# LOSSLESS COMPRESSION OF NATURAL IMAGES USING REGRESSIVE ANALYSIS

Konstantin Balashov David Akopian Jaakko Astola

Signal Processing Laboratory
Tampere University of Technology
P.O. 553, SF-33101 Tampere, Finland

## ABSTRACT

This paper develops new algorithms belonging to the class of context modeling methods, with direct application to lossless coding of gray level images. The prediction stage and the context modeling stage are performed using one or more linear predictors based on the regressive analysis.

## 1 INTRODUCTION

Lossless image compression algorithms are divided into “sequential” like FELICS [1-4], LOCO-I [6, 7], CALIC [8, 9], some new context-based algorithms [10, 11] and transform-based ones like lossless SPIHT [12].

Sequential algorithms scan image sequentially from left to right, from top to bottom (fig.1). These algorithms predict the current pixel (“predicting”), subtract actual pixel value from the estimated one, estimate the quality of prediction (“error modeling”), then encode difference between actual and predicted values according to the error model (“coding”).

Transform based algorithms apply some transforms (as a rule, separate low-bound from high-bound fraction of an image signal) and then encode transformed image. Transform based algorithms provide very good results for lossy image compression. Unexpectedly, they are significantly worse in lossless compression, than sequential ones, probably, because of transform-based lossless compression algorithms are still not developed enough.

Anyway, the aim of this paper is lossless sequential algorithms.

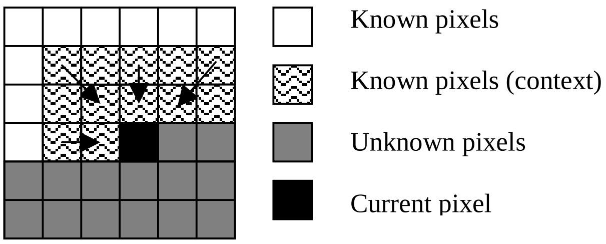


Fig. 1. Prediction of the next pixel using 12 known neighbor pixels

Both predicting and error modeling are based on some models of dependence of predicting pixel from the context. It could be heuristic estimations, DPCM predictors, L-predictors, boolean predictors and so on.

### 1.1 Regressive Analysis Based Predictor

The advantage of utilizing of regressive analysis based predictor is exact prediction, which is better, than any linear heuristic predictions.

The problem is that the predictor is linear, and dependence between the neighboring pixels and current one is not. However, this dependence is closed to linear in case of smoothness, vertical or horizontal edginess of the neighborhood. So, 3 or more linear predictors could provide a highly precise prediction.

### 1.2 Regressive Analysis

Regressive analysis converts data stream into equation coefficients. For N channels of input data it calculates (N+1)*(N+2) coefficients then solves system of N equations.

In the algorithm, each channel of the input data contains values of a neighboring pixel position for predicting pass or distribution values of prediction error for error modeling pass.

Encoder embeds calculated equation coefficients into the compressed file.

Decoder substitutes coefficients and actual neighboring pixel values into equations and calculates estimation of value of the next pixel.

## 2 ALGORITHM DESCRIPTION

Actually, encoding consists of:

1. Gathering information for prediction (pass 1).
2. Calculating regressive coefficients for prediction.
3. Gathering information for error modeling – predicting and estimating. errors (pass 2).
4. Calculating regressive coefficients for error modeling.
5. Predicting, calculating errors and encoding (pass 3).
6. Storing regressive coefficients together with the image.

Decoding consists of:

1. Reading regressive coefficients for prediction and error modeling.
2. Predicting, calculating error and decoding.

So, decoding is very simple and fast.

Regressive coefficients are 4 bytes long (floating point) and takes 2*(N+1)*4 bytes in compressed file (104 bytes for N=12).

### 2.1 Predicting

As described above, in a special pass encoder calculates regression coefficients and saves them to a compressed file. Decoder reads them from the file.

When regressive coefficients are known, predicting for the both encoder and decoder becomes very simple. Algorithm just puts regressive coefficients and actual neighboring pixel values into equation and calculates the result – predicted value of the current pixel.

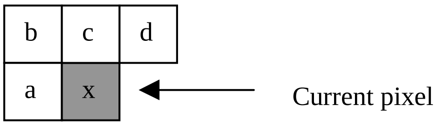


Fig. 2. Causal template for predictor choosing

Algorithm has been tested with up to 7 predictors. Predictor selection algorithm is context-depended, fixed and heuristic. The choice is based on analysis of 4 neighboring pixels (Fig. 2). We compare pixel "b" with "a" and "c" and compare the differences. If difference between "a" and "b" is less, than between "b" and "c", we recognize it as a vertical edge and as a horizontal one otherwise. Sum of differences is recognized as a "sharpness" of neighboring and used in predictor choosing as well.

### 2.2 Error Modeling

Error modeling is similar to predicting. Algorithm just estimates error instead of pixel values. Then error value is used for choosing distribution coefficient. Algorithm just chooses one of 32 pre-calculated distributions with different coefficients.

Probabilities of errors have near-to-Normal distribution for high and medium dispersion and near-to-Laplace distribution for low dispersion [2].

The algorithm is based on Normal distribution model. Dynamic distribution adaptation improves the compression ratio by 1-2%.

Predictor selection is similar to one used in pixel prediction, just edge detecting has no sense here.

### 2.3 Coding

Algorithm uses standard arithmetical coding. It provides optimal encoding and high compression ratio. Algorithm is not oriented on certain coding scheme and could be used with any other coders as well.

### 2.4 Parameters and properties

Algorithm has a number of parameters. There are a number of neighboring pixels used in regressive analysis, number of predictors used in predicting and error modeling and so on. All of them influence the compression ratio. For example, optimal number of neighboring pixels is 12. Optimal number of predictors is 6-7 for pixel predicting and 2-3 for error modeling. Unexpectedly, some advantages like prediction correction using error prediction (used in CALIC [8, 9]) or multi resolution approach (used in MRU [3]) don't improve compression ratio of the algorithm.

## 3 COMPARATIVE RESULTS

Algorithm has been tested on some test images for JPEG test image set. All of them are 720x576, grayscale, 8 bits per pixel.

| Image | RA 1-1 | RA 7-3 | ASBOSC | FSM-L | CALIC | UCM | Sunnet | LOCO-I | SPIHT |
|---|---|---|---|---|---|---|---|---|---|
| boats | 4.056 | 3.803 | 3.762 | 3.781 | 3.834 | 3.85 | 4.01 | 3.92 | 4.059 |
| girl | 3.889 | 3.744 | 3.709 | 3.740 | 3.767 | 3.81 | 3.91 | 3.90 | 3.989 |
| gold | 4.484 | 4.412 | 4.377 | 4.382 | 4.394 | 4.45 | 4.56 | 4.47 | 4.594 |
| zelda | 3.828 | 3.676 | 3.699 | 3.717 | 3.750 | 3.80 | 3.81 | 3.87 | 3.865 |

Table 1. Compression results of JPEG images in bit/pixel.

RA 1-1 and RA 7-3 are variants of proposed algorithm. "1-1" means, that algorithm uses 1 predictor for pixel predicting and 1 for error modeling. This variant of algorithm is quite linear.

"7-3" means 7 predictors for pixel predicting and 3 for error modeling.

ASBOSC [10] is binary predictor based context modeling algorithm by Ioan Tabus and Jaako Astola.

FSM-L [11] is L-predictor based context modeling algorithm by Ioan Tabus and Jaako Astola.

CALIC [8, 9] is CALIC algorithm by Xiaolin Wu.

LOCO-I [6, 7] is LOCO-I algorithm by Marcelo J. Weinberger, Gadiel Seroussi, and Guillermo Sapiro.

SPIHT [12] is a Lossless SPIHT by Amir Said and William A. Pearlman.

Linear prediction is not effective, but combination of linear predictors provides high compression performance.

## Conclusions

Usage of the regressive analysis in lossless sequential image compression is highly effective. It provides good results for both pixel values prediction and error modeling.

The bottleneck of proposed algorithm is a heuristic approach for predictor selection. Algorithm requires more precise approach to improve the compression ratio. Also,

Proposed usage of combination of linear predictors could be used in a related image processing areas like filtering and image enhancement.

## References


[1] P. G. Howard and J. S. Vitter, "Fast and efficient lossless image compression," Proc. of Data Compression Conference, pp. 351-360, 1993.

[2] P.G. Howard and J.S. Vitter, "Error Modeling for Hierarchical Lossless Image Compression," Proceedings of the 1992 IEEE Data Compression Conference (DCC '92), Snowbird, UT, pp.269-278, March 1992.

[3] P. G. Howard and J. S. Vitter, "Fast Progressive Lossless Image Compression," Image and Video Compression Conference, Symposium on Electronic Imaging: Science & Technology SPIE-2186, San Jose, California, pp.98-109, Feb. 9-10, 1994.

[4] P.G. Howard and J.S. Vitter, "Parallel Lossless Image Compression Using Huffman and Arithmetic Coding," A shortened version appears in Proceedings of the 1992 IEEE Data Compression Conference (DCC '92), Snowbird, UT, March 1992, pp.299-308.

[5] M.J. Slyz and D.L. Neuhoff, "A Nonlinear VQ-based Predictive Lossless Image Coder," IEEE Computer Society Press, pp. 304-310, March 1994.

[6] M. J. Weinberger, G. Seroussi, and G. Shapiro, "LOCO-I: A low complexity, context-based, lossless image compression algorithm," in Proc. 1996 Data Compression Conference, (Snowbird, Utah, USA), pp. 140-149, Mar. 1996.

[7] M. J. Weinberger, G. Seroussi, "The LOCO-I lossless image compression algorithm: Principles and standardization into JPEG-LS, HPL-98-193," November 1998.

[8] Xiaolin Wu, "Lossless Compression of Continuous-tone Images via Context Selection, Quantization, and Modeling", *IEEE Trans. on Image Processing*, vol. 6, no. 5, p. 656-664, May 1997.

[9] Xiaolin Wu: An Algorithmic Study on Lossless Image Compression. Data Compression Conference 1996:150-159

[10] I. Tabus , J. Rissanen , and J. Astola ., "Adaptive L-predictors based on finite state machine context selection," In Proc. ICIP'97 International Conference on Image Processing, pp. 401-404, Santa Barbara, California, Oct. 1997.

[11] I. Tabus and J. Astola, "Adaptive Boolean predictive modelling with application to lossless image coding," In SPIE - Statistical and Stochastic Methods for Image Processing II, pp. 234-245, San Diego, California, Jul. 1997.

[12] A. Said and W.A. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees," IEEE Trans. On Circuits and Systems for Video Technology, pp. 234-250, June 1996.