Evolutionary Computation using Reinforced Learning on Image Compression

Sadaf Sajjad[a], Sajjad Mohsin[b]
COMSATS Institute of Information Technology
Department of
[a]Management Sciences, [b]Computer Science
Islamabad
PAKISTAN

Abstract: The present work is based on using the reinforced learning technique on evolutionary computation for compressing an image. We have reinforced the best code vectors for a codebook in vector quantization, and kept them alive for the next generation until we got the best image. A genetic algorithm model was designed using reinforced learning with the pairwise nearest neighbor approach (PNN). The proposed method was evaluated with respect to quality of compressed image by calculating Peak Signal to Noise Ratio (PSNR). We used Lena image for our experiments and the PSNR value for proposed method is 34.6.

Key–Words: Reinforced Learning, Evolutionary Computation, Genetic Algorithms, Image Compression

1 Introduction

The concept of Reinforcement theories of learning are taken from the field of Psychology and the first one to explain the concept was Edward Thorndike. This idea explains that actions followed by positive or negative consequences have their tendency to be reselected altered accordingly. Reinforced learning is an effective way of solving complex problems. Research literature supports that reinforcement learning methods are iterative and they gradually reach the correct answer through successive approximation [1].

Evolutionary Computation has been an important research topic for many years. The boom in the evolutionary computing in general and GA in particular is due to the work of John Koza[2].

There are many kinds of compression techniques. Lots of research literature is available which explains the importance and techniques of Image compression. Recently this field has gained an immense attention of scientists and researchers[3].

In 1980, Linde, Buzo, and Gray proposed a VQ design algorithm based on a training sequence. The use of a training sequence bypasses the need for multi-dimensional integration. A VQ that is designed using this algorithm are referred to in the literature as Generalized Lloyd Algorithm (GLA)[4]. Recently Vector Quantization is considered to be a most popular technique for image Compression [5]. GA has been successfully applied to codebook design for vector quantization. It tries to solve the given problem of image compression in an efficient way[$6-8$].

The main objective of this study is to generate the codebook through a vector quantizer using reinforced learning in genetic algorithm to get the best image compression. The aim is to find a M code vectors for a given set of N training vectors using the evolutionary computation techniques.

This paper is organized as follows section 1 is the introduction outlining the background and purpose of this study. In section 2 Codebook generation along with the experimental strategies evolutionary computation are explained. Section 3 presents a discussion on the results. Finally the conclusion of the research is given in Section 4.

2 Codebook generation with Evolutionary Computation

We study the problem of generating a codebook for a vector quantizer (VQ). The aim is to find M code vectors (codebook) for a given set of N training vectors (training set).

An image is first converted into the set of $X = x_1, x_2, ..., x_N$ of N training vectors in a K-dimensional Euclidean space to find a codebook

$C = c_1, c_2, ..., c_M$ of M code vectors. The aim is to minimize the average distance between the training vectors and their representative code vectors. The distance between two vectors is defined by their squared Euclidean distance.

$$d^2 = \sum_{i=1}^{N} ||x_i - c_{pi}||^2. \quad (1)$$

The distortion of the codebook is then calculated as

$$D(P, C) = \frac{1}{N} d^2. \quad (2)$$

For codebook generation, two optimizations are essential, the first is the Partition optimization and the second is the codebook centroid optimization.

The optimization of partition P is obtained by placing each training vector x_i to its nearest code vector c_j so as to minimize euclidian distance (1) that in turn optimize (3) that can be given by

$$p_i = \arg min ||x_i - c_j||^2. \quad (3)$$

The optimization of the codebook C is obtained by calculating the centroid of the clusters c_j as the code vectors

$$c_j = \frac{\sum_{p_i=j} x_i}{\sum_{p_i=j} 1}, 1 \leq j \leq M \quad (4)$$

that in turn optimizes (3).

Conventional PNN by Equitz [10] uses the hierarchical approach of generating codebook. It starts by considering each vector as a separate code vector. Then it converge/merge the two vectors whose distortion is minimum. This process goes on till the desired size of the codebook is achieved. The distortion of the merge is calculated as

$$d_{a,b} = \frac{n_a n_b}{n_a + n_b} . ||c_a - c_b||^2 \quad (5)$$

where c_a and c_b are the merged code vectors, n_a and n_b are the size of the corresponding clusters. The PNN approach is used in GA.

Franti and Kaukoranta (1998) introduced a faster method of PNN [8]. Their main idea is to maintain the pointer of nearest neighbour to avoid the calculation of the distance between the two neighbours. After each merge the pointer table is to be updated for the merged vectors only. This in turn reduces computational time significantly.

The most cited and widely used method is GLA [4]. It starts with an initial codebook, which is iteratively improved until a local minimum is reached. The result of the GLA is highly dependent on the choice of the initial codebook. Better results can be obtained by using an optimization technique known as GA [9]. Sadaf and Sajjad has compared the GA techniques for image compression [11, 12]

GA starts with initial random selection of population. In our case the fitness function is calculated by calculating the distortion through (Eq. 2).

All the chromosomes are encoded into the binary strings and genes were selected for crossover through various selection criteria in order to get better results. As reinforcement can rapidly increase the performance of GA, and it prevents a loss of the best-found solution so we first copies the best chromosome to the new population and then the rest of the population is constructed again.

Mutation is performed as an exchange of two genes on the result of crossed over genes and the process again starts with the population selection for T times. The probability for cross over is high and for mutation it is very low.

A very basic and most important choice in the GA is the way solution is represented, since the data structure is determined through it. That in turn are modified through crossover and mutation to give optimal solution.

In 2000 Pasi Franti used the combination of partition and the codebook together to overcome the computational inefficiency of recalculating Partition optimality P and Codebook optimality C [7].

The selection method in GA is based on the Darwin theory of "survival of the fittest". In Reinforced GA, though selection is done in a greedy way in which all possible pairs are permutated. Than they are organized in an increasing order by their distortion values. With reinforced learning approach solution with least distortion are selected and other are discarded.

In crossover reinforcement is again applied by combining the two best existing solution in order to form a new parent. In addition to that, unnecessary computation is not wasted for a complete repartition but the partitions of the parent solutions are utilized.

It starts by taking the union of the two existing solution (i.e. parents) in order, and than merging them. The partition closer to the vector (smaller in distance) is chosen. The new codebook is updated using (4) in respect to the new partition. In this way we get twice the size of codebook (i.e. $2M$ instead of M).

The final size of the codebook is then obtained using the PNN algorithm. The first step of the PNN is to search for each code vector its nearest neighbour that minimizes the merge cost according to (5). After the merge, the pointers are updated, and the process is repeated until the size of the codebook is reduced to M.

In Mutations first we randomly choose code vector, then training vector is also chosen randomly for swaping. This method is known as random swap. Its basic purpose is to discover new search paths when the population becomes too homogenous for the crossover to achieve significant improvement anymore. Mutations if becomes vital, suggests that the crossover is not well-defined and needed to be modified.

3 Results and Discussions

The problem of generating a codebook for a vector quantizer (VQ) through reinforced learning is studied. For a given set of N training vectors we find M code vectors. The proposed system is coded in Matlab language and run on UNIX. We consider a set $X = x_1, x_2, ..., x_N$ of N training vectors to find a codebook $C = c_1, c_2, ..., c_M$ of M code vectors. The distortion of the encoded picture is measured by the peak-signal-to-noise ratio (PSNR).

$$PSNR = 10log_{10}[\frac{(2^n-1)^2}{\frac{1}{M}\sum_{i=1}^{M}(x_o - x_i)^2}] \qquad (6)$$

"Lena" image with resolution 256x256 pixels, 8 bits per pixel, is used here. The image is divided into 4x4 blocks for training; the codebook size is $M = 256$.

Figure 1: Original Image (256x256)

4 Conclusion

In this research, the problems of codebook design are studied and vector quantization with re-

Figure 2: Compressed Image with PSNR = 34.6

inforcement learning in GA is performed.

It is concluded from the result that Reinforced Learning can be successfully applied for image compression using evolutionary computation methods.

References:

[1] Richard S. Sutton and Andrew G. Barto "Reinforcement Learning: An Introduction" A Bradford Book, The MIT Press Cambridge, Massachusetts London, England 2005

[2] John Koza, "Genetic Programming: On the Programming of Computers by Means of Natural Selection," MIT Press, 1992.

[3] R. C. Gonzelez, and R. E. Woods, "Digital Image Processing", Newyork, Addison-Wesley, 1993.

[4] Y. Linde, A. Buzo, and R.M. Gray, "An Algorithm for Vector Quantizater Design", IEEE Transaction on Comm., pp. 84-95 Vol. 28 January 1980.

[5] A. Gresho and R. Gray, " Vector Quantization and Signal Compression" Bostan MA: Kluwer, 1992.

[6] P. Franti, J. Kivijarvi, T. Kaukoranta and O. Nevalainen, "Genetic algorithms for large scale clustering problem", The Computer Journal, 40 (9), 547-554, 1997.

[7] P. Franti, "Genetic algorithm with deterministic crossover for vector quantization", Pattern Recognition Letters, 21 (1), 61-68, 2000.

[8] P. Franti, T. Kaukoranta, D.-F. Shen and K.-S. Chang, "Fast and memory efficient implementation of the exact PNN", IEEE Transactions on Image Processing, 9 (5), 773-777, May 2000.

[9] M.Mitchell, "An Introduction To Genetic Algorithms", MIT press, Cambridg,e England, (1998).

[10] W.H. Equitz, "A new vector quantization clustering algorithm", IEEE Trans. on Acoustics, Speech and Signal Processing, pp. 1568-1575 Vol. 37, 1989.

[11] Sadaf Sajjad, Sajjad Mohsin "Comparative Study on VQ with Simple GA and Ordain GA" Proceedings of the 9th WSEAS International Conference on AUTOMATIC CONTROL, MODELING & SIMULATION (AMCOS '07) Istanbul, Turkey, May 27-29, 2007 pp. 203-207, ISBN: 978-960-8457-72-0, ISSN: 1790-5117

[12] Sajjad Mohsin, Sadaf Sajjad "Codebook generation for Image Compression with Simple and Ordain GA" International Journal of Computers and Communications, Vol. 1, No. 2, 2007 pp. 35-40

Sadaf Sajjad[a] received the M.Sc. Clinical Psychology degree from University of Peshawar, Pakistan, in 1995 and secured top position in Pakistan for which she was awarded Gold medal. She got her Advance Diploma in Clinical Psychology from Centre for Clinical Psychology, University of Punjab, Lahore Pakistan in 1997. She got M.E. degree in Computer Science and Systems Engineering from the MIT, Japan, in 2004. She is Assistant Professor/University Counselor in COMSATS Institute of Information Technology, Islamabad, Pakistan. Her research interests include Genetic Algorithms, Vector Quantization, Image Compression, Cognitive Psychology, Psycholinguistics, Gestalt Therapy, Rational Emotive Therapy, Perceptual Psychology. She has four research publications to her credit.

Sajjad Mohsin[b] received the M.Sc. Computer Science degree from Quaid-i-Azam University, Islamabad, Pakistan, in 1987, the M.E. degree in Computer Science and Systems Engineering from the Muroran Institute of Technology, Japan, in 2002, and the Ph.D. degree from the Muroran Institute of Technology, Japan, in 2005. He has 15 journal and conference publications. He is member editorial board of two international journals. He is Associate Professor in Department of Computer Science, COMSATS Institute of Information Technology, Islamabad, Pakistan. His research interests include Neural Networks, Genetic Algorithms, Fuzzy Logic, Soft Computing, Evolutionary Computing, Pattern Recognition, Vector Quantization.