. The challenge here is to analyze the dispersion of the residual values about the mean. One way of achieving this is to sort the histogram of the block residual samples.

As the neighboring pixels in the original block are highly correlated, the residual samples will tend to concentrate around zero. One can then quantize the residual samples prior to forming the histogram. The histogram of the quantized residuals may then be formed and analyzed by simply detecting its peaks. Based on the distribution of the residual samples within the test images, we choose to apply a coarse quantization, in particular a 15-level non-uniform quantizer. We now define q_j as the output of the quantizer

with index j , as shown in “Fig. 1”. The histogram of the quantized values $h(q_j)$ may then be formed to provide the occurrence of q_j . The quantized residual histogram (QRH) is then analyzed by simply detecting its peaks. According to the number of detected distinct peaks on the histogram, image blocks can be placed into two major categories of uniform and edge blocks. A histogram with a unique peak at its centre (uni-modal histogram) identifies a uniform block. Whereas, the existence of two distinct peaks implies that the processed block is an edge block. “Fig. 2” and “Fig. 3” show examples of both types of blocks.

A peak on the histogram indicates a high score of residual values; therefore it is fair to conclude that there is a considerable number of pixels that have the same dispersion about the block mean. This, in turn will lead us to conclude that the gray level values of these pixels are very close to one another. Hence, this group of pixels can be represented by a single gray value. In this analysis, a distinct peak on the QRH of the processed block represents a gray value X_j , given as :

j	-7	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7
q_j	-67	-52	-39	-28	-17	-10	-5	0	5	10	17	28	39	52	67

Figure 1. The quantizer output q_j with index j

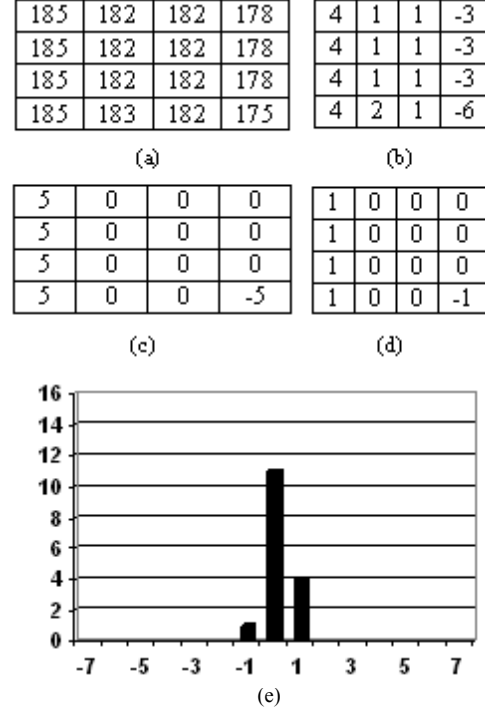


Figure 2. (a) Original uniform block; sample mean = 181, (b) Residuals (c) quantized residuals, (d) quantizer's indexes, (e) block QRH

$$X_j = X_{mean} + q_j \quad (1)$$

where X_{mean} is the block mean. For a uniform block, since the single peak occurs at the center of the histogram, where $q_j = 0$, then from Eq.1 the representative intensity X_j will be the same as the block mean. For an edge block, the two peaks of the QRH, which are positioned on the left and right hand side of the centre ($j=0$) represent the low representative intensity X_L and the high representative intensity X_H , respectively. If the two peaks are positioned at indexes j' and j'' , the two representative intensities are calculated as:

$$\begin{aligned} X_L &= X_{mean} + q_{j'} \\ X_H &= X_{mean} + q_{j''} \end{aligned} \quad (2)$$

In “Fig. 3”, $X_{mean} = 125$, and X_L and X_H are computed using Eq. 2 ; $X_L = 125 + (-39) = 86$ and $X_H = 125 + 28 = 153$.

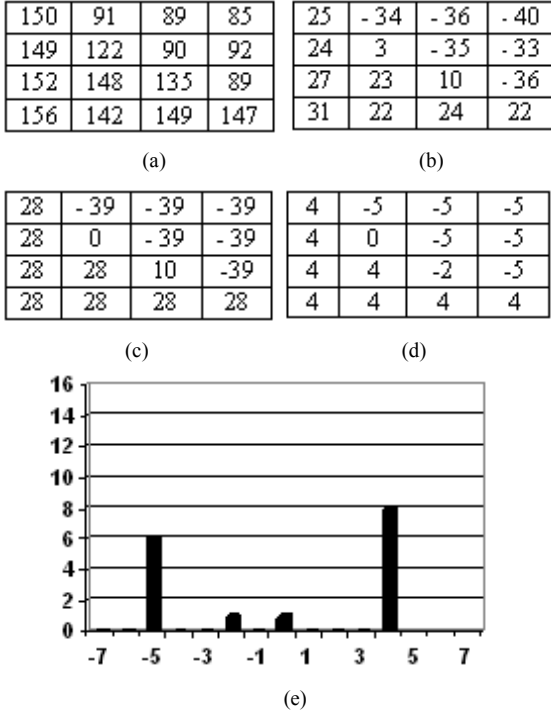


Figure 3. (a) Original edge block: sample mean = 125 (b) Residuals (c) quantized residuals, (d) quantizer's indexes, (e) QRH

B. Coding of Image Blocks

Once the image blocks have been classified, the coder, switches between a one-level (uniform block) and a bi-level (edge block) representation. A uniform block is encoded by transmitting the block mean plus an indicator to inform the decoder that the block is uniform. By forcibly clustering all pixels in an edge block into two groups, a bi-level approximation of the block is obtained. The clustering partitions a block W into two sets of pixels, W_0 and W_1 , such that $W = W_0 \cup W_1$ and $W_0 \cap W_1 = \Phi$. The clustering is carried out by marking the pixels of set W_0 and W_1 by '0' and '1', respectively. Thus the clustering can be represented as a bit-pattern, $B = \{b_1, b_2, \dots, b_{16} | b_i \in (0,1)\}$. By selecting the block mean as a threshold, the bit-pattern can be generated as :

$$b_i = \begin{cases} 1 & \text{if } x_i > X_{mean} \\ 0 & \text{if } x_i \leq X_{mean} \end{cases} \quad (3)$$

where, $x_i \in W$ are the intensities of the pixels of the edge block. It is noted that, X_L and X_H are the representative intensities of the set W_0 and W_1 , respectively. Like the original BTC, an edge block can be coded by transmitting

the representative intensities and the bit-pattern. However, in our method, we transmit the average, M and difference, l of the representative intensities, defined by :

$$M = \frac{X_H + X_L}{2}$$

$$l = \frac{X_H - X_L}{2} \quad (4)$$

The values M and l represent the low and high frequency components, respectively. It is evident from eq.3 that $X_H = M + l$ and $X_L = M - l$. During the reconstruction, the coded block can be constructed by :

$$b_i = \begin{cases} M + l & \text{if } b_i \in W_1 \\ M - l & \text{if } b_i \in W_0 \end{cases} \quad (5)$$

It should be noted that for a uniform block, since both representative intensities are the same as the block mean, therefore, $M = X_{mean}$ and $l = 0$. Instead of transmitting the whole bit-pattern of an edge block, further bit reduction can be achieved by finding the best match for the block bit pattern from a set of pre-defined patterns, P_k , $k = 0, 1, 2, \dots, N$. A set of 32 patterns shown in "Fig. 4", which preserve the location and polarity of edges in four major directions and their complements making $N=64$ is used in our method. The pattern matching stage is carried out by performing a logical exclusive NOR operation on the block bit-pattern and each pattern from the set to calculate a matching score, ms , given as :

$$ms = \sum_{i=0}^3 \sum_{j=0}^3 (P_{ij} \oplus b_{ij}) \quad (6)$$

The pattern with the highest ms is selected and its index k will be transmitted. Since, the proposed method sends k instead of the whole block bit-pattern, only $\log_2^{64} = 6$ bits are transmitted. Each image block is therefore encoded by generating a triple (M, l, k) . "Fig. 5" illustrates an edge block with block mean = 125, its bit-pattern, the selected pattern from the set ($k=23$) as well as the reconstructed block. The reconstructed values were calculated in the previous sub-section, from Eq. 2. Using Eq. 4, M and l are computed as (120, 34, 23) to generate the compression code. It should be also noted that, since for a uniform block, no pattern index is transmitted, therefore the compressed code for such a block is the pair (M, l) , where $l = 0$. The value M in the triple (M, l, k) , can be coded by 8 bits, whereas coding l requires only 6 bits, as its standard deviation is smaller than that of X_L and X_H . Therefore, the compression code of an edge block requires $20 = (8+6+6)$ bits to be transmitted. For a uniform block, the number of bits required to code the pair $(M, 0)$ is $9 = 8+1$. The bit-rate in bits per pixel (bpp) for an 8-bit gray level image size of

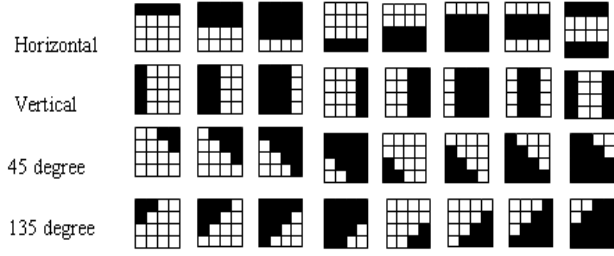


Figure 4: Set of 32 pre-defined patterns.

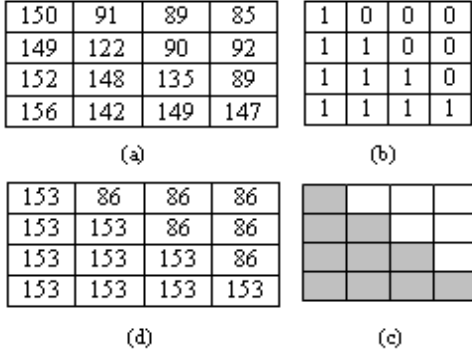


Figure 5. (a) Original edge block, (b) Bit-pattern
(c) Matched pattern, (d) Reconstructed block

$w \times h$, with N_u uniform blocks, and N_e edge blocks is given by :

$$bpp = \frac{(9N_u) + (20N_e)}{w * h} \quad (7)$$

III. PREDICTIVE CODING

As mentioned earlier that M is bias or average intensity for both types of blocks, it has smaller variation across adjacent blocks. In particular, the M value of an edge block has a stronger correlation with that of its neighboring blocks compared to X_L and X_H , because of spatial homogeneity. This suggests that if we use some form of selective prediction on the M value of a block based on the M values of the previous blocks then a further reduction in bpp can be achieved. “Fig.6” shows the blocks that are used for prediction of M . These are the neighboring left, upper, upper left and upper right blocks for the current block.

Suppose M values of the blocks numbered 1, 2, 3, 4, C are M_1 , M_2 , M_3 , M_4 , and M_c , and are known. In order to check the closeness of the M value of the current block with the M values of the neighboring blocks, we define the correlation factor R given as :

$$R_\alpha = \frac{M_c - M_\alpha}{M_c}, \quad \alpha \in \{1, 2, 3, 4\} \quad (8)$$

Used for Prediction 2	Used for Prediction 3	Used for Prediction 4
Used for Prediction 1	Current Block C	

Figure 6. Blocks used in prediction of M in current block

The factor R for each neighboring block is computed from Eq. 8 and the smallest one is compared against a pre-defined threshold Th . If it differs below the threshold then the M value of the block having the minimum R is used for the current M value. The steps involved in predicting the M values are as follows :

1. Encode the blocks starting from left to right and top to bottom sequence.
2. If the block is in the first row or first column then go to step 5.
3. Obtain the value of M for the processed block and calculate R_α for the neighboring blocks.
4. Find $R_{\min} = \text{minimum}(R_1, R_2, R_3, R_4)$
 IF $R_{\min} < Th$ THEN
 Case $R_{\min} = R_1$, encode M as 00
 Case $R_{\min} = R_2$, encode M as 01
 Case $R_{\min} = R_3$, encode M as 10
 Case $R_{\min} = R_4$, encode M as 11
 End case
 End IF
5. IF the processed block is uniform, THEN transmit the pair $(M, 0)$,
 Else transmit the triple (M, l, k)
6. Go to step 2 until all the blocks are processed.

IV. SIMULATION RESULTS

We have evaluated the performance of the proposed coding scheme through a computer simulation on two gray-level images ‘Lena’ and ‘Pepper’, shown in ‘Fig.7a’ and ‘Fig. 8a’. These images are 8 bits per pixel and 512 x 512 pixels in size. The simulation platform is Microsoft Windows XP, Pentium III, and the proposed scheme is implemented using Matlab. Two performance matrices are used to measure the performance of the proposed compression schemes : compression ratio (bpp), and image quality. The peak signal-to-noise ratio (PSNR) is used to evaluate image quality of a compressed mage generated by the proposed scheme.

The reconstructed images of the test images without prediction are shown in ‘Fig. 7b’ and ‘Fig. 8b’. We implemented the prediction technique by setting the

threshold to $Th=0.1$ and $Th=0.2$. The results obtained in terms of bit-rate and coding quality are given in the Table I. It can be observed from Table I that the quality of reconstructed images using the smaller threshold is better than that for the larger threshold. The reconstructed images for different values of threshold are shown in Figs. 7c-7d and Figs. 8c-8d. The performance of the proposed method is compared with that of Wei *et al* [7]. The computed PSNR and bpp values for the two methods are given in Table II. From Table II, we observe that the proposed method outperforms Chuang *et al.*'s scheme in the quality, with almost the same bit-rate.

TABLE I

Image	P(u)	Th	N(p)	CR (bpp)	PSNR (dB)
Lena	63%	0.1	39%	0.624	30.42
		0.2	49%	0.596	30.15
Pepper	72%	0.1	37%	0.616	30.55
		0.2	46%	0.601	30.27

P(u): Population of uniform blocks, **Th :** Threshold, **N(p):** No. of prediction blocks, **CR :** Compression ratio

TABLE II

Image	Method	Compression Ratio (bpp)	PSNR (dB)
Lena	Wei <i>et al</i> [7]	0.592	27.80
	Proposed	0.596	30.15
Pepper	Wei <i>et al</i> [7]	0.593	28.33
	Proposed	0.601	30.27



(a) Original image



(b) Coded without prediction, bpp=0.76, PSNR= 32.30

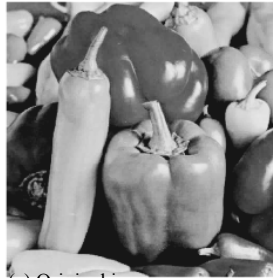


(c) Coded with prediction, $Th=0.1$ bpp=0.62, PSNR= 30.42



(d) Coded with prediction, $Th=0.2$ bpp=0.59, PSNR= 30.15

Figure 7. Compression results for the image of "Lena"



(a) Original image



(b) Coded without prediction, bpp=0.75, PSNR= 32.27



(c) Coded with prediction, $Th=0.1$ bpp=0.61, PSNR= 30.55



(d) Coded with prediction, $Th=0.2$ bpp=0.60, PSNR= 30.27

Figure 8. Compression results for the image of "pepper"

REFERENCES

- [1] P D. Solomon, "Data Compression The complete reference", 2nd edition, Springer 2001, New York..
- [2] N. M. Nasrabadi and R. A. King, "Image Coding Using Vector Quantization: A Review", IEEE Trans. on Communs., vol. 36, 1988, pp. 957-971.
- [3] E. J. Delp and O. R. Mitchell, "Image Compression Using Block Truncation Coding", IEEE Trans. on Communs., vol. 27, 1979, pp. 1335-1342.
- [4] M. D. Lema and O. R. Mitchell, "Absolute Moment Block Truncation Coding and Its Application to Color Images", IEEE Trans. on Communs., vol. 32, 1984, pp. 1148-1157.
- [5] G. K. Wallace, "The JPEG still picture compression standard," IEEE Trans. Consumer Electronics, vol. 38, no. 1, pp. 18-34, Feb. 1992.
- [6] Y. C. Hu, "predictive grayscale image coding scheme using VQ and BTC", Fundamenta Informaticae, vol. 78, 2007, pp. 239-255.
- [7] C. H. Wei and T. J. Kuo, "A hybrid coding scheme for image compression", IEEE ICSS2005 Int. Conf. on Systems & Signals, 2005, pp. 457-461.
- [8] F Keissarian, M F Daemi, "Novel quad-tree image coding technique using edge-oriented classification, Proc. SPIE, Visual Communications and Image Processing (VCIP), part of AS&T / SPIE's Electronic Imaging, Vol. 4671, p. 985-992, January 20