A Side Match Reconstruction Method Using Tree-Structured VQ for Transmitting Images Progressively

Kuo-Lung Hung and Chin-Chen Chang
Department of Computer Science and Information
Engineering, National Chung Cheng University,
Chaiyi, Taiwan 621, R.O.C.
E-mail: {jackhong,ccc}@cs.ccu.edu.tw

Tung-Shou Chen
Department of Computer Science and Information
Management, Providence University, Taichung,
Taiwan 433, R.O.C.
E-mail: tchen@pu.edu.tw

Abstract

In this paper, a new progressive image transmission (PIT) method is proposed. It is called the side-match reconstruction method using TSVQ (SMTSVQ). The proposed method is basically an improvement version of TSVQ. In order to take full advantage of the high correlation between adjacent pixels, half of the image blocks are transmitted interleavedly in each phase, and each block sends two bits. As for the receiver, a reconstruction method of side match technique is proposed to rebuild the unreceived blocks. From the experimental results, we see that the image quality of our method is better than those of the other previous works: SPITM, FRM and TSVQ. Especially at the first phase, the difference of PSNR between SMTSVQ and TSVQ is more than 3.7dB. Besides, the reconstruction time of SMTSVQ is much less than that of SPITM and is very close to those of FRM and TSVQ. SMTSVQ is therefore efficient and effective in PIT.

Keywords: side match, tree-structured VQ, progressive image transmission

1. Introduction

Images are one of the most important media among various information representations, for it can convey more information than the others. However, digital representations of images require huge numbers of bits. This means a large capacity of storage memory is necessary for images. Also, such large memory sizes of images result in long transmission time, especially in low capacity transmission channels. To alleviate this trouble, progressive image transmission (PIT) techniques [2-5] are used to send images to the receiver through many stages. The received image in PIT is changed from fuzzy to clear stage by stage. At first, the receiver is given only a few bits to achieve a rough recognizable estimate of the image; as soon as the receiver is given further bits, it enhances the perceived quality of the image by adding more details. When the received image is clear enough, the user can decide to stop transmitting. In the way, PIT can keep down the transmission bits of an image.

There are several important requirements we put upon PIT. First, it should have low bit rates and high quality at the beginning phases. Next, a later phase should reuse the data of its preceding phases. Besides, the total transmission bits of PIT should be less than those the original images had. Finally, the encoding and decoding time should not be too long. Among them, the first one is the most important criterion for a PIT algorithm, especially when we can choose to stop transmitting halfway.

Tzou classified PIT methods into three categories [5], which were the spatial domain methods, the transform methods, and the pyramid-structured progressive transmission methods. The most intuitive method in spatial domain is the bit-plane method (BPM). Yet, too high being the transmission load is the major disadvantage of BPM.

In order to push forward the drawback of BPM, the selective progressive image transmission method (SPITM) [4] and the fast reconstruction method for progressive image transmission (FRM) [2] have been proposed. Both of them are based on vector quantization (VQ). SPITM combines some blocks of indices to be a sub-image, chooses one block for each sub-image according to the diagonal sampling technique, and sends the indices of the selected blocks to the receiver in each phase. At the receiver, the side-match method is employed to predict the unreceived blocks. Note that the final image of SPITM is not the original image since VQ is a lossy image compression. However, the transmission bits are reduced. The major drawback of SPITM is that the time complexity is heavy. As for the other method, FRM is an improvement of SPITM. The differences between them are that FRM uses the central sampling techniques to select the indices of the compressed image and that it employs the pixels copy technique to replace the side-match method. The former technique can improve the image quality quickly at the beginning phases. On the other hand, by the latter technique, the reconstruction time of FRM is shorter than that of SPITM.

The most commonly used method so far for reducing search time of VQ is the tree-structured vector quantizer (TSVQ) [1]. TSVQ has the codewords arranged in a tree structure, thus achieving the search in a logarithmic time at a cost of a slight loss in image quality. Moreover, it is amenable to PIT due to its built-in feature of successive approximation. In [5], Tzou describes TSVQ's suitability for progressive transmission. Due to the successive approximation nature of TSVQ, the farther down the tree the vector is encoded, the better the reproduction will be. The original codeword indices can be reassigned to new indices according to the search paths. TSVQ then sends one bit plane of the new codeword indices from the MSB to the LSB in each phase.

This TSVQ method for progressive transmission is natural, and its decoding time is less. However, the transmission quality is not very good, especially at the beginning phases. This is because TSVQ does not take advantage of the feature that in natural images we can usually find high correlation between adjacent pixels. Based on this correlation, a new PIT method is proposed in this paper. It is called the side-match reconstruction method using TSVQ (SMTSVQ). Our method builds up a full-search tree as the TSVQ method does, and yet in each phase, half of the blocks are transmitted interleavedly, and, moreover, each block sends two bits. Only when the receiver gets the bits of the blocks, it then fills up these blocks according to the codewords of TSVQ. As for the unreceived blocks, the side match reconstruction method tries to rebuild them. Since our method takes full advantage of the spatial contiguity and interpixel correlation of image data, the results show significantly better quality than those of the above methods.

The remainder of the paper is organized as follows. Section 2 introduces the related work TSVQ. Section 3 explains the details of our new proposed method. Experimental results are presented in Section 4. The conclusions are drawn in Section 5.

2. Tree-Structured Vector Quantizer (TSVQ)

The most popular of all structured VQs for reducing search complexity is the tree-structured vector quantizer (TSVQ). TSVQ has the codewords arranged in a tree structure called the codebook tree. A codebook tree is constructed with a codeword at each node of the tree. Only the codewords in the leaves are used for encoding an input image vector. For a codebook searching, TSVQ first starts from the root node and finds out the child node whose codeword has the minimum distortion between the input image vector and its children. Next, TSVQ proceeds to the next stage and repeats the previous step until the terminal node stage is reached. Thus the search of the TSVQ method is in a logarithmic time at a cost of a slight loss in image quality.

In addition, TSVQ can also act as a progressive image transmission method. We can assign a binary value to each edge in the codebook tree, where the edge to the left child is labeled '0' and the edge to the right child is labeled '1'. The tree is then called the progressive transmission tree [7]. Figure 1 shows an example of a progressive transmission tree with the codebook size being 15, where the codewords of the tree include the intermediate nodes and the leaf nodes. When encoding an image block, TSVQ uses the labeled values in the path from root to the codeword in the progressive transmission tree to be the codeword index. TSVQ then sends one bit of the index for each codeword from the MSB to the LSB in each phase.

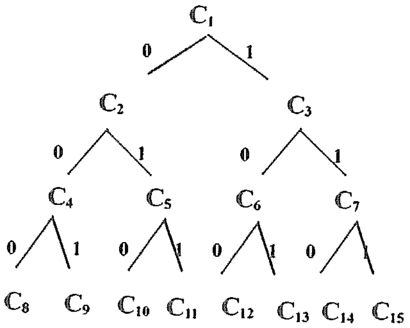


Figure 1: An example of the progressive transmission tree with codebook size 15

In the receiver, for each image block x, it is reconstructed according to the codeword in the corresponding level of the received bits of x in each phase. TSVQ traces the nodes from the root node to the corresponding level according to the following condition. If the value of the corresponding bit of x equals 0, then it is assigned to the left child; else to the right child. When TSVQ reaches the level in each phase, the image block is then replaced with the value of the codeword in the located node.

3. The Proposed Method SMTSVQ

In this section, a new efficient progressive image transmission method (SMTSVQ) is proposed which employs the TSVQ structure and takes full advantage of the high correlation feature between adjacent pixels. SMTSVQ transmits partial information of half of the image blocks in each phase only. As for the untransmitted blocks, a reconstruction method using the side match technique in SMTSVQ is also proposed to rebuild them. As we shall see, the TSVQ structure and the side match technique bring out the best in each other, and their combination makes the basis of the good performance of our proposed method. In this paper, for the sake of clearness, this section is divided into two sub-sections: the sender and the receiver subsections.

3.1 Sender

As what happens to the TSVQ method, a progressive transmission tree is first constructed in SMTSVQ. Next, the image is divided into interleaved blocks as shown in Figure 2. In each phase, only half of the blocks, namely the shadowed blocks in the figure, are taken into consideration. SMTSVQ encodes these blocks and transmits two bits of the index of each block from the sender to the receiver. The depth of the codebook tree is denoted as n; namely, the codebook size is $2\times2^n-1$. Thus, in our method, there are n phases in the whole progressive transmission process. We break the whole process into two parts, where Part One is from the first phase to the n/2-th phase, and Part Two is from the (n/2+1)-th phase to the n-th phase.

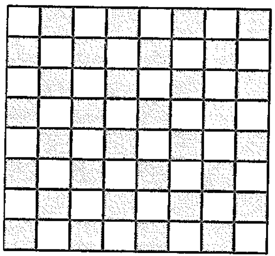


Figure 2: The image is divided into interleaved blocks in SMTSVQ

In Part One of the SMTSVQ method, only the shadowed blocks of Figure 2 are encoded by the ordinary TSVQ method. The unshadowed blocks are neglected by the encoder. Figure 3 shows the first phase of SMTSVQ. In Figure 3(b), the symbols '2' in the shadowed blocks mean that the preceding two bits of the index of each encoded shadowed block are transmitted to the receiver.

The remainder phases of Part One of SMTSVQ are analogous to the first phase. In the next phase, for example, the sender sends the next two bits to the shadowed blocks. In the same way, SMTSVQ transmits the whole index of the shadowed block. Figure 4 shows an example of four phases of Part One in SMTSVQ with the codebook size being 511.

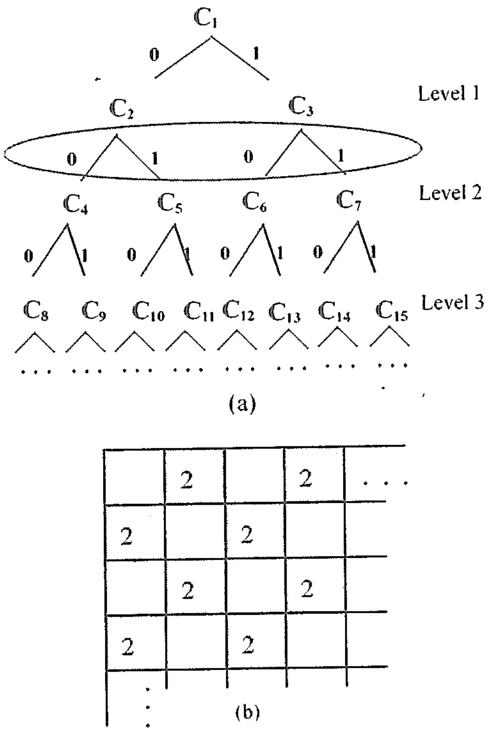
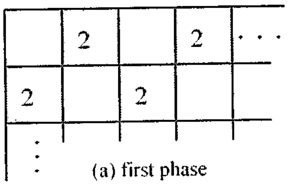
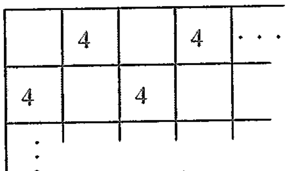


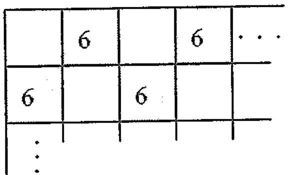
Figure 3: The first phase of SMTSVQ



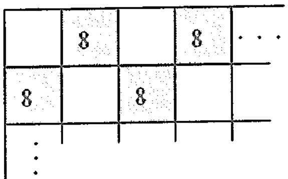
(a) first phase



(b) second phase



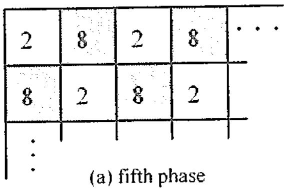
(c) third phase



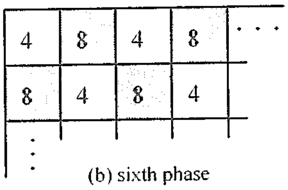
(d) fourth phase

Figure 4: The four phases of Part One in SMTSVQ with codebook size 511

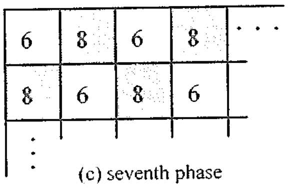
In the second part of SMTSVQ, since all bits of the shadowed blocks have been transferred, the only thing we can do is to send the encoding bits for the unshadowed blocks. As what is done to the shadowed blocks, two bits are transmitted for each unshadowed block in each phase. As we shall show in the receiver section, for an unshadowed block, these two bits can improve the image quality of its reconstruction. Moreover, if the more bits are transmitted for the block when the phases go on, the reconstructed image quality will have a lasting improvement. Figure 5 shows the four phases of the second part in the SMTSVQ method with the codebook size being 511.



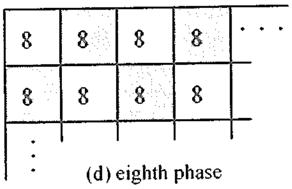
(a) fifth phase



(b) sixth phase



(c) seventh phase



(d) eighth phase

Figure 5: The four phases of Part Two in SMTSVQ with codebook size 511

3.2 Receiver

The receiver of SMTSVQ also has two parts. The phases in these two parts are the same as those in the sender section. At the first phase, since there are two bits received for the shadowed blocks, the decoding of the shadowed blocks are done according to the codewords at level 2 of the progressive transmission tree. As for the unshadowed blocks, they are reconstructed by the side match technique. The reconstruction steps of a block x are as follows. SMTSVQ traces the nodes from the root node to the corresponding level according to the following condition. If the side match distortions [6] between x and the left child are larger than those between x and the right child, then the left child has it, else the right child does. After SMTSVQ reaches the corresponding level in each phase, the block is then predicted with the value of the codeword in the located node. Algorithm 1 shows the entire decoding algorithm of Part One.

Algorithm 1: [The decoding algorithm of the Part One in SMTSVQ]

```
Step 1:  Let the current block to be decoded be x, and the number of the bits received be s.
Step 2:  If x is a shadowed block, then perform Step 3; else perform Step 5.
Step 3   /* Search the nodes of the codebook tree using TSVQ method; here we let root node be in level 0 */
         For each level L, where $L \le s$, repeat the following.
            If the value of the L-th bit of the received index of x equals 0, then locate it at the left child, else locate it at the right child.
Step 4:  Replace x with the codeword in the current leaf node, and then end the algorithm.
Step 5   /*Search the node of the codebook tree using the side match method; here we let root node be in level 0*/
         For each level L, where $L \le s$, repeat the following.
            If the side match distortions between x and the left child are larger than those between x and the right child, then locate it at the left child, else locate it at the right child.
Step 6:  Use the codeword in the current leaf node to predict x.
```

For each unshadowed block, the decoding processes of Part Two are different. In the fifth phase of SMTSVQ, Figure 5(a) shows that SMTSVQ receives two bits for an unshadowed block. As is mentioned earlier, these two bits are useful for the reconstruction of this block. We can use these two bits to find the precise node in the progressive transmission tree in advance. After that, we use the side match technique to search the codewords from the pre-located node to the leaf nodes. This action can improve the representativeness of the searched codeword. Moreover, the more precise the pre-located node is, the more representative the searched codeword is. That is, if the more bits are transmitted for each unshadowed block, the reconstructed image quality will be better. Thus, the image quality has lasting improvement when the phases go on. At the end of this section, the reconstruction algorithm of Part Two in SMTSVQ for the unshadowed blocks is drawn in the following.

Algorithm 2: [The reconstruction algorithm of Part Two in SMTSVQ for the unshadowed blocks]

```
Step 1:  Let the unshadowed block be x, and the number of the bits transmitted be s.
Step 2:  /* Locate the precise node in advance for the codebook searching; here we let root node be in level 0 */
         For each level L, where $L \le s$, repeat the following.
            If the value of the L-th bit of the received index of x equals 0, then locates it at the left child, else locates it at the right child.
Step 3:  /* Search the node of the codebook tree using the side match method */
         For each level L, where $s \le L \le n$, repeat the following.
            If the side match distortions between x and the left child are larger than those between x and the right child, then locate it at the left child, else locate it at the right child.
Step 4:  Use the codeword of the current leaf node to predict x.
```

4. Experimental Results

Our experiments are done on a SUN SPARC workstation. Each of the images we use contains 512×512 pixels, and each pixel has 256 gray levels. In addition, these images are divided into 4×4-pixel blocks in VQ. Thus each block of VQ is a 16-dimensional vector. In this paper, in order to compare the performance of PIT methods, we need to consider four factors. They are the transmission bit rate of each phase, the accumulated bit rate of phases, the quality of the reconstructed image, and the reconstruction time. We employ the bit per pixel (BPP) to estimate the transmission bit rate. It is defined as

$$BPP = B/P, \qquad (1)$$

where P is the total number of pixels in an image and B is the total number of transmitted bits for this image. As for the image quality, the peak signal-to-noise ratio (PSNR) is employed. It is defined as

$$\mathrm{PSNR} = 10\times\log_{10}\frac{255^2}{\mathrm{MSE}}\ \mathrm{dB}, \qquad (2)$$

where MSE is the mean-square error. For an m×m image, its MSE is defined as

$$\mathrm{MSE} = \left(\frac{1}{m}\right)^2\times\sum_{i=1}^{m}\sum_{j=1}^{m}(\alpha_{[i,j]}-\beta_{[i,j]})^2. \quad (3)$$

Here $\alpha_{[i,j]}$ and $\beta_{[i,j]}$ denote the original and decoded gray levels of the pixel [i,j] in the image, respectively. A larger PSNR value means that the encoded image preserves the original image quality better.

In Table 1, we show the transmission bit rates R(p)s, the accumulated bit rates T(p)s, and the image quality PSNR of SPITM, FRM, TSVQ, and SMTSVQ in our experiments. Here p denotes the phase number. The experimental image is "*Lena*". Consider the first phase of the image transmission, SPITM and FRM need 0.056 BPP each, and the PSNRs of their reconstructed images are 18.942 dB and 20.938 dB, respectively. TSVQ needs 0.0625 BPP in the first phase, and the PSNR of its reconstructed image is 18.526 dB. As for SMTSVQ, it needs 0.0625 BPP, and the PSNR of its reconstructed image is 22.25 dB after the first phase. In Table 1, SMTSVQ gets the highest image quality after the first phase. The difference of PSNR between FRM and SMTSVQ is 1.3 dB. Moreover, the difference of PSNR between SMTSVQ and TSVQ is up to 3.7dB. To compare these methods in detail, we draw a chart of the image quality of SPITM, FRM, TSVQ and SMTSVQ v.s. the accumulated bit rates in Figure 6. We find that, in the figure, the image quality of SMTSVQ is better than that of the other methods in each transmission phase. The reason why SMTSVQ has the better performance is that it takes full advantage of the spatial contiguity and interpixel correlation of image data. SMTSVQ increases the decoding quality of the first half of image blocks and uses the side match technique to recover the other half of blocks. Since images have the feature of spatial contiguity, the recovery quality is not bad.

The reconstruction time is also an important factor in PIT. If the reconstruction time is too long, then the user may as well transmit the whole image directly from the sender to the receiver. In the following, we show the reconstruction time of SPITM, FRM, TSVQ and SMTSVQ in Table 2. We see that, in the table, SPITM and SMTSVQ need more reconstruction time than the other two methods. This is because both of them employ the side match methods to rebuild the unreceived blocks. However, the reconstruction time of SMTSVQ is much less than that of SPITM and is very close to that of FRM. The reason is that SMTSVQ also adopts the TSVQ structure to accelerate the searching of the codebook. SMTSVQ is therefore efficient and effective for progressive transmission. This is because the side match method improves the image quality of TSVQ while the TSVQ structure speeds up the codebook searching of the side match.

In order to show the visual quality of SPITM, FRM, TSVQ and SMTSVQ, we print their progressive images from phase one to phase four in Figure 7. In Figure 7, we see that the visual quality of SMTSVQ is also the best among the images in each phase.

5. Conclusions

In this paper, a new PIT method is proposed. It is called the side-match reconstruction method using TSVQ (SMTSVQ). Our method builds a full-search tree as the TSVQ method does. However, in each phase, only half of the blocks are transmitted interleavedly, and, moreover, each block sends two bits. As for the unreceived blocks, the side match reconstruction technique is employed to rebuild them. Using the side match technique, we can take full advantage of the spatial contiguity and interpixel correlation of image data. Moreover, the TSVQ structure can accelerate the codebook search process of the side match technique. Since the TSVQ structure and the side match method bring out the best in each other, SMTSVQ has good performance for progressive transmission.

According to our experimental results, the image quality of our method is better than that of SPITM, FRM and TSVQ in each transmission phase. At first phases, the difference of PSNR between our method and TSVQ is even up to 3.7dB. Besides, the reconstruction time of our method is much less than that of SPITM and is very close to that of FRM and TSVQ. Therefore, our method is efficient and is more effective than the other methods proposed previously.

References

[1] A. Buzo, A. H. Gray, R. M. Gray, and J. D. Markel, "Speech Coding Based upon Vector Quantization," *IEEE Trans. Acous., Speech, Signal Processing*, Vol. ASSP-28, Oct. 1980, pp. 562-574.

[2] C. C. Chang, J. C. Jau, and T. S. Chen, "A Fast Reconstruction Method for Transmitting Images Progressively," *IEEE Trans. on Consumer Electronics*, Vol. 44, No. 4, Nov. 1998, pp. 1225-1233.

[3] G. Qiu, "A Progressively Predictive Image Pyramid for Efficient Lossless Coding," *IEEE Trans. on Image Processing*, Vol. 8, No. 1, Jan. 1999, pp. 109-115.

[4] J. H. Jiang, C. C. Chang, and T. S. Chen, "Selective Progressive Image Transmission Using Diagonal Sampling Technique," *Proceedings of International Symposium on Digital Media Information Base*, Nara, Japan, 1997 pp. 56-97.

[5] K. H. Tzou, "Progressive Image Transmission: A Review and Comparison of Technologies," *Optical Engineering*, Vol.26, No. 7,1987, pp. 581-589.

[6] R. F. Chang and W. T. Chen, "Side-Match Vector Quantization for Reconstruction of Lost Blocks," *Journal of Visual Comm. and Image Representation*, Vol. 4, No. 2, Jun. 1993, pp. 171-177.

[7] R. Y. Wang, E. A. Riskin and R. Ladner, "Codebook Organization to Enhance Maximum a Posteriori Detection of Progressive Transmission of Vector Quantized Image Over Noisy Channels," *IEEE Trans. on Image Processing*, Vol. 5, No. 1, Jan. 1996, pp. 37-48.

[8] Y. Linde, A. Buzo, and R. M. Gray, "An Algorithm for Vector Quantizer Design," *IEEE Trans. on Comm.*, COM-28, Jan. 1980, pp. 84-95.

Table 1. The bit rates and the PSNRs of SPITM, FRM, TSVQ, and SMTSVQ

Phase	SPITM			FRM			TSVQ			SMTSVQ		
	R(p)	T(p)	PSNR	R(p)	T(p)	PSNR	R(p)	T(p)	PSNR	R(p)	T(p)	PSNR
1	0.056	0.056	18.942	0.056	0.056	20.938	0.0625	0.0625	18.526	0.0625	0.0625	22.250
2	0.056	0.111	21.923	0.056	0.111	22.805	0.0625	0.125	22.974	0.0625	0.125	25.582
3	0.056	0.167	23.981	0.056	0.167	24.598	0.0625	0.1875	25.828	0.0625	0.1875	26.867
4	0.056	0.222	24.895	0.056	0.222	25.487	0.0625	0.25	27.002	0.0625	0.25	28.262
5	0.056	0.278	25.911	0.056	0.278	26.697	0.0625	0.3125	27.756	0.0625	0.3125	28.840
6	0.056	0.333	27.118	0.056	0.333	27.661	0.0625	0.375	28.793	0.0625	0.375	29.700
7	0.056	0.389	28.045	0.056	0.389	28.587	0.0625	0.4375	29.800	0.0625	0.4375	30.147
8	0.056	0.444	29.719	0.056	0.444	29.598	0.0625	0.5	30.747	0.0625	0.5	30.747
9	0.056	0.5	30.914	0.056	0.5	30.914						

Table 2. The execution time of SPITM, FRM, TSVQ, and SMTSVQ (in seconds)

	1	2	3	4	5	6	7	8	9
SPITM	10.051	10.109	9.832	9.066	7.684	6.207	4.781	2.859	0.330
FRM	0.391	0.391	0.391	0.438	0.332	0.391	0.332	0.277	0.391
TSVQ	0.172	0.223	0.168	0.172	0.160	0.172	0.211	0.223	
SMTSVQ	0.430	0.551	0.711	0.934	0.719	0.609	0.441	0.219	

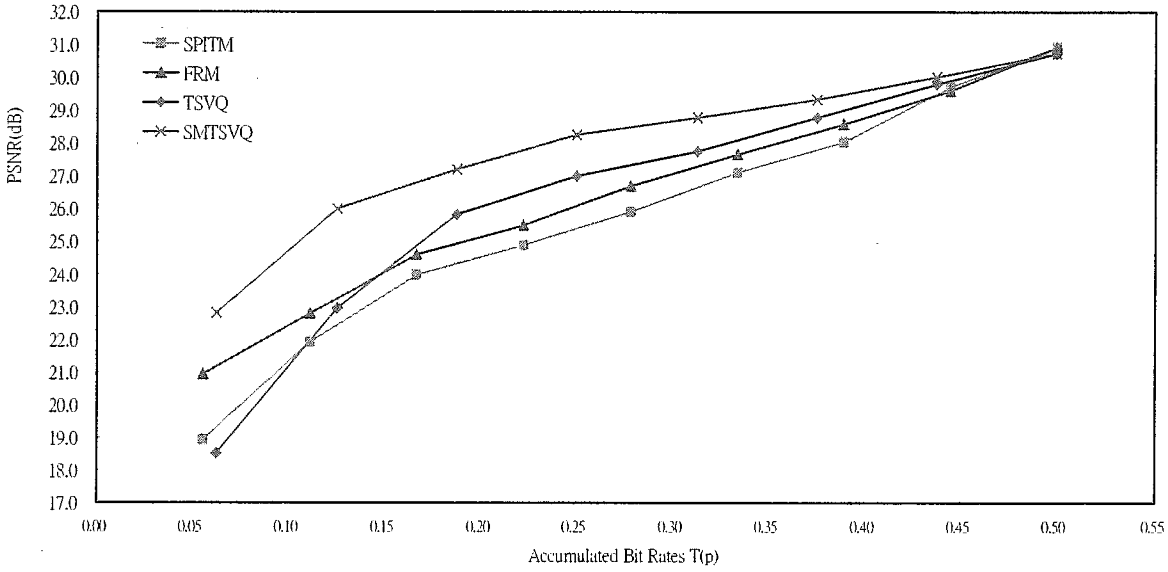
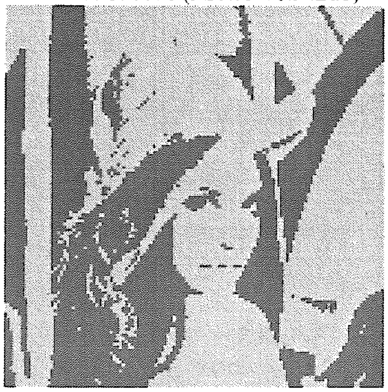


Figure 6: The image quality of SPITM, FRM, TSVQ and SMTSVQ v.s. the accumulated bit rates

(a1) Phase 1 of SPITM (PSNR=18.942db)

(b1) Phase 1 of FRM (PSNR=20.938db)



(c1) Phase 1 of TSVQ (PSNR=18.526db)

(d1) Phase 1 of SMTSVQ (PSNR=22.250db)


(a2) Phase 2 of SPITM (PSNR=21.923db)


(b2) Phase 2 of FRM (PSNR=22.805db)


(c2) Phase 2 of TSVQ (PSNR=22.974db)

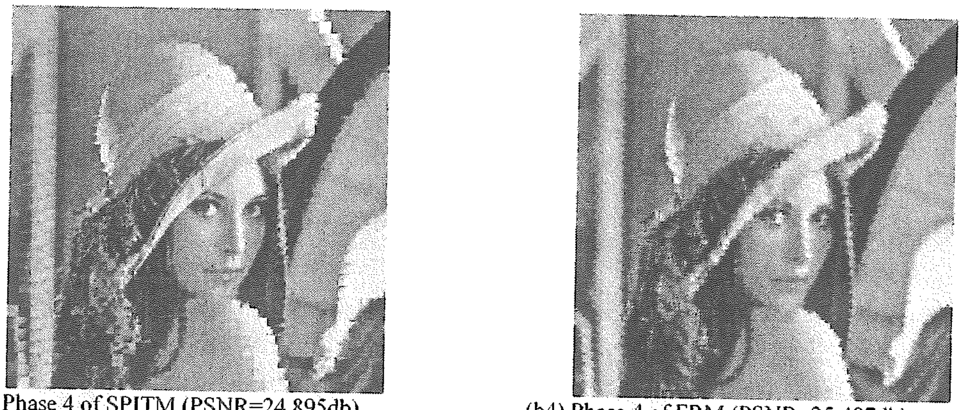

(d2) Phase 2 of SMTSVQ (PSNR=25.582db)


(a3) Phase 3 of SPITM (PSNR=23.981db)


(b3) Phase 3 of FRM (PSNR=24.598db)

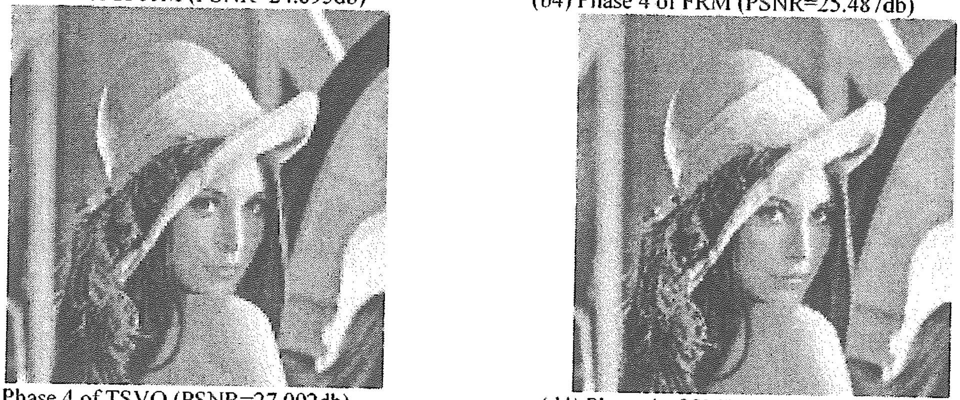

(c3)Phase 3 of TSVQ (PSNR=25.858db)


(d3) Phase 3 of SMTSVQ (PSNR=26.867db)



(a4) Phase 4 of SPITM (PSNR=24.895db)

(b4) Phase 4 of FRM (PSNR=25.487db)



(c4) Phase 4 of TSVQ (PSNR=27.002db)

(d4) Phase 4 of SMTSVQ (PSNR=28.262db)

Figure 7: The results of Phases One to Four of SPITM, FRM, TSVQ and SMTSVQ