Expert Systems with Applications 38 (2011) 1382–1392

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Image vector quantization algorithm via honey bee mating optimization

Ming-Huwi Horng *, Ting-Wei Jiang

Department of Computer Science and Information Engineering, National Pingtung Institute of Commerce, Pingtung, Taiwan

ARTICLE INFO

Keywords:
Vector quantization
LBG algorithm
Particle swarm optimization
Quantum particle swarm optimization
Honey bee mating optimization

ABSTRACT

The vector quantization (VQ) was a powerful technique in the applications of digital image compression. The traditionally widely used method such as the Linde–Buzo–Gray (LBG) algorithm always generated local optimal codebook. Recently, particle swarm optimization (PSO) is adapted to obtain the near-global optimal codebook of vector quantization. An alternative method, called the quantum particle swarm optimization (QPSO) had been developed to improve the results of original PSO algorithm. In this paper, we applied a new swarm algorithm, honey bee mating optimization, to construct the codebook of vector quantization. The results were compared with the other three methods that are LBG, PSO–LBG and QPSO–LBG algorithms. Experimental results showed that the proposed HBMO–LBG algorithm is more reliable and the reconstructed images get higher quality than those generated from the other three methods.

1. Introduction

The vector quantization (VQ) techniques have been used for a number of years for image compression. The operations of VQ techniques include the partition of the image into many input vectors (or blocks), and then compare to the codewords of codebook in order to find its reproduction vector. The codeword, which is most similar to an input vector, is called the reproduction vector of input vector. In the encoding process, an index that points to the closest codeword of an input vector is determined. In general, the size of the codebook is much smaller than the original image data set. Therefore, the purpose of image compression is achieved. In the decoding process, the associated sub-image is exactly retrieved by the same codebook which has been used in the encoding phase. When each sub-image is completely reconstructed, the decoding is completed.

Vector quantization (VQ) algorithms have been performed by many researchers; new algorithms continue to appear. The generation of codebook is known as the most important process of VQ. Patane and Russo (2001) classified all of the VQ algorithms into two categories: (1) competitive learning-based and (2) *k*-means-based. In the competitive learning-based methods, the codebooks are obtained by means of a consequence of a process of mutual competition. The typical methods of competitive learning-based group include the self-organizing map (SOM) (Kohonen, 1982), growing neural gas (GNG) (Frezza-Buet, 2007), and neural network (NN) (Han, Chen, Lo, & Wang, 2007).

The *k*-means-based algorithms are designed to minimize distortion error by selecting a suitable codebook. A well-known method is the LBG algorithm (Linde, Buzo, & Gray, 1980); however, the LBG algorithm is a local search procedure. It suffers from the serious drawback that its performance depends heavily on the initial starting conditions. Recently, the evolutionary optimization algorithms had been developed to design the codebook for improving the results of LGB algorithm. Rajpoot, Hussain, Ali, Saleem, and Qureshi (2004) applied the ant colony system algorithm to develop the algorithms for the codebook design. The generation of codebook using ACS is facilitated by representing the coefficient vectors in a bidirectional graph, followed by defining a suitable mechanism of depositing pheromone on the edges of graph. Chen, Yang, and Gou (2005) proposed an improvement based on the particle swarm optimization (PSO). The result of LBG algorithm is used to initialize global best particle by which it can speed the convergence of PSO. In addition, Wang et al. (2007) proposed a quantum particle swarm algorithm (QPSO) to solve the 0–1 knapsack problem. Zhao, Fang, Wang, and Pang (2007) employed a quantum particle swarm optimization to select the thresholds of the multilevel thresholding. The algorithm of QPSO computes the local point from the *pbest* and *gbest* for each particle and further defines two parameters, *u* and *z*, update the position of corresponding particle.

Over the last decade, modeling the behavior of social insects, such as ants and bees, for the purpose of search and problems solving had been the context of the emerging area of swarm intelligence. Therefore, the honey bee mating may also be considered as a typical swarm-based approach for searching for the optimal solution in many application domains such as clustering (Fathian, Amiri, & Maroosi, 2007), market segmentation (Amiri & Fathian,

* Corresponding author. Tel./fax: +886 8 7238700.
E-mail address: horng@npic.edu.tw (M.-H. Horng).

0957-4174/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2010.07.037

2007), image coding (Jiang, 2009; Horng, 2009) and multi-level image thresholding selection (Horng, 2010). In this paper, the honey bee mating optimization (HBMO) associated with the LBG algorithm is proposed to search for the optimal codebook that minimizes the distortion between the input vectors and the codewords of codebook. In other words, HBMO algorithm is a search technique that finds the optimal codebook form the training of the input vectors. Experimental results demonstrated that the HBMO–LBG algorithms performed better than those of the LBG, PSO–LBG, and QPSO–LBG algorithms consistently.

This work is organized as follows: Section 2 introduces the vector quantization and LBG algorithm. Section 3 presents the PSO–LBG algorithm and QPSO–LBG algorithm for designing the codebook of the vector quantization. Section 4 presents the proposed method which searches for the optimal codebook using the HBMO algorithm. Performance evaluation is discussed in detail in Section 5. Conclusions are presented in Section 6.

2. Vector quantization and LBG algorithm

This section provides some basic concepts of vector quantization and introduces the traditional LBG algorithm.

2.1. Definition

Vector quantization (VQ) is a lossy data compression technique in block coding. The generation of codebook is known as the most important process of VQ. Let the size of original image $Y = \{y_{ij}\}$ be $M \times M$ pixels that divided into several blocks with size of $n \times n$ pixels. In other words, there are $N_b = \left[\frac{N}{n}\right] \times \left[\frac{N}{n}\right]$ blocks that represented by a collection of input vectors $X = (x_i, i = 1, 2, \ldots, N_b)$. Let L be $n \times n$. The input vector x_i, $x_i \in \Re^L$, where $\Re^L$ is L-dimensional Euclidean space. A codebook C comprises N_c L-dimensional codewords, i.e. $C = \{c_1, c_2, \ldots, c_{N_c}\},\ c_j \in \Re^L,\ \forall j = 1, 2, \ldots, N_c$. Each input vector is represented by a row vector $x_i = (x_{i1}, x_{i2}, \ldots, x_{iL})$ and each codeword of the codebook is denoted as $c_j = (c_{j1}, c_{j2}, \ldots, c_{jL})$. The VQ techniques assign each input vector to a related codeword, and the codeword will replace the associated input vectors finally to obtain the aim of compression.

The optimization of C in terms of mean square error (MSE) can be formulated by minimizing the distortion function D. In general, the lower the value of D is, the better the quality of C will be.

$$D(C) = \frac{1}{N_b} \sum_{j=1}^{N_c} \sum_{i=1}^{N_b} \mu_{ij} \cdot \|x_i - c_j\|^2, \tag{1}$$

subject to the following constraints:

$$\sum_{j=1}^{N_c} \mu_{ij} = 1, \quad \forall i \in \{1, 2, \ldots, N_b\}, \tag{2}$$

$$\mu_{ij} = \begin{cases} 1 & \text{if } x_i \text{ is in the } j\text{th cluster}, \\ 0 & \text{otherwise} \end{cases}, \tag{3}$$

and

$$L_k \leqslant c_{jk} \leqslant U_k, \quad k = 1, 2, \ldots, L, \tag{4}$$

where L_k is the minimum of the kth component in the all training vectors, and U_k is the maximum of the kth component in all input vectors. The $||x - c||$ is the Euclidean distance between the vector x and codeword c.

Two necessary conditions exist for an optimal vector quantizer.

(1) The partition R_j, $j = 1, \ldots, m$ must satisfy

$$R_j \supset \{x \in X : d(x, c_j) < d(x, c_k),\ \forall k \neq j\}. \tag{5}$$

(2) The codeword c_j must be given by the centroid of R_j:

$$c_j = \frac{1}{N_j} \sum_{i=1}^{N_j} x_i, \quad x_i \in R_j, \tag{6}$$

where N_j is the total number of vectors belonging to R_j

2.2. LBG algorithm

An algorithm for a scalar quantizer was proposed by Lloyd (1957). Linde et al. (1980) generalized it for vector quantization. This algorithm is known as LBG or generalized Lloyd algorithm (GLA). It applies the two following conditions to input vectors for determining the codebook.

Given input vectors, x_i, $i = 1, 2, \ldots, N_b$, distance function d, and an initial codewords $c_j(0)$, $j = 1, \ldots, N_c$. The LBG iteratively applies the two conditions to produce optimal codebook with the following algorithm:

(1) Partition the input vectors into several groups using the minimum distance rule. This resulting partition is stored in a $N_b \times N_c$ binary indicator matrix U whose elements are defined as the following:

$$\mu_{ij} = \begin{cases} 1 & \text{if } d(x_i, c_j(k)) = \min_p d(x_i, c_p(k)), \\ 0 & \text{otherwise}. \end{cases} \tag{7}$$

(2) Determine the centroids of each partition. Replace the old codewords with these centroids:

$$c_j(k+1) = \frac{\sum_{i=1}^{N_b} \mu_{ij} x_i}{\sum_{i=1}^{N_b} \mu_{ij}}, \quad j = 1, \ldots, N_c. \tag{8}$$

(3) Repeat steps (1) and (2) until no c_j, $j = 1, \ldots, N_c$ changes anymore.

3. PSO–LBG and QPSO–LBG algorithms

Particle swarm optimization (PSO) is a new branch of evolutionary computation technique originally presented by the work of Kennedy and Eberhart (1995). In the multi-dimensional space, each particle represents a potential solution to a problem. There exists a fitness evaluation function that assigns a fitness values to this potential solution for designing the codebook C based on the Eq. (9).

$$\text{Fitness}(C) = \frac{1}{D(C)} = \frac{N_b}{\sum_{j=1}^{N_c} \sum_{i=1}^{N_b} \mu_{ij} \cdot \|x_i - c_j\|^2}. \tag{9}$$

Two positions are recorded by every particle. One is named global best (*gbest*) position, which has the highest fitness value in the whole population. The other is called personal best (*pbest*) position, which has the highest fitness value of itself at present. The population of particles is flying in the search space and every particle changes his position according the *gbest* and the *pbest* with Eqs. (10) and (11).

$$v_{ik}^{n+1} = v_{ik}^n + c_1 r_1^n (pbest_{ik}^n - x_{ik}^n) + c_2 r_2^n (gbest_k^n - x_{ik}^n), \tag{10}$$

$$x_{ik}^{n+1} = x_{ik}^n + v_{ik}^{n+1}, \tag{11}$$

where k is the number of dimensions ($k = 1, 2, \ldots, L$) and i represents a particle of the population ($i = 1, 2, \ldots, s$). X means the position of particle in the search space; v is the velocity vector for the particle to change its position. Parameters c_1 and c_2 are the cognitive and social learning rates, respectively. r_1 and r_2 are two random numbers that belongs to [0, 1]. Suppose that there exist s particles