

Side-Match Vector Quantizers Using Neural Network Based Variance Predictor for Image Coding

Shuangteng Zhang

*Department of Computer Science
Eastern Kentucky University
Richmond, KY 40475, U.S.A.*

shuangteng.zhang@eku.edu

Abstract

Side-match vector quantizer reduces bit-rates in image coding by using smaller-sized state codebooks generated from a master codebook through exploiting the correlations of neighboring vectors. This paper presents a new neural network based side-match vector quantization method for image coding. In this method, based on the variance of a vector which is predicted by a neural network, a subset of the codewords in the master codebook is selected for the side-matching to construct the state codebook for the encoding of the vector. This technique generates a lower encoding bit rate with a higher reconstructed image quality. Experimental results demonstrate that in terms of PSNR (Peak Signal-to-Noise Ratio) of the reconstructed images, the proposed method significantly outperforms the regular side-match vector quantizer, especially at lower coding bit-rates.

Keywords: Vector Quantization, Image Coding, Side Match, Neural Network.

1. INTRODUCTION

Image data compression is very important for various image and video processing applications which require reduced bit-rate/channel bandwidth. Such applications include digital television, video conferencing, telemedicine, multimedia, and remote sensing images from satellite and reconnaissance aircrafts, as well as the storage systems of multispectral image data from space programs, medical images, finger prints, and facial images.

There are many approaches to data compression. Among them, vector quantization (VQ) [1-3], which achieves data compression through mapping n-dimensional vectors onto a finite set of representative vectors called a codebook, is an important and heavily conducted research area. In VQ image compression [4], the images are usually partitioned into blocks with each block forming an n-dimensional vector. Each of these vectors is then coded as an index of its best matching vector in the codebook. The reconstructed images are obtained simply by selecting the corresponding codeword vectors from the codebook using the indices.

VQ image compression takes advantage of the correlation of the image pixels within a block (vector) for the coding bit-rate reduction and yields acceptable performance at low bit-rates. To further improve VQ's performance at low bit-rates, a side-match vector quantization method (SMVQ) [5] has been proposed. This method explores not only the redundancy within a vector but also the strong correlation between the neighboring vectors for high quality image coding at low bit-rates. In practice, SMVQ assumes the continuity of edges across neighboring image blocks' boundaries and generates a smaller-sized state codebook from the master codebook for each block's encoding. The selection of each state codebook is a subset of codewords in the master codebook which are the best matches of the upper and left blocks of the block to be encoded. In other words, SMVQ reduces the coding bit-rates by predicting the current block using its upper and left boundary blocks. Therefore, SMVQ's performance relies on the accuracy of the prediction. Inaccurate prediction, which occurs often when the block is within an edge area, may result in reduced quality of reconstructed images.

To increase the accuracy of the prediction and therefore further improve the performance of SMVQ, various SMVQ schemes have been developed. Chang and Chen [6] proposed variable-rate side-match finite-state vector quantization with a block classifier. Wei *et al.* [7] reported a three-sided side match technique which uses not only the upper and left sides, but one of the bottom and right sides for the prediction. Yang and Tseng [8] developed a smooth side-match classified vector quantizer which selects state codebook according to the smoothness of the gray levels between neighboring blocks. Chang proposed a gradient match quantization method [9] using gradient match error for the selection of the codewords in the state codebook. The performance of the gradient side match vector quantization method was further improved by combining the non-iterative fractal block coding technique [10,11]. Some other techniques have also been reported for the enhancement of the original SMVQ method. Examples include pattern-based side match VQ [12], side match VQ using gradient based classifier [13], smooth side match weighted method [14] and side match vector quantizers with variable rates according to both the coding quality and the bit rates [15].

In this paper, a neural network based side-match vector quantization method for image coding is presented. In this method, a neural network is designed to predict the variances of the vectors, which are in turn used to select the codewords in the master codebook for the generation of state codebooks for the encoding of the vectors. This technique generates a lower encoding bit rate with a higher reconstructed image quality. Experimental results demonstrate that in terms of PSNR (Peak Signal-to-Noise Ratio) of the reconstructed images, the proposed method significantly outperforms the regular side-match vector quantizer at similar low coding bit-rates.

2. SMVQ FOR IMAGE CODING

As mentioned in previous section, SMVQ takes advantage of both the redundancy within a block and strong correlation between the neighboring blocks for high quality image coding at low bit-rates. The original SMVQ encodes each image block by a smaller-sized state codebook generated from a master codebook using a side-match selection function as shown in Figure 1.

Assume that the master codebook has N codewords with each codeword an $m \times n$ vector denoting by $C_i, i=1, 2, \dots, N$. Also assume that the image to be encoded is partitioned into blocks of size $m \times n$. SMVQ encodes the image blocks in an order from left to right and top to bottom. For each block being encoded, SMVQ uses the side information of its upper and left neighboring blocks to produce the state codebook. The block is encoded as the index of the codeword in the state codebook which is the best match to the block.

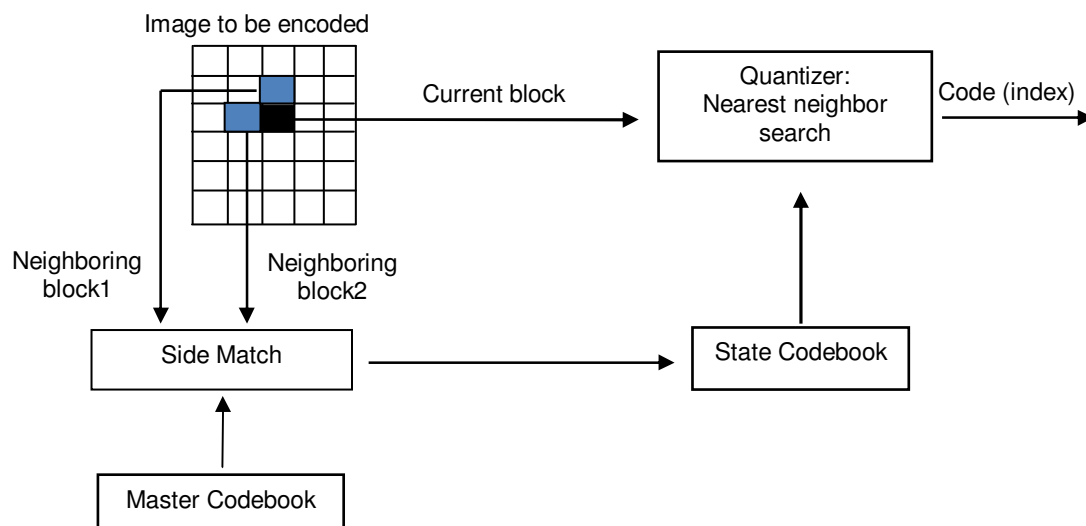


FIGURE 1: Block diagram of SMVQ encoder

Let $x(i, j)$ ($i=1, 2, \dots, m$ and $j=1, 2, \dots, n$) be the pixel in the current image block \mathbf{X} , and $u(i, j)$ ($i=1, 2, \dots, m$ and $j=1, 2, \dots, n$) and $l(i, j)$ ($i=1, 2, \dots, m$ and $j=1, 2, \dots, n$) be its upper block \mathbf{U} and left block \mathbf{L} , respectively. The state codebook is generated and the block is encoded according the following steps:

- (1) For each of the codewords \mathbf{C}_i in the master codebook, calculate the side-match error e_i ,

$$e_i = \sum_{k=1}^n (u(m, k) - c_i(1, k))^2 + \sum_{k=1}^m (l(k, n) - c_i(k, 1))^2 \quad (1)$$

- (2) Select M ($M \leq N$) codewords in the master codebook with the smallest side-match errors as the state codebook codewords \mathbf{S}_i , $i=1, 2, \dots, M$,
- (3) For each of the codewords \mathbf{S}_i in the state codebook, calculate its distortion measure $d_i(\mathbf{S}_i, \mathbf{X})$,

$$d_i(\mathbf{S}_i, \mathbf{X}) = \sum_{k=1}^m \sum_{r=1}^n (s_i(k, r) - x(k, r))^2 \quad (2)$$

- (4) The image block \mathbf{X} is encoded as the index j of the codeword \mathbf{S}_j which produces the smallest distortion $d_j(\mathbf{S}_j, \mathbf{X})$.

For the decoding of each image block, SMVQ first generates the state codebook which was used to encode it according to step (1) and (2) in the encoding process. Once the state codebook is generated, the reconstructed block is obtained by simply selecting the corresponding codeword in the state codebook using the index which is the code of the block generated in the encoding process. The block diagram of SMVQ decoder is shown in Figure 2.

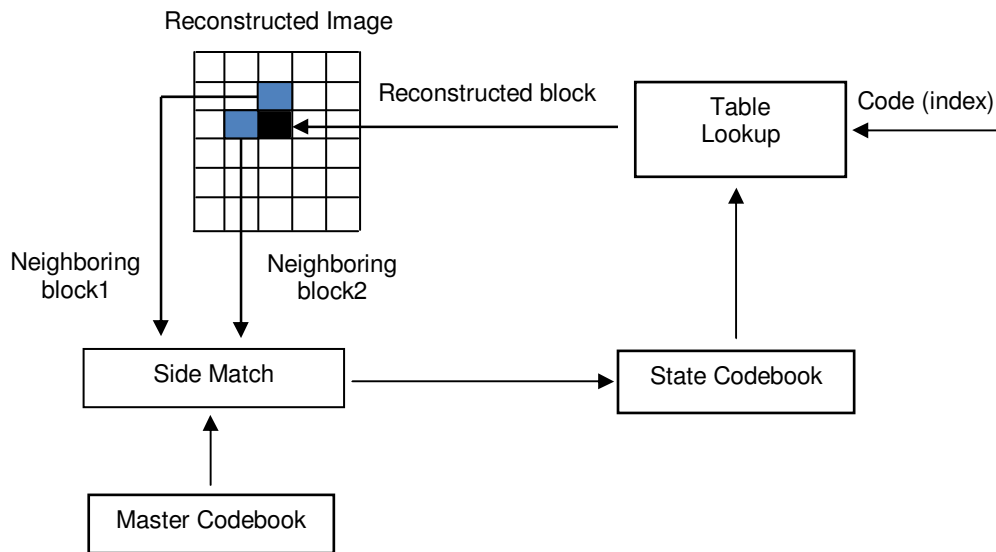


FIGURE 2: Block diagram of SMVQ decoder

3. PROPOSED NEURAL NETWORK BASED SMVQ

3.1 Image Block Variance for SMVQ

SMVQ image coding achieves low bit-rates through using smaller-sized state codebooks generated from master codebook for the coding of the image blocks. The quality of the reconstructed images coded at low bit-rates relies on the accuracy of the prediction of the blocks. An accurate prediction of a block may keep the codeword, which is the best match of the block when a full search in the master codebook is conducted, in the state codebook generated by the side match function. The more accurate prediction makes it possible to select a smaller-sized state codebook for the coding so that the image can be coded at a lower bit-rate with a quality

similar to the one obtained when the master codebook is used. However, the prediction may be inaccurate, especially when the block is within an edge area. An inaccurate prediction on the other hand may keep the codeword which is the best match of the block in a full search out of the generated state codebook when a small size is chosen, and therefore results in a degraded reconstructed image quality.

Since edge and non-edge blocks may have different variances with an edge block highly likely to have a higher variance, the best matched codeword for each block in a full search may scatter around the codeword in the master codebook whose variance is equal to or close to the variance of the block to be encoded, assume that the master codebook is sorted according to the variances of the codewords. Figure 3 shows the distribution of the best matched codewords in full search for the encoding of image Lena using a codebook of size 256 which is sorted by variance of the codewords. The x-axis is the difference value of the index of codeword whose variance is equal or close to the variance of the block being encoded and the index of its best matched codeword in a full search, and the y-axis is the number of image blocks associated with the same index difference value.

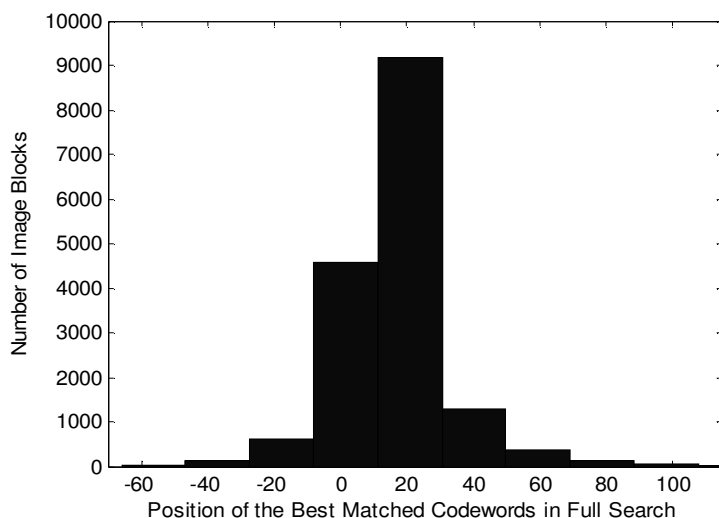


FIGURE 3: Distribution of the best matched codewords in full search

From Figure 3, it can be seen that more than 90% of the best matched codewords are distributed in the range from 6 codewords before to 54 codewords after the codeword whose variance is equal to or close to the variance of the block to be coded. In other words, if the 60 codewords around the codeword whose variance matches the variance of the block being coded are used to be side-matched to generate the state codebook, then there is a much higher possibility that the best matched codeword will be in the state codebook even if the size of the state codebook is selected to be smaller than that selected in the regular SMVQ. This will result in a faster processing speed since a subset of codewords in the master codebook instead of the whole is used for side matching. Furthermore, the coding bit rate can be further lowered due to the fact that smaller size of state codebooks can be used while preserving similar reconstructed image quality.

Based on the above observation, in our proposed SMVQ method, the variances of the image blocks, which are usually sensitive to image edges, are combined into the side matching and state codebook generating process for more accurate prediction and therefore lowered coding bit-rate with preserved image quality.

3.2 Neural Network Based Variance Predictor

In the proposed method, the variance of the block being coded is used to select a subset of the codewords in the master codebook for the side matching and state codebook generation. To implement the process, the variance of the block must be known. In the proposed algorithm, the variance is predicted by a three-layered feed-forward neural network [16] as shown in Figure 4.

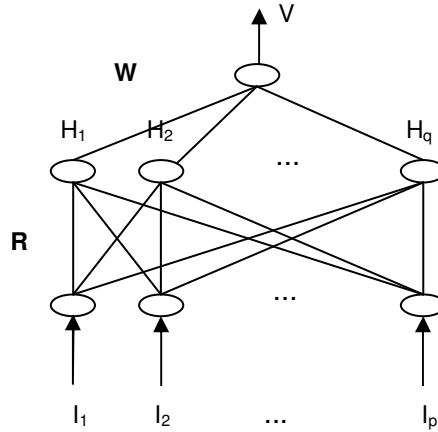


FIGURE 4: Neural network structure of the variance predictor

This neural network contains an input layer with p external inputs, a hidden layer with q neurons and an output layer with one neuron. The external inputs of the network consist of the mean value k of the pixels in the last two rows of the upper block and last two columns of the left block of the current block and the difference values z_i ($i=1, 2, \dots, p-1$) between the mean k and each pixel value in those rows and columns, denoted as $I=(I_1, I_2, \dots, I_p)=(k, z_1, z_2, \dots, z_{p-1})$, where $p = 2*(m+n)+1$, k is calculated as,

$$k = \frac{1}{2*(m+n)} \left(\sum_{i=m-1}^m \sum_{j=1}^n u(i, j) + \sum_{i=1}^m \sum_{j=n-1}^n l(i, j) \right) \tag{3}$$

The output of the network is the predicted variance value of the current block, which is formulated as,

$$V = f \left(\sum_{j=1}^q w_j f \left(\sum_{i=1}^p r_{ij} I_i \right) \right) \tag{4}$$

where V is the output of the network, I_i is the i -th element of the input vector I , r_{ij} is the weight of the connection between the i -th neuron in the input layer and the j -th neuron in the hidden layer, w_j is the connection weight between the j -th neuron in the hidden layer and the neuron in the output layer, and $f(\cdot)$ is the activation function of the neurons, which is

$$f(x) = \frac{2}{1 + e^{-\lambda x}} - 1 \tag{5}$$

where $\lambda > 0$ is the neuron activation function coefficient determining the steepness of the function. The network is trained using back-propagation learning algorithm [16] with sample images. After trained, it can be used to predict the variance of each block being encoded.

3.3 The Encoding and Decoding Algorithms

Given the information presented in the previous sections, the proposed neural network based SMVQ encoder and decoder algorithms are described in this section. The encoder consists of four elements including neural network variance predictor, candidate codeword selector, side

match function, and quantizer as shown in Figure 5. The encoder algorithm can be implemented as follows:

- (1) Sort the master codebook by the variances of the codewords in an ascending order,
- (2) Use the neural network variance predictor to predict the variance of the current block,
- (3) Select T (predefined) ($T < N$) codewords in the master codebook around the codeword whose variance is equal to or is closest to the predicted variance value,
- (4) For each of the codewords C_i selected in step (3), calculate the side-match error e_i using equation (1),
- (5) Select M ($M \leq T$) codewords with the smallest side-match errors from the T codewords selected in step (3) as the state codebook codewords $S_i, i=1, 2, \dots, M$,
- (6) For each of the codewords S_i in the state codebook, calculate its distortion measure $d_i(S_i, X)$ using equation (2),
- (7) If the smallest distortion $d_i(S_i, X)$ is greater than a predefined threshold, the block is coded using full search. Otherwise, it is encoded as the index j of the codeword S_j which produces the smallest distortion $d_j(S_j, X)$,
- (8) Repeat step (2) – (7) for next block encoding.

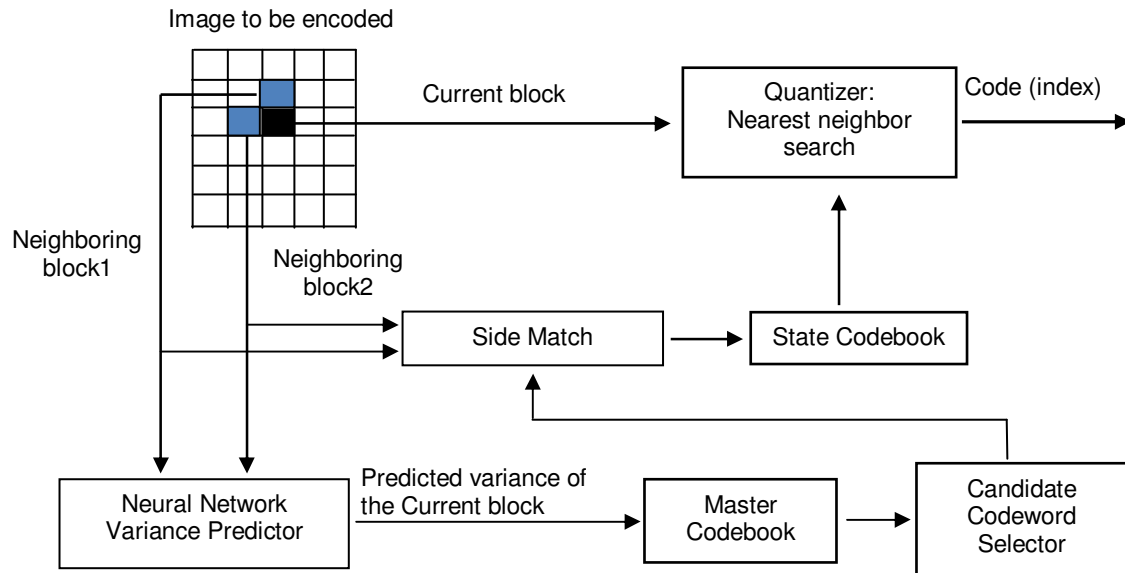


FIGURE 5: Block diagram of the proposed neural network based SMVQ encoder

The structure of the decoder is same as that of the encoder except that the quantizer element in the encoder is replaced by the table lookup element. The block diagram of the decoder is shown as in Figure 6. For the decoding of each image block, the coder first generates the state codebook which was used to encode it according to step (2) and (5) in the encoding process. Once the state codebook is generated, the reconstructed block is obtained by simply selecting the corresponding codeword in the state codebook using the index which is the code of the block generated in the encoding process.

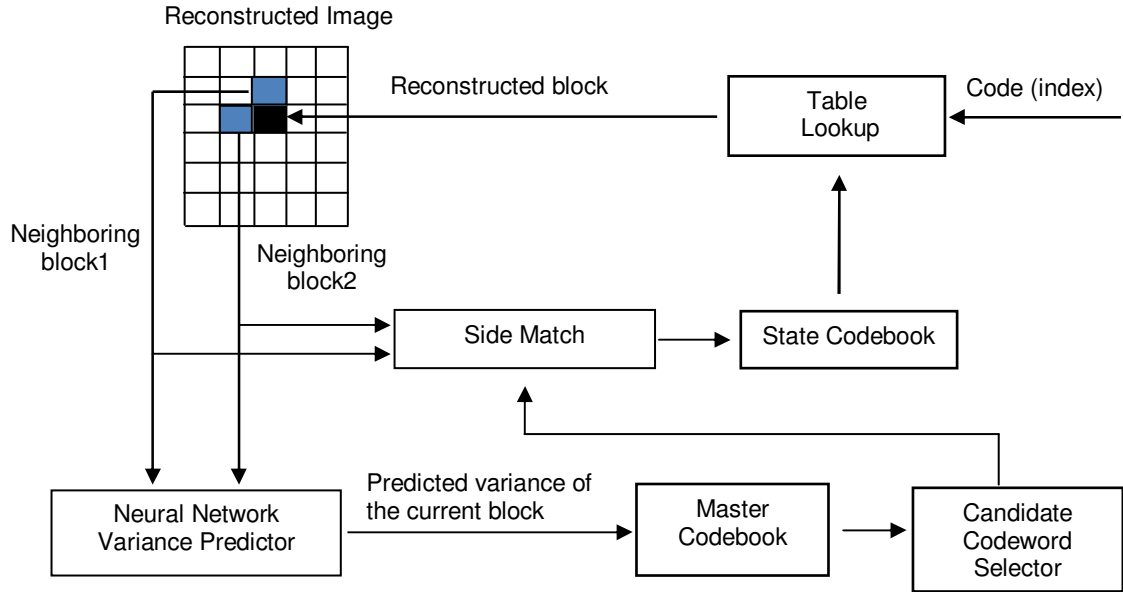


FIGURE 6: Block diagram of the proposed neural network based SMVQ decoder

4. EXPERIMENTS AND DISCUSSION

In this section, the performance of the proposed algorithm is evaluated by experiments. The peak signal-to-noise ratio (PSNR) is used as a quantitative measure of the quality of the reconstructed images. PSNR can be calculated as,

$$PSNR = 10 * \log_{10} \frac{255^2}{\frac{1}{B * B} \sum_{i=1}^B \sum_{j=1}^B (\hat{z}(i, j) - z(i, j))^2} \quad (6)$$

where $\hat{z}(i, j)$ and $z(i, j)$ are the (i, j) th pixels in the reconstructed and original images, respectively, and the images are of size $B \times B$.

In our experiments, the master codebook of size 256 was generated using LBG algorithm [17] and the neural network based variance predictor was trained using 8-bit gray level image Lena of size 512×512 . The trained neural network predictor and coding algorithm were then applied to code the image Lena as well as some other gray images of different objects at various bit rates. For comparison purpose, the corresponding images were also coded by using the regular SMVQ method.

Table 1 shows the PSNRs of the reconstructed images coded around bit rate 0.24bpp (bit per pixel), 0.31bpp and 0.37bpp by using the regular SMVQ and the proposed method. This table indicates that the proposed algorithm improves the PSNR of the reconstructed image over the regular SMVQ average by 1.66dB, 1.06dB and 0.42dB when bit rate is around 0.24bpp, 0.32bpp, and 0.37bpp, respectively. These results demonstrate that in terms of PSNR, the proposed algorithm significantly outperforms the regular SMVQ method, especially when the coding bit rate is low.

	Bit rate: around 0.24bpp		Bit rate: around 0.31bpp		Bit rate: around 0.37bpp	
	SMVQ	Proposed Method	SMVQ	Proposed Method	SMVQ	Proposed Method
Lena	28.70	31.17	30.27	31.83	31.38	31.93
Peppers	27.87	28.86	28.66	29.20	29.06	29.25
Elaine	28.32	29.35	29.14	29.82	29.60	29.86
Airplane	26.50	28.19	27.34	28.43	28.06	28.46
Flowers	30.31	31.75	31.67	32.53	32.38	32.65
Kids	26.15	28.51	27.32	28.95	28.18	29.02
Average	27.98	29.64	29.07	30.13	29.78	30.20

TABLE 1: PSNRs (dB) comparison (Coding Bit Rate around 0.24bpp, 0.31bpp and 0.37bpp)

Figure 7 shows the curves of PSNRs as a function of coding bit-rate for the proposed and regular SMVQ methods. From the figure, it can be observed that from low to high bit-rates, the PSNRs of the images coded by using the proposed method approach much faster than the regular SMVQ method to a constant value, which is the highest PSNR value SMVQ methods can reach and can be obtained when the images are coded using the master codebook. This also confirms that the proposed method has a more accurate prediction of the block being coded and therefore allows to use smaller state codebooks which generate lower coding bit-rates with higher images quality.

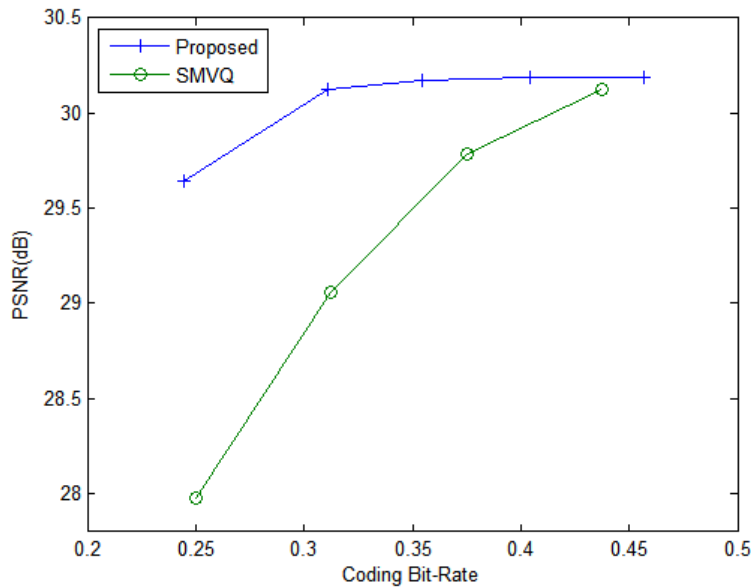


FIGURE 7: PSNR as a function of coding bit-rate

The visual quality of the reconstructed images is also examined. Figure 8 shows the original and corresponding Lena images coded by master codebook at 0.5bpp, the proposed method at 0.30bpp and the regular SMVQ method at 0.31bpp. Magnified face portions of the original and the reconstructed Lena images are shown in Figure 9 for further comparison. These figures show that the images coded by the proposed method are at the quality similar to those coded by the master codebook and they are sharper and visually look much more pleasant than the ones

coded by the regular SMVQ method at the similar coding bit-rate. This is consistent with the PSNR results shown in Table I.



FIGURE 8: (a) Original Lena image; (b) Lena image coded by using the master codebook at 0.5bpp; (c) Lena image coded by using the proposed method at 0.3bpp; (d) Lena image coded by using the regular SMVQ at 0.31bpp

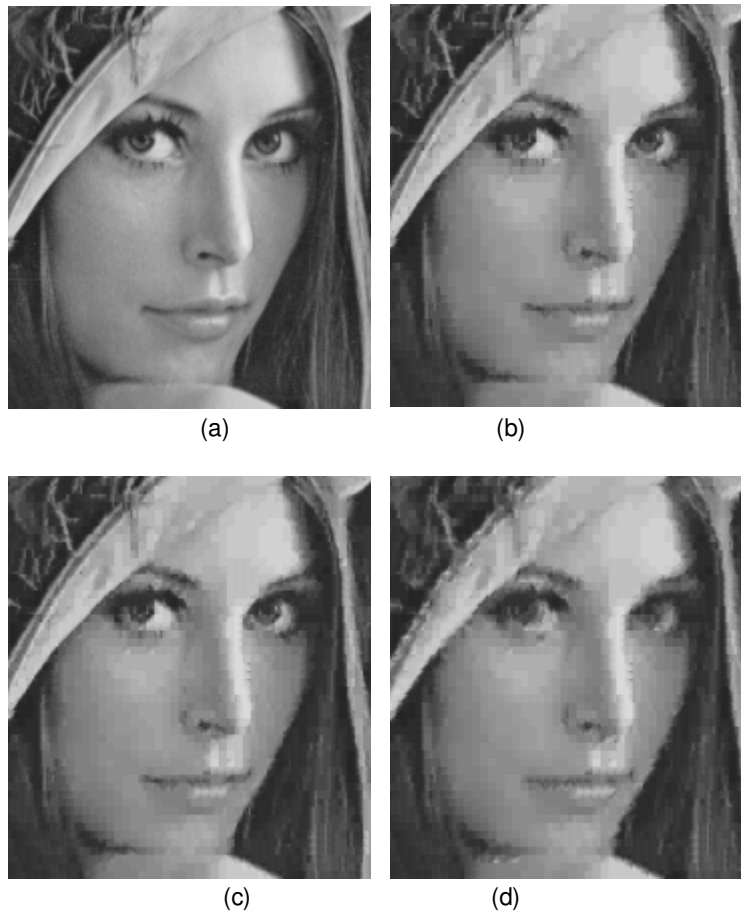


FIGURE 9: Magnified face portion from (a) Original Lena image; (b) Lena image coded by using the master codebook at 0.5bpp; (c) Lena image coded by using the proposed method at 0.3bpp; (d) Lena image coded by using the regular SMVQ at 0.31bpp

5. CONCLUSION

We have presented a new side-match vector quantization method for image coding using a neural network-based variance predictor. In this method, a neural network is used to predict the variances of the image blocks. The predicted variances are in turn used to select a subset of the codewords in the master codebook for the side matching to generate the state codebooks for the encoding of the image blocks. With the involvement of the image block variances, the regular SMVQ is enhanced with more accurate prediction of the current block by using side information and therefore generates smaller state codebooks which encode the images at a lower bit rate with higher image quality. Experimental results demonstrate that in terms of PSNR (Peak Signal-to-Noise Ratio) of the reconstructed images, the proposed method significantly outperforms the regular side-match vector quantizer at similar low coding bit-rates.

6. REFERENCES

- [1] R. M. Gray, "Vector quantization", IEEE ASSP Magazine 1, pp. 4-29, 1984.
- [2] A. Gersho, R. M. Gray, Vector Quantization and Signal, Compression, Kluwer Academic Publishers, 1992.
- [3] N. M. Nasrabadi, R. A. King, "Image coding using vector quantization: a review", IEEE Tran. Communications, vol. 36, no. 8, pp. 957-971, 1988.
- [4] K. Sayood, Introduction to data compression, Morgan Kaufmann Publishers, San Francisco, CA 1996.

- [5] T. Kim, "Side match and overlap match vector quantizers for images", IEEE Trans. Image Process., vol. 1, no. 2, pp. 170 -185, 1992.
- [6] R. F. Chang and W. -T. Chen, "Image coding using variable-rate side-match finite-state vector quantization", IEEE Tran. Image Processing, vol. 2, no. 1, pp. 104-108, 1993.
- [7] H. Wei, P. Tsai and J. Wang, "Three-sided side match vector quantization", IEEE Trans. Circuits and Systems for Video Technology, vol. 10, no. 1, pp. 51–58, 2000.
- [8] S. B. Yang and L. Y. Tseng, "Smooth side-match classified vector quantizer with variable block size", IEEE Tran. Image Processing, vol. 10, no. 5, pp. 677-685, 2001.
- [9] H. T. Chang, "Gradient match vector quantizers for images", Opt. Eng., vol. 39, no. 8, pp.2046-2057, 2000.
- [10] H. T. Chang, "Gradient match and side match fractal vector quantizers for images", IEEE Trans. Image Process., vol. 11, no. 1, pp. 1-9, 2002.
- [11] H. T. Chang and C. J. Kuo, "Iteration-free fractal image coding based on efficient domain pool design", IEEE Tran. Image Processing, vol. 9, pp.329-339, 2000.
- [12] C. C. Chang, F. C. Shiue, T. S. Chen, "Pattern-based side match vector quantization for image compression", Imaging Science Journal, vol. 48, no. 2, pp. 63-76, 2000.
- [13] Z. M. LU, B. Yang, S. H. SUN, "Image compression algorithms based on side-match vector quantizer with gradient-based classifiers", IEICE TRAN. Information and Systems, vol. E85-D, no.9, pp.1409-1415, 2002.
- [14] S. B. Yang, "Smooth side-match weighted vector quantiser with variable block size for image coding", IEE Proc. Vis. Image Signal Processing, vol. 152, no. 6, pp. 763-770, 2005.
- [15] S. B. Yang, "New variable-rate finite state vector quantizer for image coding", Opt. Eng., vol. 44, no. 6, 067001, 2005.
- [16] M. H. Hassoun, Fundamentals of Artificial Neural Network, MIT Press, Cambridge, MA, 1995.
- [17] Y. Linde, A. Buzo and R. M. Gray, "An algorithm for vector quantization design", IEEE Trans. Communications, vol. 28, pp. 84-95, 1980.