Adaptive Prediction for Lossless Image Compression

Sangho Park*
*Andong National University, spark@andong.ac.kr

***Abstract* -.** Genetic algorithm based predictor for lossless image compression is propsed. We describe a genetic algorithm to learn predictive model for lossless image compression. The error image can be further compressed using entropy coding such as Huffman coding or arithmetic coding. We show that the proposed algorithm can be feasible to lossless image compression algorithm.

Keywords: Lossless compression, predictive coding,.data compression, image coding, genetic algorithm.

1 Introduction

Data compression is the technology of representing information in a compact form. It is one of most important technology for the multimedia era. Lossless compression involves no loss of information. The compressed data can be recovered exactly same as the original data if data have been losslessly compressed. Lossless image compression has been studied by many researchers because it is applied to much real-world application. Most lossless image compression methods in the literature consist of four components[1]. A selector to choose the next pixel to be encoded, a predictor to estimate the intensity of the pixel, an error modeler to estimate the distribution of the prediction error, and a statistical coder to code the prediction error using the error distribution. State of the art lossless image compression to achieve high compression ratio is predictive coding[2].

Predictive coding is an image compression scheme. It uses a compact model of a data to predict pixel values of data based on the values of neighboring pixels. Various techniques have been suggested for predictive lossless compression. In [3], the concept of prediction was shown to be useful in providing additional modeling features to the context algorithm. State of the art lossless image compression is suggested in the literature: CALIC(Context adaptive lossless image compression) algorithm[4] and LOCO-I[5].This paper introduces genetic algorithm based predictor for lossless image compression. Image compression scheme based on the predictive prediction is illustrated in Fig. 1.

The paper is organized as follows: In section 2, we discuss the genetic algorithm. In section 3, adaptive prediction based on genetic algorithm is proposed for lossless image compression. Section 4 presents the evaluation of our algorithm, and section 6 concludes our discussion.

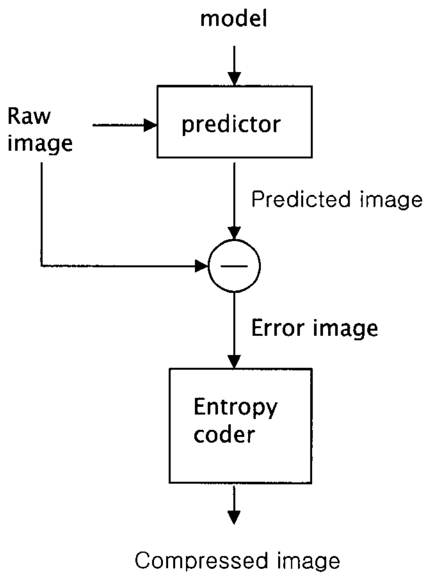

Fig. 1. Image compression system based on predictive coding

2 Genetic Algorithm

Genetic algorithm[6] is iterative procedure that maintains a population during iterations. GA can find the optimal solution for a particular problem by seeking the maximum or minimum of the appropriate fitness function. A population consists of a number of strings which represent possible candidate solutions. At each iteration a new population is created from the previous population using a set of genetic operations. The basic procedure of a typical genetic algorithm is depicted in Fig. 2. During each iteration *t*, called generation, strings in the current population *P(t)* are evaluated on the basis of their values from the fitness function and have probability of selection for next generation. This iteration process of selecting new strings is called reproduction. To generate a new population for the next generation, usually two selected strings are recombined by specific genetic operators such as crossover and mutation. This procedure would continuously generate new population until a termination condition is reached. After termination of the iteration, the best string in the final population is chosen as the solution. A genetic algorithm may be terminated by determining the maximum number of iteration or after finding an acceptable solution.

```
Set initial iteration t = 0
Initialize P(t)
Evaluate P(t)
while (termination condition not satisfied)
do
begin
    t = t+1
    Generate P(t+1) from P(t)
    Recombine P(t+1)
    Evaluate P(t+1)
end
```

Fig. 2. Procedure of genetic algorithm

3 Adaptive Predictor

Input image data is read by processing its pixels in raster-scan order. Each pixel value *x(i,j)* is predicted by a weighted sum of its six neighbors. Pixel's neighbors are context of pixel or prediction mask. The prediction mask for pixel *x(i,j)* is depicted in Fig. 3. Predictive pixel value of pixel *x(i,j)* is

$$\bar{x}(i,j) = INT(w_0 x(i-2,j) + w_1 x(i-1,j) + w_2 x(i-1,j-1) + w_3 x(i,j-1) + w_4 x(i+1,j-1) + w_5 x(i,j-2))$$

Predictor's weights are determined using genetic algorithm to take into account local features of the image.

Prediction error *ERR(i,j)* is the difference of pixel value and predicted pixel value.

$$ERR(i,j) = \bar{x}(i,j) - x(i,j)$$

The *ERR(i,j)* is further encoded by entropy coding to compress image data. Encoded data are sent to receiver. The decoder reconstruct input image only use predicted error *ERR(i,j)* . Predictive coding scheme is illustrated in Fig. 4. where Model(i,j) is a function returns a predicted value of pixel from pixel value.

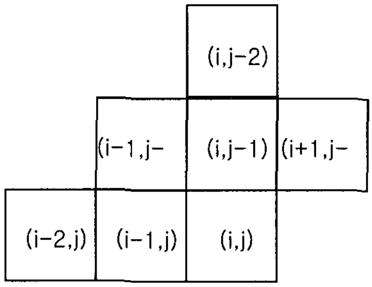

Fig . 3. Six pixels prediction mask used in prediction

```
ENCODER(Model, Image)
for i = 0 to imax
    for j = 0 to jmax
        ERR(i,j) = Image(i,j) - Model(i,j)

DECODER(Model)
for i = 0 to imax
    for j = 0 to jmax
        Image(i,j)   =   Model(i,j)   +
ERR(i,j)
```

Fig. 4. Predictive coding algorithm

3.1 The structure of strings

There are *P* strings in the population and each string consists of *Q* weights. Each string represents six weights where the range of weights is 0 to 1. A population is constructed by a two-dimensional array of size *P*x*Q*. The strings of artificial genetic systems are analogous to chromosomes in biological systems.

3.2 Fitness function

The fitness function, which estimates the goodness of the string, consists of the probability of selecting a string among strings in the population, and the probability of selecting a particular string among strings. The fitness function or objective function for string id defined by

$$f_i = x(i,j) - INT(w_0 x(i-2,j) + w_1 x(i-1,j) + w_2 x(i-1,j-1) + w_3 x(i,j-1) + w_4 x(i+1,j-1) + w_5 x(i,j-2))$$

The total objective function of a population is equivalent to the sum of the object function for string.

$$f_{total} = \sum_{i=0}^{P-1} f_i$$

The problem to find optimal weight is formulated to energy minimization problem. Weight which has smallest fitness is selected as weights for prediction.

3.3 Genetic Operators

There are our algorithm is composed of four operations: reproduction, crossover, and mutation. Two genetic operators, crossover and mutation are developed in order to find the globally optimal weights for adaptive predictive coding. In the reproduction process, individual strings are selected according to the value of their fitness function. After selection of strings, strings are recombined using genetic operators ion order to generate new strings. Strings with high fitness values have higher probabilities of selection: therefore these strings produce more offsprings in the reproduction process than the strings with lower fitness values.

A pair of mated strings called parent strings produce two tentative strings called offstrings under crossover operation. Crossover operation is only allowed to be performed between a pair of substrings. First a crossover point along the string is randomly selected within the range of 1 and Q-1. Two new strings are created by exchanging all the valued between the position of crossover point and Q-1. This crossover operation is applied to all the strings of the parent strings.

Although the reproduction process and the crossover operators will search the solution space effectively, occasionally they may lose some useful solution patterns. Mutation operator will protect against such an irrecoverable loss and will avoid the algorithm to get trapped into a local minimum and will enable it to jump to the global minimum. Mutation is a process of finding a new search space by changing the value of a randomly chosen position within a string which is chosen at random. The jth element of a string can be changed to any real value between 0 and 1.

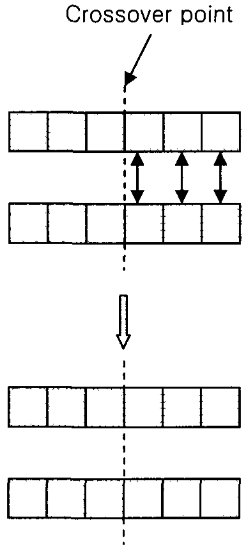

Fig. 5. Crossover operation

4 Experimental Results

Input image data is read by processing its pixels in raster-scan order. An initial population is randomly generated using random numbers. The fitness value for each string in the initial population is evaluated by fitness function. Genetic operators such as crossover, mutation are applied to selected strings to generate offsprings for the next generation. Generation of a new population under genetic operation continues until termination condition is satisfied. After termination the string which has the highest fitness is selected as the best solution. In genetic algorithm there are a number of parameters such as the number of strings in the population, the number of generations(termination condition), the probability of crossover, the probability of mutation, the typical values used parameters for experiments are 20, 30, 0.8, and 0.03, respectively.

In proposed system we need two steps to find out weights. And only one set of weights are selected and used for all pixels. The current JPEG lossless still picture compression standard[7,8] provides eight different prediction parameters. The user can select one parameter and try to test to find out best parameter for given image. First we find weights for every pixels using proposed genetic algorithm. Next image is divided by 8by 8 subgroups. Centroid of 64 weights set is calculated for all subgroups in image. Centroids are divided by 8 by 8 subgroups. And calculate centroid of 64 centroids. This process is continued until one centroid is finding. Final centroid is used as predictor's weights. Experiments were performed on 3 images of 512 by 512 pixels. Those are Lena, Pepper, and Airplane. These test sets are widely used for comparisons in the image compression algorithms. All images were gray-level digitized with a resolution of 8 bits/pixel. After compression, average bit per pixel for test images are similar to previously reported algorithm such as CALIC and LOCO-I.

5 Conclusions

In this paper, we proposed a prediction scheme for image compression. To find nearly optimal solution genetic algorithm is adopted. A representational structure for the string is proposed and some genetic operators such as crossover and mutation are developed in order to generate good population during the generation process and to find nearly optimal solution. An appropriate fitness function, which represents all the constraints imposed by the prediction process was formulated in order to select the best parents and thus to produce good offsprings. The results of our experiments are encouraging.

References

[1] P. G. Howard and J. S. Vitter, "New methods for lossless image compression using arithmetic coding," Information Processing and Management, vol. 28, pp. 765-779, 1992.

[2] N. Memon and X. Wu, "Recent progress in lossless image coding," *The Computer Journal*, 40: pp. 127-136, 1997.

[3] M. Weinberger, J. Rissanen, andR. Arps, "Applications of universal context modeling to lossless compression of gray-scale images," *IEEE Trans. Image Processing*, IP-5, pp. 575-586, Apr. 1996.

[4] X. Wu and N. Memon, "Context-based, adaptive, lossless image codec," *IEEE Trans., Commun.*, vol. 45, Apr. 1997.

[5] M. Weinberger, G. Seroussi, G. Sapiro, "LOCO-I: A Low Complexity, Context-Based, Lossless Image Compression Algorithm," *Proc. IEEE Data Compression Conference*, Snowbird, Utah, March-April 1996.

[6] D. E. Goldberg, Genetic Algorithm in Search, Optimization, and Machine Learning, Addison-Wesley, 1989.

[7] G. K. Wallace, "The JPEG still picture compression standard," *Communications of the ACM*, 34:31-44, April 1991.

[8] Khalid Sayood, *Introduction to Data Compression*, 2nd edition, Academic press, 2000.