

<http://ansinet.com/itj>

ITJ

ISSN 1812-5638

# INFORMATION TECHNOLOGY JOURNAL

**ANSI***net*

Asian Network for Scientific Information  
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

## K- Means Clustering Based Image Compression in Wavelet Domain

<sup>1</sup>N. Venkateswaran and <sup>2</sup>Y.V. Ramana Rao

<sup>1</sup>Department of Electronics and Communication Engineering,  
Sri Venkateswara College of Engineering, Sriperumbudur-602 105, India

<sup>2</sup>Department of Electronics and Communication Engineering,  
College of Engineering, Anna University, Chennai-600 025, India

---

**Abstract:** This study suggests a novel image compression scheme, using the discrete wavelet transformation (DWT) and k-means clustering technique. The goal is to achieve higher compression rates based on clustering the wavelet coefficients of each DWT band. This methodology is different from other schemes wherein DWT is applied to the whole original image. Here the sub blocks of size 16×16 from an image are subjected to one level wavelet decomposition and the coefficients are clustered using the well-known K-means clustering algorithm. The centriods of each cluster is arranged in the form of a codebook and indexed. The index values are only transmitted over the line. By decreasing the number of cluster, high compression ratio can be achieved. Simulation results indicate that the compression ratio varies from 56.8 to 8.0 for varying the cluster size from 10 to 100 for an acceptable image quality.

**Key words:** Image compression, K-means clustering, discrete wavelet transform

---

### INTRODUCTION

Importance of image compression increases with advancing communication technology. In the context of image processing, compression schemes are aimed to reduce the transmission rate for still (or fixed images), while maintaining a good level of visual quality. Several compression techniques have been developed, such as DPCM (Differential Pulse Code Modulation), DCT (Discrete Cosine Transform) and DFT (Discrete Fourier Transform), together with numerous vector quantization methods (Cosman *et al.*, 1996). Vector quantization is based on the creation of a codebook, constituted by M elements (codewords). Each codeword is supported by a vector of the same size as the size of the input vectors. Each input vector will be approximated by a codeword in such a way that the global distortion will be minimized.

Recently the application of wavelets in image compression (Antonini *et al.*, 1992) has received significant attention and several very efficient wavelet based image compression algorithms have been proposed. This is due to the fact that the wavelet transform can provide multi-resolution representation for image or signal with localization in both time and frequency domain. In this study we suggested the use of K-means algorithm in wavelet domain for vector

quantization. One function of this algorithm is to realize the quantization of a continuous input space of stimuli into a discrete output space, the codebook. In order to use vector quantization techniques, the transformed image has to be coded into vectors. Each of these vectors is then approximated and replaced by the index of one of the elements of the codebook, the only useful information in this index, for each of the blocks of the original image, thus reducing the transmission bit rate.

### DISCRETE WAVELET TRANSFORM

One of the drawbacks of discrete cosine transformation (DCT) is that it uses blocks of image data to compute the frequency content of the image. When large blocks are used, the transform coefficients fail to adapt to local variations in the image. When small blocks are used (as in JPEG), the image can exhibit blocking artifacts, which impart a mosaic appearance. But the wavelet transform can provide both reasonable spatial frequency isolation and reasonable spatial localization.

The basic flow of the single stage of a wavelet transform is shown in Fig. 1. It consists of a low-pass and high-pass filter pair. Operations are performed separable in both horizontal and vertical directions. All possible combinations of filter and orientation are represented, with a sub sampling by a factor of 2 in each direction. The

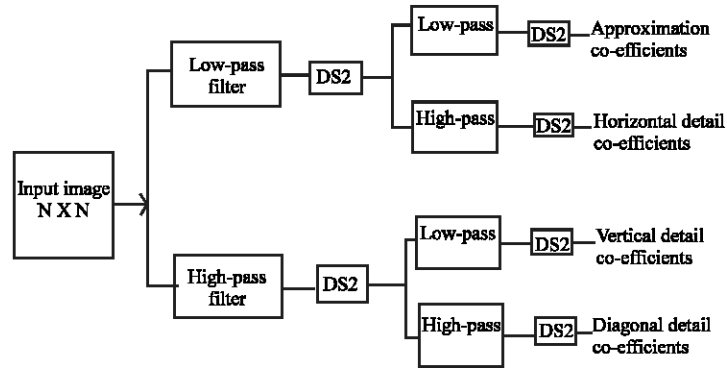


Fig. 1: One level 2-D wavelet decomposition

result is four sub-images, each of which is  $\frac{1}{4}$  the size of the input image. The sub-images passing through one or more high-pass filters retain edge content at the particular spatial frequency represented at this stage of the transform. The sub-image resulting from exclusive application of low-pass filters is a low-resolution version of the starting image and retains much of the pixel-to-pixel spatial correlation that was originally present.

The wavelet transform can be thought of as a filter bank through which the original image is passed (Soman *et al.*, 2005). The filters in the bank, which are computed from the selected wavelet function, are either high-pass or low-pass filters. The image is first filtered in the horizontal direction by both the low-pass and high-pass filters. Each of the resulting images is then down sampled by a factor of two in the horizontal direction. These images are then filtered vertically by both the low-pass and high-pass filters. The resulting four images are then down sampled in the vertical direction by a factor of two. Thus, the transform produces four images as shown in Fig. 1. They are: One that has been high-pass filtered in both directions, one that has been high-pass filtered horizontally and low-pass filtered vertically, one that has been low-pass filtered horizontally and high-pass filtered vertically and one that has been low-pass filtered in both directions. The high-pass filtered images appear much like edge maps. They typically have strong responses only where significant image variation exists in the direction of the high-pass filter. The image that has been low-pass filtered in both directions appears much like the original-in fact, it's simply a reduced-resolution version. As such, it shares many of the essential statistical properties of the original. The transform result can then be quantized and coded to reduce the overall amount of information. Low-spatial-frequency data are retained with greatest

fidelity and higher spatial frequencies can be selectively de-emphasized or discarded to minimize the overall visual distortion introduced in the compression process.

**K-means clustering algorithm:** Clustering is an unsupervised task that can divide the given set of data into several non-overlapping homogenous groups. Each group, called cluster, consists of objects that are similar between themselves and dissimilar to objects of other groups. Clustering technique is extensively used in various applications in engineering, statistics and numerical analysis. The k-means algorithm is by far the most popular clustering tool used in scientific and industrial applications. The name comes from representing each of K clusters  $S_j$  by the mean (or weighted average)  $s_j$  of its points, called as the center of the cluster or the centroid. The sum of the Euclidean distance  $d(x,y)$  between a point  $(x_i)$  and its centroid  $(s_j)$  is used as an objective function. The k-means algorithm partitions the data to minimize the criterion:

$$E = \sum_{j=1}^k \sum_{x_i \in S_j} d^2(x_i, s_j)$$

The K-means clustering algorithm is shown in Fig. 2. It usually involves choosing a random initial partition or centers and repeatedly recomputing the centers based on partition and then recomputing the partition based on the centers. The k-means algorithm is popular because it is simple, easy to implement and works with any standard norms. It is also insensitive to data ordering. The K-means algorithm also has some drawback: 1) The result strongly depends on the initial guess of centriods. 2) It is not obvious what is correct value of number of clusters and 3) The resulting clusters can be unbalanced. The basic K-means algorithm (Jain *et al.*, 1999) used for clustering the data in this paper is outlined below.

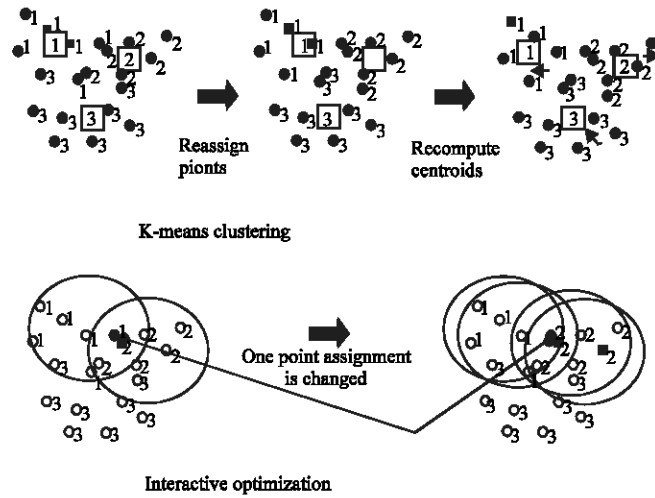


Fig. 2. K-means clustering algorithm

**Basic K-means algorithm:** The basic steps involved in K-means clustering is outlined as:

- Select K-data points as the initial centroids.
- (Re) assign all points to their closest centroids.
- Recompute the centroid of each newly assembled cluster.
- Repeat steps 2 and 3 until the centroids do not change.

By representing data by less number of clusters, the fine details of the data are lost, but the representation is simpler.

**PROPOSED METHOD**

The compression method proposed in this paper combines the powerful Discrete Wavelet Transform (DWT) (Mallat, 1989), (DeVore *et al.*, 1992) and the general k-means clustering method in order achieve high compression ratio with minimal loss of quality. The steps necessary for the method are illustrated in block diagram of Fig. 3. First the image is divided in to sub blocks of size 16×16. A one level wavelet decomposition of the sub blocks is carried out using the Daubechies family of basis functions, which is widely used in image compression. These results in four sub bands for each sub block of size 8×8: LL, LH, HL and HH. The notation L and H stand respectively for low pass and high pass filtering (the first letter denotes vertical and the second horizontal directions). Figure 1 shows the typical one level wavelet decomposition for images. Each Sub band coefficients are arranged in a 1 × 64 vector. This is done for all sub blocks of an image.

After collecting the coefficients from all sub blocks, K-means clustering is performed for individual sub band wavelet coefficients as shown in Fig. 4. For a specific value of the cluster size K, an equal number of centroid vector of size 1 × 64 results. These centroids are stored in the codebook and the corresponding index value is used for encoding and transmission. The advantage of this approach is that only index values representing the closest approximation of the given coefficient vector is chosen, thus enabling high compression factor. At the receiver the index values are decoded. From the copy of the codebook, the wavelet coefficients for each sub bands are retrieved based on the index. Inverse discrete wavelet transform is applied on these coefficients and the image is reconstructed.

**The complete algorithm:**

- The given image matrix is divided in to (b×b) sub matrices. Here for our proposed work b is taken as 16.
1. Each sub matrix is transformed by DWT.
  2. Each sub band decomposition is rearranged in to matrix (b/2×b/2).
  3. Each sub band matrix of a given sub matrix is reshaped in to a (1×b<sup>2</sup>/4) vector.
  4. The vectors thus obtained are clustered to get N centroids. Here for our proposed work, 'N' varies from 10 to 100.
  5. Centroids are stored as the codebook entries.
  6. Every input vector is assigned a cluster number, based on minimum Euclidian distance obtained by comparing with stored Centroids.
  7. Cluster Number along with stored centroids form the compressed data for the given image.

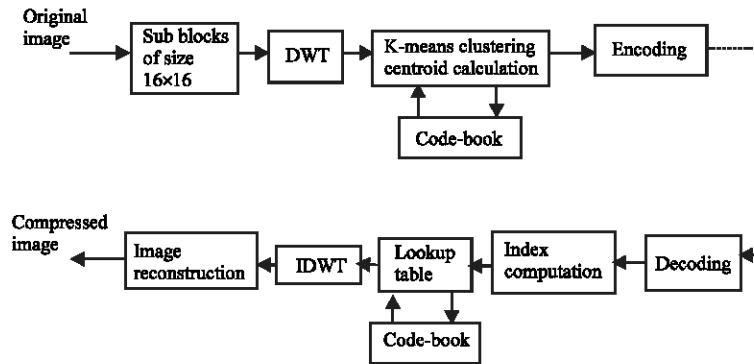


Fig. 3: Block diagram of the DWT based clustering for compression

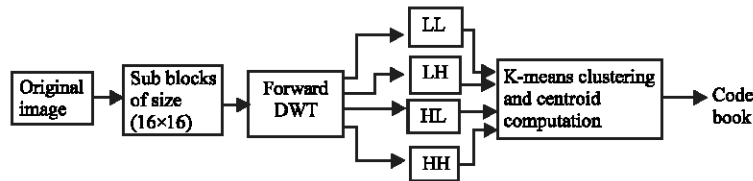


Fig. 4: Codebook generation

### EXPERIMENTAL RESULTS

For the simulation of the entire compression process, we used  $16 \times 16$  points image blocks (as a compromise between the compression rate and the correlation between successive blocks). The K-means clustering was used to obtain the centroids of each sub band. Simulations have been carried out on test images boat, pepper and Lena images of size  $512 \times 512$ . For each test image, objective and subjective picture quality are analyzed. The standard Lena image is used in this article for illustration purposes. The cluster size and hence the codebook size in our algorithm is used as the variable. Table 1 shows the evolution of the compression rate on the Lena image, when the cluster size is varied from 10 to 100. To keep an acceptable subjective quality of the image, the tests are made with our compression scheme with an initial codebook size of 10. The compressed images obtained as a function of the cluster size are shown in Fig. 5, wherein it is shown for cluster sizes 50 and 100, respectively. The computed PSNR indicates that the resulting images for various cluster size are uniform. Comparison of this work with the most prominent transformation coding techniques in the field, namely the JPEG compression (Wallace, 1991), indicates a better visual quality for higher cluster size retaining similar compression ratio. Extensive simulation was carried out by varying the number of cluster size for different image. Based on the simulation results, the following remarks are relevant.

- At low bit rates (below 0.5 bpp), degradation in picture quality (low PSNR) is observed due to the block based processing in wavelet domain.
- The compression ratio varies very little with the increase in codebook size beyond 60.
- The signal to noise ratio (PSNR) does not vary too much with the increase in codebook size. The simulation result indicates, that more or less the same PSNR is obtained for increase in codebook size. This is due to the fact that the reconstruction of the image is from the entire four-sub band code vector index, including the high frequency sub band, which predominantly contains only the high frequency information. However, the visual quality of the reconstructed image increases as the code vector size increase.

**Rate distortion optimization:** The Rate-Distortion theory describes the performance limit of lossy compression. In image compression, the objective is to use minimum number of bits needed in compressing the image data at a given distortion level. Generally, the smaller the number of bits, the larger is the distortion for the reconstructed image. When more and more bits are used for encoding, the quality of the reconstructed image will be gradually improved, at the cost of compression ratio. Thus, the optimization of compression ratio with quality is a key problem in image compression. In this study as a result of clustering each sub band is regarded as a source to be encoded and all the four sub bands are considered as significant for encoding. It is found from the simulation

Table 1: Comparison of rate of transmission with compression ratio

Size of the code book	10	20	30	40	50	60	70	80	90	100
Compression ratio	56.82	32.6	23.3	18.23	15.0	12.7	11.1	9.87	8.87	8.05
Bites pixel	0.15	0.25	0.34	0.44	0.53	0.62	0.71	0.81	0.90	0.99

Table 2: Bits per pixel with PSNR

Bites per poxel	0.15	0.25	0.34	0.44	0.53	0.62	0.71	0.81	0.90	0.99
PSNR in dB	18.19	21.05	22.01	20.76	23.16	22.22	25.28	25.96	25.09	25.70

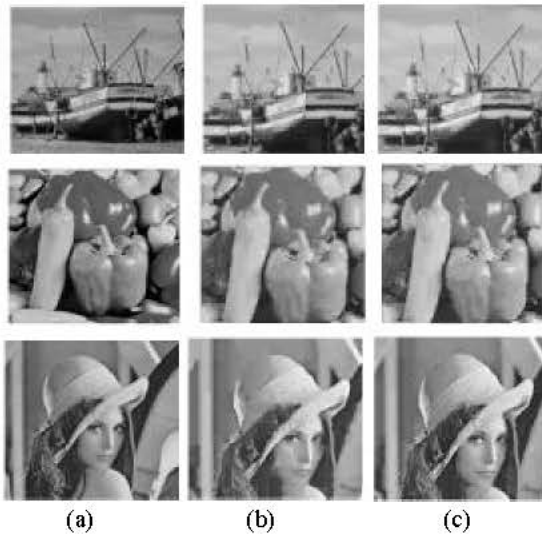


Fig. 5: Results of clustering based compression, Column (a): Original boat, pepper and lena image. Column (b): Compressed image with a cluster size of 50. Column(c): Compressed image with a cluster size of 100

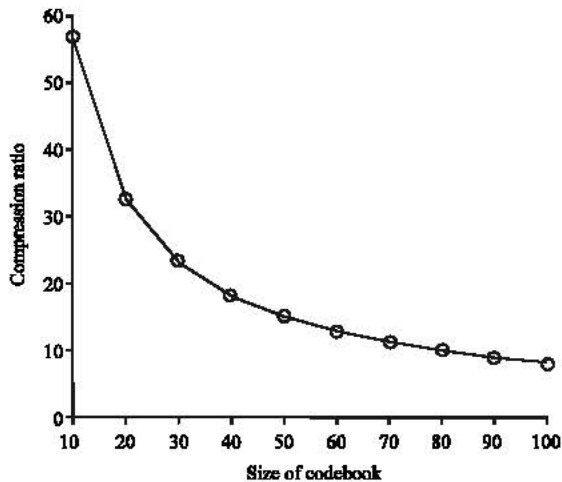


Fig. 6: Size of codebook vs. compression ratio

results that high compression ratio is obtained, for smaller values of cluster size. Table 1 shows the comparison of rate of transmission (bpp) with compression ratio. Figure 6 shows that the compression ratio is exponentially

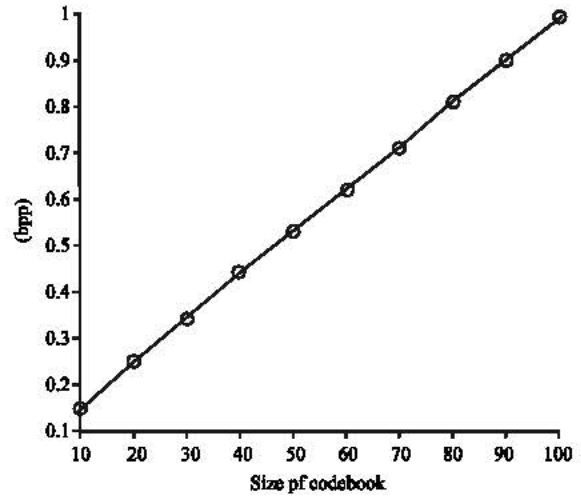


Fig. 7: Size of codebook vs. bpp

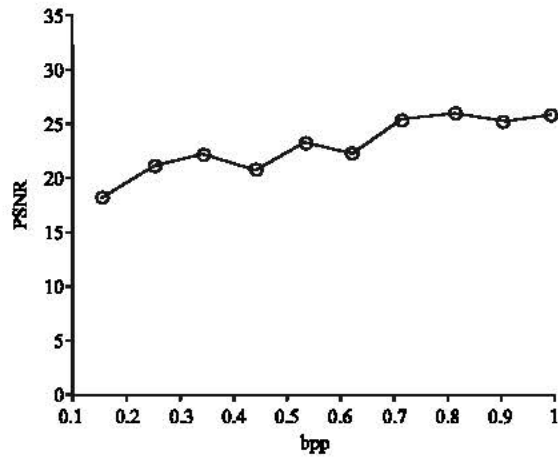


Fig. 8: Bits per pixel vs. PSNR

decreasing when the size of the codebook is increased. This is true for any compression scheme based on vector quantization.

Figure 7 shows that, the bit per pixel is a simple linear function of the size of the codebook. Also the simulation results indicate that the PSNR values remains more or less constant as the codebook size is varied. However the visual quality of the images is found to increase. Table 2 shows the variation of PSNR values of the compressed

'Lena' image for various compression ratios. Figure 8, shows the variation of the PSNR for various values of bpp. It is found that the PSNR values found to increase but not strictly, for higher values of bpp. It is attributed to the random initial selection of the centroids in the clustering mechanism.

### CONCLUSIONS

In this study, a new image compression algorithm in wavelet domain is presented. The proposed algorithm has three main features, which make it very efficient in image coding. First it takes advantage of the inter-sub band self-similarity of different sub band coefficients in the wavelet transform domain. Second, it uses the well-known K-means clustering algorithm to compute the centroid of each cluster. Finally, Vector quantization technique has been employed in order to achieve high compression rate. Simulation results indicate that the compression ratio and bpp obtained are comparable to other still image compression algorithms.

### REFERENCES

- Antonini, M. *et al.*, 1992. Image coding using wavelet transform. *IEEE Trans. Image Proces.*, 1: 205-220.
- Cosman, P.C. *et al.*, 1996. Vector quantization of image subbands: A survey. *IEEE Trans. Image Proces.*, 5: 202-225.
- DeVore, R.A. *et al.*, 1992. Image compression through wavelet transform coding. *IEEE Trans. Inform. Theory*, 38: 719-746.
- Jain, A.K. *et al.*, 1999. Data Clustering: A Review *ACM Computing Surveys*, 3: 264-324.
- Mallat, S., 1989. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 11: 674-693.
- Soman, K.P. *et al.*, 2005. *Insight In To Wavelets. From Theory To Practice*. PHI.
- Wallace, G.K., 1991. The JPEG still picture compression standard. *Comm. ACM.*, 34: 30-44.