Least Squares-Adapted Edge-look-ahead Prediction with Run-Length Encodings for Lossless Compression of Images

Lih-Jen Kau and Yuan-Pei Lin
Dept. Elec. and Control Engr., National Chiao Tung Univ., Hsinchu, Taiwan, R.O.C.

***Abstract*— Many coding methods are more efficient with certain types of images than others. In particular, run-length coding is very useful for coding areas of little changes. Adaptive predictive coding achieves high coding efficiency for fast changing areas like edges. In this paper, we propose a switching coding scheme that will combine the advantages of both Run-length and Adaptive Linear Predictive coding (RALP) for lossless compression of images. For pixels in slowly varying areas, run-length coding is used; otherwise LS (least square)-adapted predictive coding is used. Instead of performing LS adaptation in a pixel-by-pixel manner, we adapt the predictor coefficients only when an edge is detected so that the computational complexity can be significantly reduced. For this, we propose an edge detector using only causal pixels. This way, the predictor can look ahead if the coding pixel is around an edge and initiate the LS adaptation in advance to prevent the occurrence of a large prediction error. With the proposed switching structure, very good prediction results can be obtained in both slowly varying areas and pixels around boundaries as we will see in the experiments.**

***Index Terms*: Lossless Image Coding, Edge Detection, Adaptive Prediction, Least-squares Optimization, Run-length Encodings**

I. Introduction

Lossless image coding is required by many applications, such as medical imaging, remote sensing, and image archiving. It has remained a challenge to source coding community for the difficulty of obtaining a high compression ratio. Therefore, many state-of-the-art approaches have been proposed [1]-[6]. Among which, the LOCO-I coder [1] has been standardized into JPEG-LS for its low complexity and effectiveness of removing statistical redundancy. To pursue higher coding gains, linear predictors adapted by LS optimization have been proposed recently to accommodate varying statistics of coding images [3]-[5]. Among LS approaches, the EDP [4] pointed out that the superiority of LS optimization is in its edge-directed property. To reduce the computational complexity, the EDP initiates the LS optimization process only when the prediction error is beyond a preselected threshold. The EDP has made a noticeable improvement on the bit rate performance over the state-of-the-art lossless coders [1][2].

In this paper, we propose a switching coding scheme that combines the advantages of both Run-length and Adaptive Linear Predictive coding (RALP). There are other switching methods that achieve very low bit rates [6]. However the results are usually obtained with a very high computational complexity [6]. On the contrary, the proposed RALP coder can achieve a very good coding efficiency but with a moderate computational complexity. In the proposed approach, the run-length encoder is used for pixels in slowly varying areas; otherwise an LS-based adaptive predictor, which has been shown to be very useful for the prediction of pixels around an edge [4][5], is used. Moreover, we adapt the predictor coefficients only when an edge is detected or when the prediction error is beyond a pre-selected threshold so that the computational cost can be significantly reduced [5]. To do this, we use a simple and efficient edge detector that uses only causal pixels. This way, the predictor can look ahead if the coding pixel is around an edge and initiate the LS adaptation process beforehand to prevent the occurrence of a large prediction error. With the proposed switching structure, very good prediction results can be obtained in both slowly varying areas and pixels around boundaries. As we will see in the experiments that the switching structure combined with edge-look-ahead prediction renders the proposed RALP highly adaptable and very feasible under limited resources. A very good trade-off between coding efficiency and computational complexity can be achieved. Some preliminary results regarding the proposed LS-based predictor with edge-look-ahead can be found in [5].

The rest of the paper is organized as follows. Section II gives an overview of the proposed RALP system. Experiments of the proposed method and comparisons to existing predictors and coders are given in section III. A conclusion is given in section IV.

II. Proposed RALP System

The proposed RALP system has two operation modes, run mode and regular mode. If the current pixel is in an local area of little changes, the run mode is triggered and the current pixel is encoded using run-length encoding. If not, the regular mode is assumed and the pixel is encoded using predictive coding.

A. Run mode

It is known that the run-length coding is most efficient for the encoding of consecutive pixels with identical grey values. The case that consecutive pixels are identical can usually occur in an artificial image or in slowly varying areas of a natural image. Therefore, we use the run-length coding for the encoding of pixels in an area of little changes. If the four pixels $x_n(1), \cdots, x_n(4)$ in Fig. 1 are identical, the run mode is switched on and the run-length is encoded using an arithmetic coder with an alphabet set of $\{0, 1, \cdots, 20\}$. In run mode, if the four pixels $x_n(1), \cdots, x_n(4)$ are not of the same value, an '0'(escape symbol) is encoded to indicate an unsuccessful run

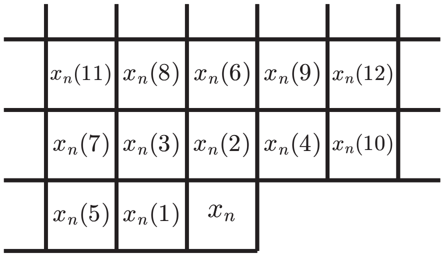


Fig. 1. The ordering of pixels for prediction inputs.

and regular mode is used for encoding the current pixel. The regular mode is assumed for the first pixel of every line. The encoding of an escape symbol can cause penalty and degrade the coding efficiency. Therefore, we record the number of times of run mode triggered and the number of unsuccessful runs. If the percentage of unsuccessful run is greater than a predefined threshold, the run mode is disabled and not to be used for the rest of the coding process. We notice that all the pixels used for mode selection are causal and the decoder knows when to switch modes without any side information.

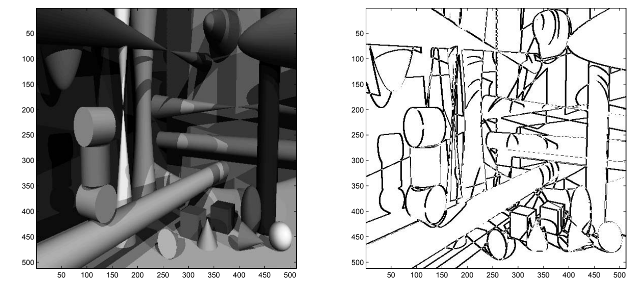
B. Regular mode

In the regular mode, pixels are encoded using predictive coding. Predictive coding can be very efficient for the removal of statistical redundancy between neighboring pixels in slowly varying areas. However there can have a large prediction error around boundaries. We will use LS optimization to update the predictor coefficients on the fly so that the predictor can adapt to the varying statistics. It is known that the LS-based adaptive predictor can improve the prediction result around boundaries for its edge-directed property [4][5]. However a pixel-by-pixel adaptation of the predictor coefficients is computationally expensive and often not necessary. Therefore, we will initiate adaptation only when the prediction is inadequate, which is around an edge. For this, an edge detector is used to look ahead and determine if the coding pixel is around an edge so that the predictor can adapt beforehand to prevent the occurrence of a large prediction error. It should be noted that conventional edge detectors, e.g., “Sobel” operator, can not be applied here because they use non-causal pixels, i.e., pixels yet to be encoded.

We observe that an area that contains an edge usually has a large variance. Furthermore, the histogram of such an area tends to have two peaks, one on each side of the mean value. We will use these two observations to determine the existence of an edge. Moreover, the set of the four pixels $\kappa = \{x_n(1), \cdots, x_n(4)\}$ in Fig. 1 are used for the detection. The mean $\bar{x}$ and variance σ^2 of the set κ are calculated. Furthermore, the four pixels can be divided into two groups, the pixels with gray levels higher than $\bar{x}$ in group κ_h and the rest in group κ_l. We also compute the respective variance σ_h^2, σ_l^2 of the pixels in κ_h and κ_l.

A pixel around an edge is likely to have a large σ^2 but small σ_h^2 and σ_l^2. We determine whether the coding pixel is around an edge if the following two conditions are both satisfied,

$$\sigma^2 \geq \gamma_1, \quad \text{and} \quad \sigma^2 \geq \gamma_2\,(\sigma_h^2 + \sigma_l^2). \tag{1}$$



(a) (b)

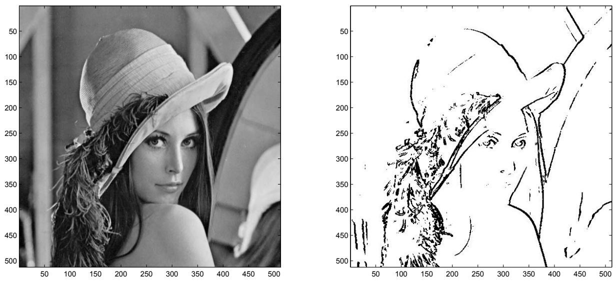
Fig. 2. (a) The image “Shapes”. (b) Pixels for which (1) is satisfied in the image “Shapes”.

The second condition in (1), which represents a histogram with two peaks, is included because a region with uniformly distributed gray values also results in a large σ^2. Therefore, the switch first examines if σ^2 is large then checks the second inequality. The LS adaptation process in the regular mode is activated whenever the two conditions in (1) are satisfied. We have found through experiments that $\gamma_1 = 100$ and $\gamma_2 = 10$ work very well and these values will be used throughout the paper. It should also be noted that the run mode will be triggered when $\sigma^2 = 0$, i.e., the case that $x_n(1), \cdots, x_n(4)$ are identical. In this case, we do not have to check the conditions in (1). As we will see later in experiments that the proposed detector is very effective in detecting edges although only four pixels are used. Moreover, since we use only causal pixels for the detection of an edge, the decoder can perform the same edge detection operation and switches on the LS adaptation process.

In the regular mode, the LS adaptation process is activated whenever the two conditions in (1) are both satisfied or when the prediction error is greater than a predefined threshold θ. It is noted that the decoder also knows when to activate the LS adaptation process; no additional side information needs to be transmitted. The corresponding predictor inputs for different prediction orders are shown in Fig. 1 where the ordering of pixels is based on the distance to the pixel to be encoded. Furthermore, we do not use any training set for the optimization of initial predictor coefficients. The initial coefficients for the proposed predictor are equally weighted, i.e., the coefficients for the Nth-order predictor are $1/N$ respectively. The prediction error is further refined through an error modeling mechanism and then entropy encoded using conditional arithmetic coding to produce the coded bit stream.

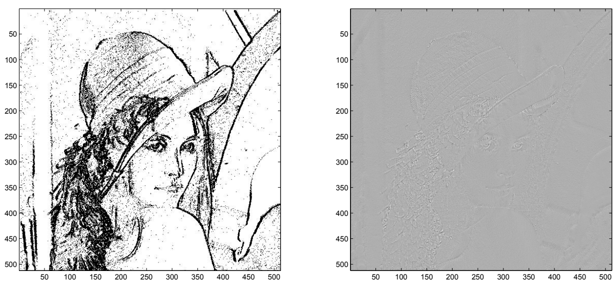
III. Experiments

In this section, we evaluate the performance of the proposed lossless image codec. Comparisons to existing linear and non-linear coders are also given. All the test images used in the experiments are from the website of TMW [6]. We first demonstrate the usefulness of the proposed edge detector and then the effectiveness of the proposed edge-look-ahead mechanism in the regular mode. Finally, the bit rate performance of the system is presented.



(a) (b)

Fig. 3. (a) The image “Lennagrey”. (b) Pixels for which (1) is satisfied in the image “Lennagrey”.



(a) (b)

Fig. 4. (a) Pixels for which LS adaption is used in the proposed edge-look-ahead predictor for the image “Lennagrey”. (using a sixth-order predictor with $\gamma_1 = 100$, $\gamma_2 = 10$) (b) Image of uncompensated prediction errors using the proposed edge-look-ahead approach for “Lennagrey”.

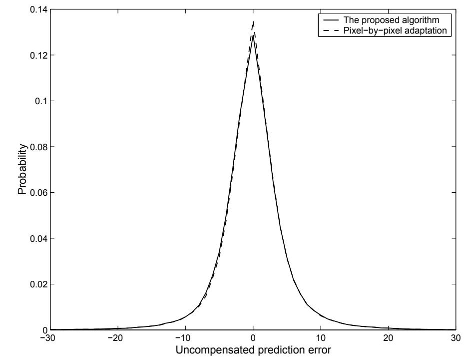


Fig. 5. Histogram of uncompensated prediction errors for the proposed approach and that of a pixel-by-pixel adaptation. (both using a sixth-order predictor)

TABLE I

COMPRESSION RATIO AND THE RUNNING TIME (IN SECONDS, ON A PIII-1.06GHZ MACHINE) OF THE CONSTRUCTED CODER VARY WITH DIFFERENT PREDICTION ORDER USING THE PROPOSED APPROACH.

Image	N=4		N=6		N=8		N=10	
	Compression Ratio	Run Time	Compression Ratio	Run Time	Compression Ratio	Run Time	Compression Ratio	Run Time
Baboon	1.35	3.82	1.38	6.80	1.38	12.37	1.38	14.21
Lena	1.82	3.09	1.84	4.22	1.84	5.71	1.84	6.95
Lennagrey	2.01	3.04	2.03	3.79	2.03	5.18	2.03	5.84
Peppers	1.86	3.03	1.88	3.86	1.88	4.59	1.89	6.06
Barb	1.88	3.38	1.95	4.80	1.96	9.07	1.97	8.62
Barb2	1.73	5.41	1.77	8.28	1.77	14.33	1.77	15.26
Boats	2.11	4.82	2.15	6.02	2.16	8.43	2.17	9.18
Gold Hill	1.82	4.76	1.83	6.58	1.84	8.30	1.84	10.97
Average	1.82	3.92	1.85	5.54	1.86	8.50	1.86	9.64

A. The Edge Detector

To demonstrate the effectiveness of the proposed edge detector, we use the image “Shapes” (Fig. 2(a)), which is an artificial image with many edges and lines. The pixels that satisfy the two conditions in (1) are marked in Fig. 2(b). We can see from Fig. 2(b) that the edge detector has successfully picked out the pixels around edges. In addition to artificial images, we also apply the edge detector to “Lennagrey”, a natural image that is shown in Fig. 3(a). As can be seen in Fig. 3(b), the pixels around the edges have been picked out successfully.

B. Effectiveness of the Edge-look-ahead mechanism

The usefulness of the proposed predictor with edge-look-ahead can be demonstrated through the following experiment. As we will see later that a sixth-order predictor is effective in removing the statistical redundancy, we construct two sixth-order LS based predictors for the regular mode; one with the use of the proposed edge-look-ahead mechanism and the other performs LS adaptation in a pixel-by-pixel manner. Then we compare the performance of the two predictors. In this experiment, the run mode is also enabled and the image “Lennagrey” in Fig. 3(a) is used for this comparison.

For the predictor with edge-look-ahead, the pixels for which LS adaption is activated are shown in Fig. 4(a). Overall, about 17% of pixels activate the LS adaptation process. The image of uncompensated prediction errors and the corresponding histogram are shown in Fig. 4(b) and Fig. 5 respectively. For comparison, we also show in Fig. 5 the histogram of uncompensated prediction error when the LS adaptation process is performed in a pixel-by-pixel manner. The histogram using the proposed approach is very close to that with pixel-by-pixel adaptation although only 17% of pixels activate the LS adaptation process. The proposed edge-look-ahead approach has made a good tradeoff between prediction efficiency and computational complexity. Indeed, the entropies corresponding to the two histograms in Fig. 5 are respectively 4.20 bits (proposed approach) and 4.18 bits (adapted in a pixel-by-pixel manner).

C. Order of predictor

The order of the predictor affects coding gain in the regular mode. We list in Table I the compression ratio for order $N = 4, 6, 8, 10$ with the run mode disabled. The execution time is also listed in the Table. As can be seen in Table I, the compression ratio quickly saturates when the prediction order is greater than six. Moreover, the increases in the execution time does not justify the use of prediction order higher than six. Therefore, the use of a sixth-order predictor in the regular mode is a proper choice and this will be used in the design of a lossless image coder afterward for comparison with existing state-of-the-art lossless image coders.

TABLE II

PERCENTAGE OF PIXELS PERFORMING LS ADAPTION AND THE RESULTING FIRST-ORDER ENTROPY BY VARYING THE VARIANCE THRESHOLD γ_1 IN THE PROPOSED APPROACH (THE IMAGE "LENNAGREY" IS USED FOR THE TEST).

Parameter	Number of Pixels with LS Adaptation	Percentage with LS Adaptation (A)	Percentage Detected as around an Edge (B)	Percentage with LS Adaptation in slowly varying areas (C=A-B)	First-order Entropy
γ_1=100	37692	14.38	11.47	2.91	4.176
γ_1=200	28205	10.76	6.87	3.89	4.184
γ_1=300	24117	9.20	4.79	4.41	4.188
γ_1=400	21819	8.32	3.58	4.75	4.190
γ_1=500	20342	7.76	2.75	5.01	4.193

TABLE III

COMPARISONS WITH EXISTING LOSSLESS IMAGE CODERS (IN BITS/SAMPLE). THE FIFTH COLUMN IS THE EXECUTION TIME OF THE PROPOSED APPROACH ON A PIII-1.06GHZ MACHINE.

Image	First-oreder Entropy	RALP	% of LS adaptation	seconds	JPEG-LS [1]	CALIC [2]	EDP* [4]	TMW [6]
Airplane	3.82	3.71	16.6	3.75	3.82	3.74	N/A	3.60
Baboon	5.91	5.81	63.8	6.82	6.04	5.88	5.81	5.73
Balloon	2.56	2.55	4.0	4.50	2.90	2.83	N/A	2.66
Barb	4.15	4.12	33.6	4.80	4.69	4.32	4.11	4.09
Barb2	4.57	4.51	37.7	8.02	4.69	4.53	4.52	4.38
Boats	3.84	3.75	18.1	6.07	3.93	3.83	3.80	3.61
Camera	4.42	4.24	26.7	1.18	4.31	4.19	N/A	4.10
Couple	3.63	3.63	15.8	1.01	3.70	3.61	N/A	3.45
Gold Hill	4.41	4.32	23.6	6.81	4.48	4.39	4.39	4.27
Lena	4.36	4.35	23.5	4.27	4.61	4.48	4.40	4.30
Lennagrey	3.97	3.95	17.4	3.82	4.24	4.11	4.02	3.91
Noisesquare	5.35	5.37	56.7	1.60	5.68	5.44	N/A	5.54
Peppers	4.27	4.27	18.2	3.97	4.51	4.42	4.35	4.25
Shapes	1.87	1.52	7.1	1.94	1.21	1.14	N/A	0.76
Average	4.08	4.01	25.92	4.18	4.20	4.07	4.43	3.90

D. LS adaptation

In this subsection, we investigate how the prediction performance (entropy) varies with the variance threshold γ_1 in (1) for LS adaptation. We construct a tenth-order predictor with edge-look-ahead for the experiment and the image "Lennagrey" (Fig. 3(a)) is used for the test. For LS adaptation, we use the same training area as defined in EDP [4]. Moreover, we set $\gamma_2 = 10$ and $\theta = 10$ for all cases in the experiment. By varying the variance threshold γ_1, the number of pixels that activates LS adaptation also changes. The experimental results using the proposed approach with various γ_1 are shown in Table II.

We observe from Table II that a small γ_1 may result in a small entropy at the expense of an increased number of pixels performing LS adaptation process. The percentage of pixels regarded as around an edge increases as γ_1 decreases, but the percentage of pixels that activates the LS adaptation in slowly varying areas almost remain unchanged. This can be best observed from the fifth column, which is obtained by subtracting the fourth column from the third column, of Table II, and we can find that about 3% to 5% of pixels in slowly varying areas will activates the LS adaptation. Therefore, the improvement on the entropy in the proposed approach is mainly around edges.

E. Comparisons to existing state-of-the-art coders

Table III gives the actual bit rates of proposed RALP coder, JPEG-LS [1], CALIC [2], EDP [4] and TMW [6] for a set of fourteen test images. All the bit rates of the proposed algorithm are obtained using the same parameters described in previous sections and no individual optimization is performed. Besides, we show in the second column and the fourth column respectively the first-order entropies of the compensated prediction errors and the percentage of pixels performing LS adaptation using the proposed approach. We have also shown in the fifth column the execution time of the proposed coder so that we can get a picture on the runtime performance of the proposed approach. It should be noted that some of the results in EDP are denoted by "N/A" because they are not reported in the paper of EDP [4]. Table III shows that RALP has lower bit rates than JPEG-LS in thirteen out of the fourteen test images and outperforms CALIC [2] in eleven of fourteen test images. Encouragingly, the proposed RALP achieves lower bit rates than the highly complex TMW in two images, "Balloon" and "Noise square".

IV. CONCLUSION

In this paper, a switching coding scheme that combines the advantages of both run-length and adaptive linear predictive coding is proposed. We adapt the predictor coefficients in regular mode only when an edge is detected or when the prediction error is large to reduce complexity. For this, we propose a simple yet effective causal edge detector to look ahead so that the occurrence of a large prediction error can be avoided. When compared with the pixel-by-pixel LS adaptation, the proposed approach can achieve a noticeable reduction in complexity with only a minor degradation in entropy. A good tradeoff between computational complexity and prediction results has been obtained and the proposed approach is very feasible under limited resources. With the proposed switching structure, very good results can be obtained in slowly varying areas and edge pixels as shown in our experiments. Furthermore, comparisons to existing state-of-the-art lossless image predictors and coders have demonstrated the superiority of the proposed system.

REFERENCES

[1] 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.

[2] X. Wu, and N. Memon, "Context-based, adaptive, lossless image coding," *IEEE Trans. Communications*, Vol. 45, No. 4, pp. 437-444, April 1997.

[3] N. Kuroki, T. Nomura, M. Tomita and K. Hirano, "Lossless image compression by two-dimensional linear prediction with variable coefficients," *IEICE Trans. Fundamentals*, Vol. E75-A, No. 7, pp. 882-889, July 1992.

[4] Xin Li and Michael T. Orchard, "Edge-directed prediction for lossless compression of natural images," *IEEE Trans. Image Process.*, Vol. 10, No. 6, pp. 813-817, June 2001.

[5] Lih-Jen Kau and Yuan-Pei Lin, "Adaptive lossless image coding using least squares optimization with edge-look-ahead," *IEEE Trans. Circuits and Systems II*, Vol. 52, No. 11, pp. 751-755, Nov. 2005.

[6] B. Meyer, and P. E. Tischer, "TMW-a new method for lossless image compression," in *Proc. Int. Picture Coding Symp.*, Berlin, Germany, Oct. 1997.