Available online at www.ijsrnsc.org

IJSRNSC

Volume-6, Issue-3, June 2018
Research Paper

www.ijsrnsc.org

Int. J. Sc. Res. in
Network Security
and Communication

ISSN: 2321-3256

# Adaptive Vector Quantization for Improved Coding Efficiency

**S. Vimala[1*], P. Uma[2], S. Senbagam[3]**

[1]Dept. Of Computer Science, Mother Teresa Women's University, Kodaikanal, Tamil Nadu, India.
[2]Dept. Of Computer Science, Mother Teresa Women's University, Kodaikanal, Tamil Nadu, India.
[3]Dept. Of Computer Science, Mother Teresa Women's University, Kodaikanal, Tamil Nadu, India.

*[*]Corresponding Author:vimalaharini@gmail.com, Tel.: +91-94446-90081*

Received: 02/Mar/2018, Revised: 16/Mar/2018, Accepted: 11/Apr/2018, Published: 30/Jun/2018

***Abstract--*** In this paper, we propose a novel method of improving the initial codebook for Vector Quantization (VQ) to compress still images. VQ is a simple and efficient compression technique which comprises of three phases: 1. Codebook Generation 2. Index Map Generation and 3. Image Reconstruction. Default codebook is generated first and is improved by refining the representative vectors called the code vectors. The codebook optimization technique proposed in this paper improves the quality of reconstructed images to a greater extent where the average *bpp* is decreased to a value of 0.79 which is a significant improvement. Benchmark images such as Lena, Kush, Cameraman, Barbara are tested with the proposed technique and this technique produces better results.

***Keywords***- Image Compression, Vector Quantization, Index Map, bpp, Still Image

## I. INTRODUCTION

### A. *Vector Quantization*

Image compression is a technique for reducing the storage space required for storing the data. This leads to reduction in cost associated with storage and transmission of data [1]. Compression techniques are of two types: (i) Lossy Compression and (ii) Lossless Compression. The image that has been compressed using Lossy compression method cannot be reconstructed exactly and the Lossy compression technique is suitable for compressing natural images. The reconstructed images out of Lossless compression technique will be exact replica of the original images and the lossless technique is suitable for medical images [2]. The process of Compression procedure is illustrated in Fig. 1.

In compression model, as in Fig. 1, the encoder transforms the input image into reduced set of data, which in turn is used for reconstructing the compressed image. The reconstruction of image is done in Decoder [3]. The quality of the reconstructed image is measured using a metric called Mean Square Error (MSE) and the inverse of the same is Peak Signal to Noise Ratio (PSNR). Hence the MSE must be less and the PSNR value must be high. MSE and PSNR are calculated using(1) and (2).

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - x_i)^2 \qquad (1)$$

$$PSNR = 10\log_{10}\left[\frac{255^2}{MSE}\right] \qquad (2)$$

Quantization is a process of transforming large image values into reduced set of values [4]. There are two types of Quantization (i) Scalar Quantization (ii) Vector Quantization. In Scalar Quantization, all input values are treated separately. In Vector Quantization (VQ), all input values are grouped into number of vectors. VQ is one of the simple and efficient compression techniques. VQ enables to have improved coding efficiency. Vector Quantization comprises of three phases: 1. Codebook Generation, 2. Image Encoding and 3. Image Decoding.

In VQ, the input image is divided in to non-overlapping blocks of size 4 x 4 pixels and the set of these blocks is called Training Set (TS). The individual vectors in the TS are called Training Set. Representative vectors from TS are selected to form a codebook of desired size n. All training vectors are searched against the codebook and the corresponding indices are stored collectively as IndexMap. The compressed image is stored or transmitted in the form of Codebook and IndexMap. In the receiving end, image decoding is done. Each index in the IndexMap is transformed into the corresponding codevector. The image thus reconstructed will be an approximation of the original image [5].

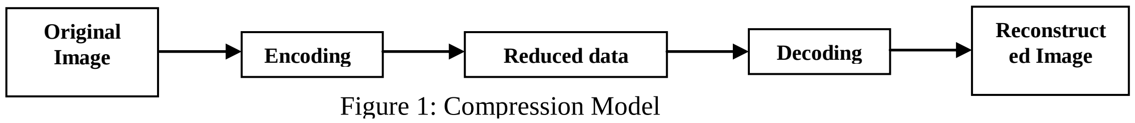


Figure 1: Compression Model

$d(x_i, c_i) = d(x_i - c_i)/N$ (3)

where N is the size of the Codebook

## II. EXISTING METHOD

*A. Vector Quantization*

- Divide the image to be coded into vectors or blocks.
- Select the representation vectors to form to codebook.
- Vector quantizes each block by searching the codebook and finding the codeword that matches the image block with lease distortion.
- Transmit the index of the selected codeword.
- At the decoder, retrieve the codeword that corresponds to the received index.
- The image is reconstructed when all the codeword have been retrieved.

*B. Codebook Generation using Vector Quantization*

An image is first converted into the set of $X = \{X_1,X_2,X_3,...X_N\}$ of N training vectors in a K-dimensional Euclidean space to find a codebook $C= \{C_1,C_2,C_3,...C_M\}$ of M code vectors. The distance between two vectors is defined by their Euclidean distance.

$D=\sum^{N}_{i=1} \| X_i - C_i \|$ (4)

*C. Linde Buzo Gray (LBG)*

In this method, the image is divided into the number of vectors which is called as training vectors [6]. The Centroid vector is generated by calculating the average of all vectors in training set. A constant error is added and subtracted respectively to the Centroid to obtained two vectors V1 and V2. Compare these two vectors with the each vector in training set and find the minimum distance. Based on these distance group all vectors in to two clusters and again the constant error is added to the centroid to get further code vectors. Repeat these steps until the desired size of codebook is generated [7].

*D. LBG Algorithm*

**Step 1:** Input the given image of size N × M pixels.
**Step 2**: Divide the input image into 4 × 4 blocks.
**Step 3:** Generate Training set with N vectors.
**Step 4:** Calculate the average to find the Centroid vector.
**Step 5:** Add and subtract the constant error {1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1} to the Centroid respectively to get two vectors V1 and V2.
**Step 6:** Calculate the minimum distance to compute two clusters which is grouping the nearest Vectors of V1 and V2.
**Step 7:** Repeat the Step 4, 5, and 6 until the desired size of codebook is generated.

*E. Improved Differential Evolution (IDE)*

The IDE method is combined with LBG algorithm to improve the codebook efficiency. The input image is converted into N number of blocks to generate the Training Set. Each codevector from the codebook is compared against all the training vectors. The closest vectors are identified by computing the Euclidean Distance using (5). The codebook is revised by adding the centroid generated from vectors of the same group. Calculate the total distortion (Dn) between the centroids of subsequent iterations of all vectors is calculated from respective centroid values. If $(D_{n-1} – D_n) < \varepsilon$ then the above process terminates & the current codebook is taken to final otherwise repeat the above steps [8].

## III. PROPOSED METHOD

*A. Improved Code Book Optimization*

In the proposed work, the performance of VQ is improved in terms of compression rate. In the existing techniques such as Simple Codebook Generation (SCG) [9], all the input blocks are treated equally. The initial codebook thus generated is further improved using the clustering method. This improves the quality of the reconstructed image. Pick up a codebook from $C_j$ in CB. Find all training vectors $X_i$ that are closer to $C_j$ to form a cluster ***j*** by computing the Euclidean Distance using (5).

$d(X_i, C_j) < th$ for $i \neq j$ (5)

where *th* is a threshold value, 1 <= i <= 4096 and 1<=j<=256.

The distance between the training vectors $X_i$ and the codevector $C_i$ is computed using (5)

$d(X_i,C_j) = \sum^{k}_{j=1} | X_{ik} – C_{jk} |$ (6)

Calculate the sum vector by adding all the training vectors $X_i$ that are closer to $C_j$ from (6).

$Sum_{jk} = \sum_{j=1}^{n} X_{jk}$ (7)

where 1 <= k <= 16 and *n* is the size of the cluster.
The centroid for the cluster ***j*** is generated from (8)

$Cen_{jk} = Sum_{jk}/n$ (8)

where ***k*** ranges from 1 to 16 and ***j*** is the cluster number. The jth codevector in the codebook is now replaced with the newly generated centroid. Every time, when a new centroid is generated, the difference between the present centroid and the previous centroid is computed. If this distortion is less than a minimum error value ε, the iteration is stopped and the current set of centroids form the codevectors of the final codebook, which is a refined one.

B. *Codebook Optimisation Algorithm*

**Step 1:** Input image is converted into blocks called training vectors and generate the codebook by selecting the training vector from every pth position from the training vectors.
**Step 2:** Remove the duplicate vector using (5) to optimize the codebook size.

**Step 3:** The Training vectors are grouped into M clusters based on the Euclidean distance between the codevector and the training vectors using (6).

**Step 4:** Compute the sum vector for every cluster by adding the corresponding components of all the training vectors that belong to the same cluster using (7).

**Step 5:** Compute the centroid for each cluster by dividing the individual components of the sum vector by the cluster size ***n*** using (8).

**Step 6:** Find the distortion between the present centroid and the previous centroid.

**Step 7:** If the sum of distortion is less than ε, terminate the process. Otherwise, replace the existing centroid with the new centroid to form the revised codebook.

**Step 8:** Repeat the steps 1 through 4 till the codebooks of the consecutive iterations coverage.

## IV. RESULTS AND DISCUSSION

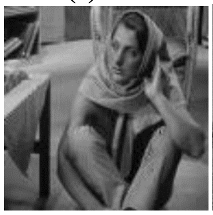
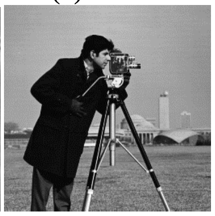
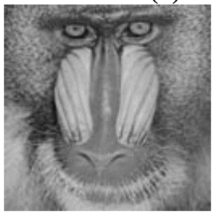
Bench marks images such as Lena, Kush, Cameraman, Barbara and Baboon are tested with the proposed method using Matlab R2014b version. VQ and IDE give a uniform bpp of 1 for storing the images
but the proposed method is an adaptive vector quantization, where the bitrate changes depending on the nature of the gray level distribution and yields an average bpp of 0.79 bits per pixel, which is a significant improvement.

The simulated results of the proposed method in terms of bpp are compared against the results obtained with the existing techniques such as VQ and IDE. In all the cases, the results of the proposed method are better. A minimum bpp of value 0.62 is obtained for Cameraman image and a maximum bpp value of 0.94 is achieved for Kush image.

Table 1: Results in terms of bpp with respect to VQ, IDE and the proposed method

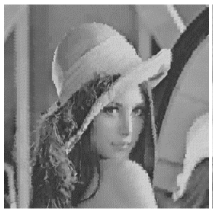
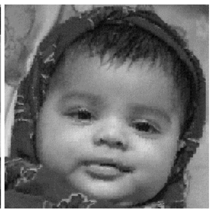
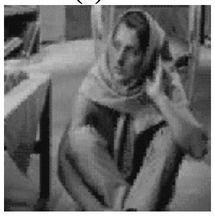
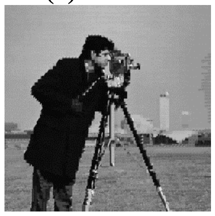
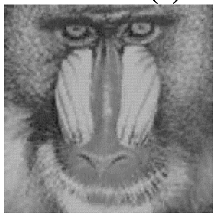
| *Image* | VQ | IDE | **Proposed Method Threshold Value** | |
|---|---|---|---|---|
| | | | **32** | **48** |
| | BPP | BPP | **BPP** | **BPP** |
| *Lena* | 1 | 1 | 0.86 | **0.69** |
| *Kush* | 1 | 1 | 0.98 | **0.94** |
| *Cameraman* | 1 | 1 | 0.74 | **0.62** |
| *Barbara* | 1 | 1 | 0.91 | **0.81** |
| *Baboon* | 1 | 1 | 0.98 | **0.91** |
| **Average** | **1** | **1** | **0.89** | **0.79** |
| | | | | |

(a)Lena

(b) Kush

(c) Barbara

(d) Cameraman

(e) Baboon

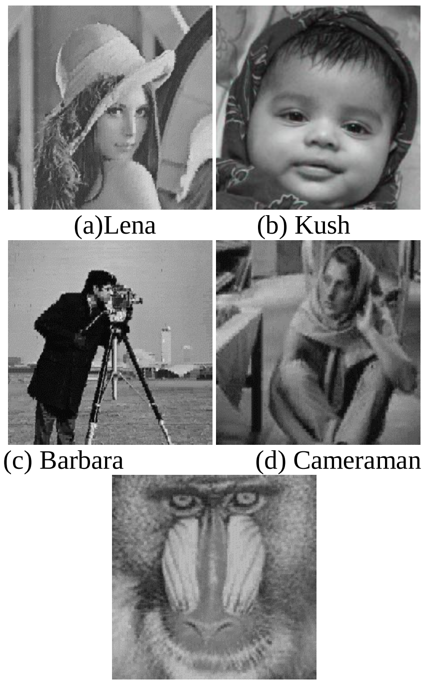
Average bpp : 8 bits

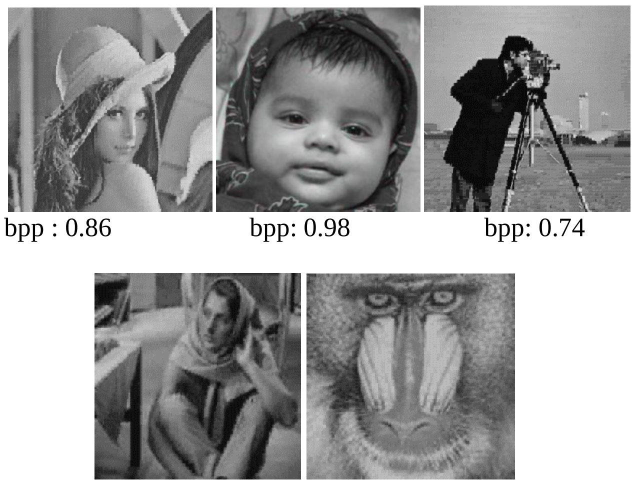
Figure 2. Input Images


(a)Lena

(b) Kush

(c) Barbara

(d) Cameraman

(e) Baboon

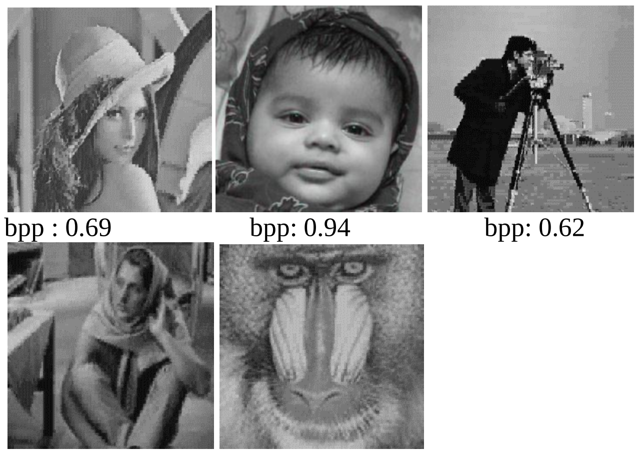
Average bpp: 1 bit

Figure 3. Vector Quantization Technique


(a)Lena (b) Kush
(c) Barbara (d) Cameraman
(e) Baboon
Average bpp: 1 bit
Figure 4. Improved Differential Evolution


bpp : 0.86 bpp: 0.98 bpp: 0.74
bpp: 0.91 bpp: 0.98
Average bpp: 0.89
Figure 5. Adaptive Vector Quantization (Threshold: 32)

The reconstructed images using the methods VQ, IDE and the proposed method are given in Fig. 2, 3, 4, 5 and 6 for visual comparison. Fig. 2 gives the input images taken for the study. Fig. 3 and Fig. 4 give the list of compressed images that are reconstructed using VQ method and IDE method respectively. Fig. 5 and 6 represent the images reconstructed using the proposed method with two different threshold values 32 and 48, used for reducing the number of duplicate vectors.


bpp : 0.69 bpp: 0.94 bpp: 0.62
bpp: 0.81 bpp: 0.91
Average bpp: 0.79
Figure 6. Adaptive Vector Quantization (Threshold: 48)

## V. CONCLUSION

In this paper, the initial codebook that is generated using SCG is improved by eliminating duplicate code vectors leading to improved coding efficiency. The quality of the reconstructed images is also improved by optimising the initial codebook using the IDE method. The average *bpp* is reduced to a value of 0.79 which is a significant improvement. This method is simple and efficient and gives better results, in terms of *bpp*. Generally when *bpp* is improved, there will be a fall in PSNR and vice-versa. But the proposed method gives better performance in achieving both improved *bpp* and PSNR values. This method is also applicable for compressing color images.

## REFERENCES

[1] Khalid Sayood, *Introduction to DataCompression*",3rd Edition.

[2] Rafael C. Gonzalez, Richard E. Woods, *Digital Image Processing*, 2nd Edition.

[3] R. Gray, "*Vector Quantization*", IEEE ASSP Mag., pp.4-29,1989.

[4] Yoseph Linde, Andres Buzo, Robert M.Gray, "*An Algorithm for Vector Quantizatizer Design*", IEEE Transactions on Communications, Vol. COM-28, pp.84-95, Jan 1980.

[5] Dr. H.B. Kekre, Ms. Tanuja K. Sarode, "*Vector Quantized Codebook Optimization Using KMeans Algorithm*", IJCSE, Vol 1(1), pp. 283-290, 2009.

[6] Arup Kumar and Anup Sar, "*An Efficient Codebook Initialization Approach For LBG Algorithm*", IJCSEA, Vol.1, pp.72-80, Aug 2011.

[7] Ms. Asmita A Bardekar, Mr. P.A. Tijare, "*A Review on LBG Algoithm for Image Compression*", IJCSIT, Vol.2(6), pp. 2584-2589, 2011.

[8] Sayan Nag, "*Vector Quantization Using the Improved Differential Evolution Algorithm for Image Compression*", 2017.

[9] K.Somasundaram and S.Vimala, "*Simple and Fast Ordered Codebook for Vector Quantization*", Proceedings of the National Conferene on Image Processing, Gandhigram Rural Institute (NCIMP), 2010.

## Authors Profile

*Dr. S.Vimala* pursued her UG Degree in M.K.University, P.G. Degree in Bharathiar University and M.Phil. & Ph.D. Degrees in Mother Teresa Women's University. She has authored nearly 33 research papers in reputed international journals and a book on 'C Programming'. She is a life member of IACSIT. She has been awareded the 'Best Thesis Award' by ICFAI Business School, Hyderabad and Michigan State University, USA. Her research focus is on Image Compression. She has twenty years of teaching experience and 10 years of research experience.

*Ms. P.Uma* pursued her UG Degree in M.K.University, P.G. Degree in Anna University, M.Phil. Degree in Mohter Teresa Women's University. She is doing her research in Mother Teresa Women's University. She has authored 10 research papers in reputed International Journals.Her research focus is on 'Image Compression'.

*Ms. S.Senbagam* pursued her UG and PG Degrees in Madurai Kamaraj University. She is doing her M.Phil. Degree in Mother Teresa Women's University and her research focus is on 'Image Compression'.