社団法人　情報処理学会　研究報告
IPSJ SIG Technical Report

2003－ＡＶＭ－43　(31)
2003／12／19

複数走査による可逆画像圧縮法

唐 海江† 鎌田 清一郎† 小林 正明‡

†早稲田大学大学院情報生産システム研究科 〒808-0135 北九州市若松区ひびきの 2-7
‡パナソニックコミュニケーションズ株式会社 〒812-8531 福岡市博多区美野島 4-1-62
E-mail: †hitang@akane.waseda.jp, kam@waseda.jp

あらまし 本論文では，濃淡画像の可逆圧縮手法について述べる．本手法は，入力画像を複数の画素ブロックに分割し，異なる走査方法によって得られた隣接画素の統計的性質から適応的な線形予測モデルを適用する．予測係数は，符号化効率が良くなるように走査中に更新する．異なる走査による線形予測モデルのうち，予測誤差を最小化するものを選択する．実験の結果，JPEG-LS に比べて 4~5%圧縮効率のよいことが確認された．

キーワード 可逆圧縮、予測、複数走査

A Multi-scan Approach for Lossless Image Compression

TANG Haijiang† , KAMATA Sei-ichiro† and KOBAYASHI Masa-aki‡

† Graduate School of Information, Production and Systems, Waseda University, 2-7 Hibikino, Wakamatsu-ku, Kitakyushu-shi, 808-0135, Japan

‡ Panasonic Communications Co., Ltd., 4-1-62 Minoshima, Hakata-ku, Fukuoka-shi, 812-8531, Japan

E-mail: † hjtang@akane.waseda.jp, kam@waseda.jp

Abstract This paper presents an efficient lossless compression approach for gray scale images. We divide the input image to blocks of pixels and use different scanning methods for compression. An adaptive linear prediction model is applied corresponding to the statistics of neighbor pixels obtained from each scanning. The prediction coefficients are updated during the scanning to optimize the coding accuracy. From the models built by different scanning, we select the one that minimizes the prediction error for coding. Experiment results show that our method out-performed JPEG-LS 4~5% in compression efficiency.

Keyword lossless compression, prediction, multi-scan

1. Introduction

Predictive coding has been widely used in image compression. The idea is to remove redundancy in successive pixels by only encoding residual between actual and predicted data. The residual, or prediction error, hopefully has much smaller dynamic range than the original data, thus requires less bits to encode. The frame work of this technique consists of scanning the data array, constructing a model of the data, applying a predictor corresponding to the model, and then encoding the difference between the prediction and the actual outcome. That is, a given two dimensional data X is first converted to a one-dimensional sequence $\{x_i\}$, where each x_i is a pixel value. Simultaneously, the estimation of x_i is obtained and makes another sequence

$$\hat{x}_i = f(n_1, n_2, \ldots, n_k) . \qquad (1)$$

where $f(\cdot)$ is the prediction function and $n_j, j = 1, \ldots, k$ are the neighbor pixels chosen by the predictor, and the total residual is $\sum_i | x_i - \hat{x}_i |$. Our motivation is: suppose the predictor used to generate $\hat{x}_i$ is fixed, choosing the optimal scanning can minimize the total residual.

Many works have been conducted on analysis of scanning techniques, e.g. [1], [2]. However, these works only theoretically discussed how the scanning could affect the compression, but no actual optimal scanning strategy was proposed. Some works argued that the Hilbert scan is sometimes the best method, e.g. [3], [4], yet [1] proved that Hilbert scan is still not a global optimal for image compression. In fact, many works (e.g. [1], [5]) proved that there is no single universal optimal scanning for all images.

In this paper, we use different scan methods along with predictive estimation, and then choose the best residual obtained for coding. Our experimental results show that by employing a few simplistic scanning schemes the compression efficiency is significantly improved without serious increasing in computational cost.

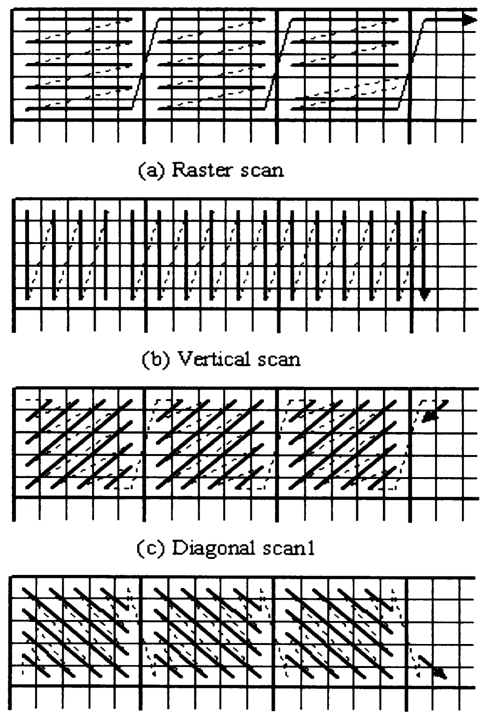
The rest of this paper is organized as follows: in the next section we introduce our multi-scan strategy. Section 3 discusses the linear prediction techniques and a new adaptive linear predictor is proposed. This new method is a modified version of adaptive linear coding prediction (ALCM). The experimental results and some discussions are presented in section 4.

2. Multi-scan strategy for predictive coding

In predictive coding, each predicted pixel value depends on a small number of neighboring pixels defined by a specific scanning. Thus the encoding may substantially depend on the scanning method. The amplitude of the residual has a great rising when the target pixel has a significant difference with its contexts. Therefore we should choose the scanning that try to avoid crossing edges. For example to compress an image like fig.1, vertical scan is obviously better than raster scan.

Fig. 1

In this paper, we apply four different scanning techniques: raster scan, vertical scan and two diagonal scan as shown in fig 2. The prediction error is computed as the difference between the actual pixel value at position x and its predicted value (fig. 3).



(a) Raster scan

(b) Vertical scan

(c) Diagonal scan1

(d) Diagonal scan2

Fig. 2 Scan orders for square block data

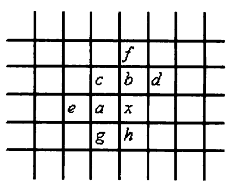


Fig. 3 neighbor pixels for prediction

The specific prediction transform for each scanning is described as follows:

Horizontal scan:

$$\hat{x} = w_{ah}a + w_{bh}b + w_{ch}c + w_{dh}d + w_{eh}e$$

Vertical scan:

$$\hat{x} = w_{av}a + w_{bv}b + w_{cv}c + w_{fv}f + w_{gv}g$$

Diagonal scan1:

$$\hat{x} = w_{ad1}a + w_{bd1}b + w_{cd1}c + w_{dd1}d + w_{fd1}f$$

Diagonal scan2:

$$\hat{x} = w_{ad2}a + w_{cd2}c + w_{ed2}e + w_{gd1}g + w_{hd1}h \ .$$

After the predictive transform, the prediction error $e(x) = x - \hat{x}$ of each pixel x is evaluated and we select the most appropriate residual for coding. This strategy is illustrated in fig 4. The residual selection approach is empirically determined and presented in the last section.

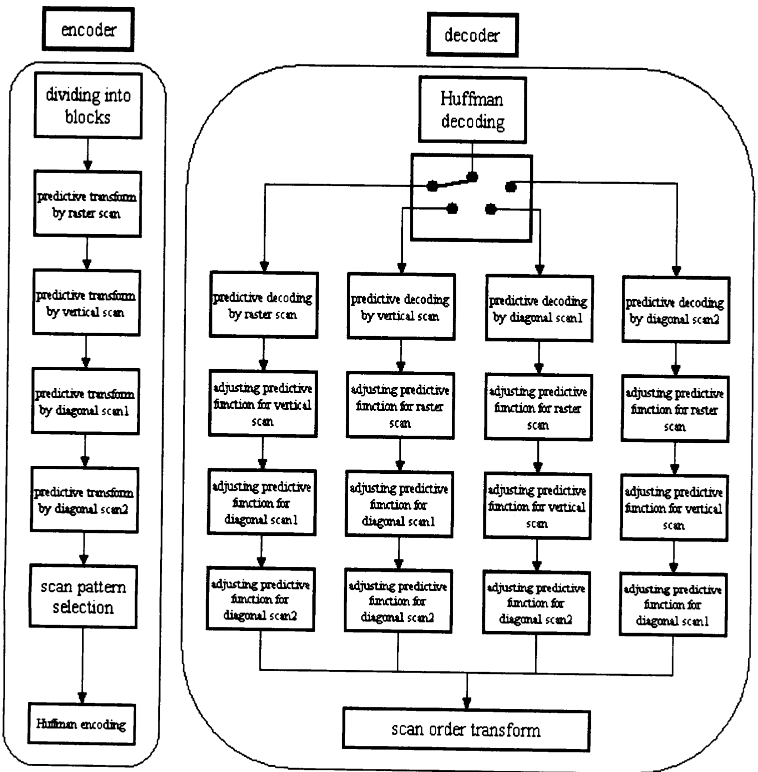


Fig.4: Construction of lossless compression

3. Predictive coding

In this section, we begin with an introduction to prediction coding techniques, including two specific predictors: MED and ALCM. Then we present our new predictive coding scheme, which is a revised version of ALCM.

3.1. MED and ALCM

Many predictors have been proposed for lossless image compression. Among them, the median edge detection predictor (MED) proposed in LOCO-I [6] has a great simplicity as well as efficiency. MED detects horizontal or vertical edges by three contexts of the target pixel (refer to fig. 3):

$$\hat{x} = \begin{cases} \min(a,b) & \text{if } \ c \geq \max(a,b) \\ \max(a,b) & \text{if } \ c \geq \max(a,b) \\ a+b-c & \text{otherwise} \end{cases}$$

The idea of MED is: if a vertical edge occurs, b is used as the prediction of x; while there's a horizontal edge, a is chosen for prediction. Finally, if there's no edge, planar interpolation is used. Despite its simplistic form, MED is proved to outperform many linear predictors [7].

Activity level classification model (ALCM) is an adaptive linear predictor in the form of $\hat{x} = \sum w_i n_i, i = 1,\ldots,k$, where $\hat{x}$ is the predicted value of pixel x , n_i are neighbor pixels of x , w_i are weights and k is the total number of neighbors acquired. In the original proposal of ALCM, k equals to 5. It choose the context pixels through raster scan and

$$\hat{x} = w_a a + w_b b + w_c c + w_d d + w_e e$$

where w are the weights for each context pixel value (refer to fig. 3), and they must satisfy:

$$w_a + w_b + w_c + w_d + w_e = 1$$

Initially, all the weights are assigned equal. As the prediction goes on, the weights are changed as follows. If $\hat{x} < x$ then the largest context pixel's weight is decremented by $\frac{1}{256}$ and the smallest context pixel's weight is incremented by $\frac{1}{256}$. If $\hat{x} > x$, the weight of the largest context pixel is decremented while the weight of the smallest pixel is incremented. There is an arbitrary sequence to decide the weight changing priority when there are more than one pixels have the highest or lowest value.

3.2. Modified ALCM (MALCM)

In this sub-section, we propose a modified version of ALCM by changing the adaptation value in ALCM from $\frac{1}{256}$ to $\frac{1}{256}e(x)$. ALCM adapts the weights each time by a fixed number, however, this does not reflect the actual change of prediction error. In ALCM, the prediction error $e(x)$ is decided by:

$$e(x) = x - \hat{x}$$

$$= x - (w_a a + w_b b + w_c c + w_d d + w_e e)$$

$$= x - \vec{w}\vec{r}^T \qquad (3)$$

where $\vec{w} = [w_a, w_b, w_c, w_d, w_e]$ is the weight vector and $\vec{r} = [a,b,c,d,e]$ is the context vector. The weights must satisfy

$$w_a + w_b + w_c + w_d + w_e = 1$$

$$\Delta w_a + \Delta w_b + \Delta w_c + \Delta w_d + \Delta w_e = 0 \qquad (4)$$

We want to change the weight vector as the prediction going on so that minimize $|e(x)|$. This is equivalent to minimize $\varepsilon = |e(x)|^2 = (x - \vec{w}\vec{r}^T)^2$. That is, we need to determine the change of the weight factor so that

$$\Delta\varepsilon = [x - (\vec{w} + \Delta\vec{w})\vec{r}^T]^2 - (x - \vec{w}\vec{r}^T)^2 \le 0 \qquad (5)$$

It can be easily found that:

$$\Delta\varepsilon = [x - (\vec{w} + \Delta\vec{w})\vec{r}^T]^2 - (x - \vec{w}\vec{r}^T)^2$$

$$= [(x - \vec{w}\vec{r}^T) - \Delta\vec{w}\vec{r}^T]^2 - (x - \vec{w}\vec{r}^T)^2$$

$$= -2(x - \vec{w}\vec{r}^T)\Delta\vec{w}\vec{r}^T + (\Delta\vec{w}\vec{r}^T)^2$$

We neglect the second order term of $\Delta\vec{w}$, and get

$$\Delta\varepsilon = 2(\vec{w}\vec{r}^T - x)\Delta\vec{w}\vec{r}^T = -2e(x)\Delta\vec{w}\vec{r}^T \qquad (6)$$

Thus we can easily choose the $\Delta\vec{w}$ satisfies (4) by let $\Delta\vec{w} = \vec{o}e(x)$ where $\vec{o}\vec{r}^T \ge 0$. In practice we can choose $\vec{o}$ to be very small to make sure $\vec{o}\vec{r}^T \ge 0$. Considering (4) and computational cost, we propose our modified version of ALCM as described at the beginning of this sub-section. Table 1 shows the zero order entropy of prediction error of ALCM and our method MALCM.

Table 1: zero order entropy of prediction error using ALCM and MALCM (bit/pixel)

	ALCM	MALCM
Barb1	4.582	4.418
Barb2	4.861	4.777
Boats	4.132	4.031
Gold	4.603	4.571
Zelda	3.799	3.738
Hotel	4.688	4.579
Lena	4.628	4.563
Peppers	4.571	4.496
Baboon	6.044	6.064
Average	4.657	4.582

4. Results and discussion

Experiments were performed to evaluate our methods and compare with other methods. The results are reported by average number of bits to encode a pixel, i.e., zero order entropy. Table 2 and 3 show how the block size affects the linear predictive coding. As we can see, dividing the image into 16×16 blocks is an appropriate choice. Also, it is clear how our MALCM method is more efficient than ALCM.

Table 2: Relation between block size and bit rate of ALCM (bit/pixel)

	4×4	8×8	16×16	32×32
Barb1	4.488	4.3512	4.273	4.320
Barb2	4.685	4.604	4.568	4.652
Boats	3.945	3.875	3.854	3.914
Gold	4.513	4.457	4.444	4.569
Zelda	3.821	3.746	3.722	3.777
Hotel	4.453	4.392	4.380	4.494
Lena	4.591	4.526	4.503	4.499
Peppers	4.562	4.485	4.459	4.447
Baboon	6.106	6.054	6.029	6.024
Average	4.502	4.426	4.397	4.456

Table 3: Relation between block size and bit rate of MALCM (bit/pixel)

	4×4	8×8	16×16	32×32
Barb1	4.391	4.249	4.192	4.266
Barb2	4.649	4.572	4.510	4.616
Boats	3.895	3.818	3.799	3.883
Gold	4.483	4.435	4.414	4.562
Zelda	3.895	3.715	3.698	3.767
Hotel	4.379	4.319	4.306	4.448
Lena	4.564	4.497	4.475	4.498
Peppers	4.545	4.461	4.420	4.433
Baboon	6.077	6.050	5.941	6.080
Average	4.456	4.381	4.344	4.436

Table 4 presents the effectiveness of multi scan strategy. H denotes horizontal scan, V for vertical scan, and DD refer to the two diagonal scans. Form these results we can conclude that using multiple scanning approaches can achieve better compression performance; however, adding the two diagonal scans does not provide much improvement.

Experiments presented in Table 5 and 6 shows the study of scanning selection. We applied different criterion to determine which scanning's outcome is to be encoded. MALCM1, MALCM2 and MALCM3 use code length, zero order entropy, and total residual ($\sum|e(x)|$) respectively. The results show that entropy is the best selection criterion, though the three methods achieved similar performance and all of them significantly out performed JPEG-LS in the sense of compression rate. However, there is clearly a trade-off between compression efficiency and computational expenditure.

Table 4: Relation between scan pattern and bit rate (bit/pixel)

	ALCM(H)	MALCM(H)	ALCM (HV)	MALCM(HV)	ALCM(HVDD)	MALCM(HVDD)
Barb1	4.462	4.349	4.292	4.222	4.273	4.192
Barb2	4.650	4.595	4.565	4.530	4.568	4.510
Boats	3.974	3.904	3.846	3.807	3.854	3.799
Gold	4.490	4.477	4.430	4.418	4.444	4.414
Zelda	3.809	3.776	3.724	3.707	3.722	3.698
Hotel	4.433	4.394	4.347	4.309	4.380	4.306
Lena	4.569	4.553	4.502	4.491	4.503	4.475
Peppers	4.507	4.478	4.444	4.427	4.459	4.420
Baboon	6.062	6.036	5.941	5.939	6.029	5.941
Average	4.551	4.507	4.384	4.354	4.397	4.344

Table 5: Compression results of different methods (bit/pixel)

	JPEG-LS	MALCM1	MALCM2	MALCM3
Barb1	4.691	4.222	4.301	4.252
Barb2	4.686	4.530	4.604	4.577
Boats	3.933	3.807	3.856	3.837
Gold	4.477	4.418	4.458	4.445
Zelda	3.888	3.707	3.745	3.722
Hotel	4.382	4.309	4.380	4.364
Lena	4.607	4.491	4.533	4.517
Peppers	4.513	4.427	4.483	4.464
Baboon	6.037	5.939	6.050	6.017
Average	4.579	4.354	4.416	4.392

Table 6: Processing time of different methods (second)

	JPEG-LS	MALCM1	MALCM2	MALCM3
Barb1	0.330	0.700	0.590	0.600
Barb2	0.330	0.700	0.590	0.600
Boats	0.330	0.700	0.590	0.600
Gold	0.330	0.700	0.590	0.600
Zelda	0.330	0.700	0.590	0.600
Hotel	0.330	0.700	0.590	0.600
Lena	0.220	0.470	0.400	0.410
Peppers	0.220	0.470	0.400	0.410
Baboon	0.220	0.470	0.400	0.410
Average	0.293	0.623	0.527	0.537

Reference

[1] Nasir Memon, David Neuhoff, and Sunil Shende, "An analysis of scanning techniques for lossless image coding", *IEEE Transactions on Image Processing*, Vol. 9, No. 11, Nov. 2000, pp. 1837-1848.

[2] N. Merhav, and T. Weissman, "Scanning and Prediction in Multidimensional Data Arrays", *IEEE Transactions on Information Theory* 49(1), Seiten 65-82. 2003.

[3] A. Lempel and J. Ziv, "compression of two-dimensional data", *IEEE Transactions on Information Theory*, vol. IT-32 no. 1, pp1-8, 1996

[4] A. Perez, S. Kamata and E. Kawaguchi, "Arithmetic coding model for compression of landsat images", *Vis. Communication Image Processing*, pt. 2, vol. SPIE 1605, pp. 879-884, 1991

[5] A. Quin and Y. Yanagisawa, "On data compaction of scanning curves," *Comput. J.*, col. 32, no. 6, pp.563-566, 1989

[6] M.J. Weinberger, G.Seroussi, "The LOCO-I lossless image compression algorithm: Principles and standardization in to JPEG-LS", *IEEE Trans. Image Processing*, vol.9, no.8, pp 1309-1324, 2000

[7] N.D. Memon and K. Sayood "Lossless image compression- a comparative study", *Still image Compression, SPIE proc.*, vol. 2418, pp. 8-20