IMAGE VECTOR QUANTIZATION INDICES RECOVERY USING LAGRANGE INTERPOLATION

Yung-Gi Wu and Chia-Hao Wu
Leader University
Institute of Applied Information
Department of Computer Science and Information Engineering
Tainan, Taiwan
wyg@mail.leader.edu.tw, heeroart@yahoo.com.tw

ABSTRACT

Vector quantization (VQ) is an efficient coding algorithm due to its fast decoding efficiency. Indices of VQ will be lost during the transmission because of the signal interference. In this paper, we propose an efficient estimation method by using the Lagrange interpolation formula to recover the lost indices in image vector quantization codec. If the image or video has the limitation of the period of validity, re-transmitting the data wastes of time. Therefore, using the received correct data to estimate and recover the lost data is efficient in time-constraint situation such as network conference. For nature image or video, the pixels with its neighbors are correlative. Since the VQ partitions the image into sub-blocks and quantize them to form the indices to transmit, the correlation between adjacent indices is very strong. There are two important parts of the proposed method. One is preprocessing process and the other is the estimation process. In preprocessing, we modify the order of code-vectors in the VQ codebook to increases the correlation between neighboring vectors. On the second part, the recovery process on the decoder, using the Lagrange interpolation formula to constitute a polynomial to describe the tendency of VQ indices, and use the polynomial to estimate the lost VQ indices. The simulation results demonstrate that our method can efficient estimate the lost indices in acceptable visual quality.

KEYWORDS
Vector Quantization, Lagrange interpolation, index recovery

1. INTRODUCTION

In 1980, Gersho and Gray developed the vector quantization and many other researches have been working on the research topic of VQ [1] [2] [3] [4] [5] [6] [7]. Many other coding schemes have been devised like as discrete cosine transform, block truncation coding, wavelets coding, etc. VQ is still one of the most successful signal processing techniques. VQ is powerful because it is quick and simple on decoding so that many applications that desire fast decoding select this technique to compress data before transmission. Although using this technique decays the image fidelity, in many time-constraint situations, it is acceptable. Under the limit of time emergency during the transmission, the system must ensure visual acceptable on decoder in short time. Hence, saving the re-transmit time when some data lost during transmission is the major topic in this paper.

We introduce the image VQ briefly. During encoding, it searches a best match code-vector in the codebook to replace input vector and transmit the code-vector index to the channel. On the decoder, using these received indices to get code-vectors from the codebook and paste them to reconstruct the decoded image. VQ can be viewed as a form of pattern recognition where an input pattern is ''approximated'' by one of a predetermined set of standard patterns by matching it with one of a stored set of vector indices in the codebook.

During the Internet transmission, random noises may cause the indices to lose. In order to save time, we propose a method to estimate these lost data and to recover them rather than transmit the whole data again. But, the data lost rate must limited by the network rules, large data lost may cause the receiver determine the network disconnection. The main ideal of our proposed method is decreasing the network traffic capacity while maintaining the quality as well as possible. Therefore the data lost rate causes network disconnection does not include in our result. In general, recovery will not be considered when the data is lost seriously. In which, system will re-transmit the data. The proposed method is efficient with respect to time constraint situation and bandwidth usage.

2. THE PROPOSED SCHEME

In this paper, we show a method using Lagrange interpolation formula [8] [9] [10] to estimate and recover the lost data for VQ. There are two important parts of our proposed method. One of them is preprocessing process, which sorts codebook to improve the correlation between neighboring vectors. Notice that, we just sort the codebook once before any other process so that the pre-processing process is off-line. Another part of our method is the recovery process, which uses Lagrange interpolation formula to estimate the lost indices.

We sort codebook to increase the relationship between near vectors of codebook. Figure 1 shows the steps of the preprocessing process. At first, we calculate the difference D_i to classify the codewords in the codebook as follow:

$$D_i = \max\{V_i\} - \min\{V_i\} \tag{1}$$

$$\begin{cases} Smooth & , D_i \le \varepsilon \\ Edge & , otherwise \end{cases}$$

Where V represents the set of code-vectors in the codebook and V_i denotes the i-th codevector. The equation (1) means to select the maximum and minimum gray in each codeword and calculate the difference of them. We classify code-vectors into two classes which are smooth and edge, according to the difference values D_i and the given threshold value ε. Then, we use the mean value m_i of each vector to sort the two classes of code-vectors.

Finally, the sorted vectors in the codebook would be stored and used in all of the encoding.

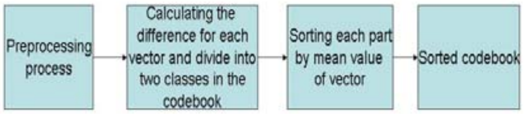

Figure 1 Preprocessing process diagram

When some VQ indices are lost during the transmission, the proposed method uses the correct indices to estimate and recover the lost ones. In this paper, we use Lagrange interpolation to describe the tendency of VQ indices. The Lagrange interpolating polynomial [10] can be defined as *P(x)*, which is given by

$$P(x) = \sum_{i=1}^{n} P_i(x) \tag{2}$$

where

$$P_i(x) = y_i \prod_{k=1, k \neq i}^{n} \frac{x - x_k}{x_i - x_k}. \tag{3}$$

Written explicitly,

$$P(x) = \frac{(x-x_2)(x-x_3)\ldots(x-x_n)}{(x_1-x_2)(x_1-x_3)\ldots(x_1-x_n)} y_1 + \frac{(x-x_1)(x-x_3)\ldots(x-x_n)}{(x_2-x_1)(x_2-x_3)\ldots(x_2-x_n)} y_2 + \ldots + \frac{(x-x_1)(x-x_2)\ldots(x-x_{n-1})}{(x_n-x_1)(x_n-x_2)\ldots(x_n-x_{n-1})} y_n. \tag{4}$$

The Lagrange interpolating polynomial can be used to estimate the lost index based on the correlation of neighboring data. Here, we give those indices a set of corresponding coordinates as the Table 1 lists.

Table 1 Polynomial corresponding coordinates for each index.

y_i	L_1	L_2	M	R_2	R_1
x_i	-2	-1	0	1	2

In order to get acceptable quality of reconstructed image, we use the two-way polynomial to estimate the lost index. Assume that the lost index M has a coordinate [m, n], and L_1, L_2, R_2, R_1 are the correct indices. If L_1 or L_2 was lost, getting their left one index. If R_1 or R_2 was lost, getting their right one index. First, we constitute a polynomial using correct four indices as follows:

$$P(x) = \frac{(x+1)(x-1)(x+2)}{-12} L_1 + \frac{(x+2)(x-1)(x-2)}{6} L_2 + \frac{(x+2)(x+1)(x-2)}{-6} R_2 + \frac{(x+2)(x+1)(x-1)}{12} R_1 \tag{5}$$

Then we use two-way polynomial to estimate the lost index M, getting different four coefficients for each way. Table 2 shows the coordinates of coefficient in each way.

Table 2 Coordinates of coefficient

y_i	L_1	L_2	M	R_2	R_1
horizontal	[m-1, n-1]	[m, n-1]	[m, n]	[m, n+1]	[m+1, n+1]
vertical	[m-1, n+1]	[m, n+1]	[m, n]	[m, n-1]	[m+1, n-1]

By using the equation (5) and Table 2, we can get two values of $P_h(x)$ and $P_v(x)$. Because the both two ways are adopted to estimate the lost index, the final estimated value is determined by (7).

$$P_h(x) = \frac{(x+1)(x-1)(x+2)}{-12}[m-1, n-1] + \frac{(x+2)(x-1)(x-2)}{6}[m, n-1] + \frac{(x+2)(x+1)(x-2)}{-6}[m, n+1] + \frac{(x+2)(x+1)(x-1)}{12}[m+1, n+1] \tag{6a}$$

$$P_v(x) = \frac{(x+1)(x-1)(x+2)}{-12}[m-1, n+1] + \frac{(x+2)(x-1)(x-2)}{6}[m, n+1] + \frac{(x+2)(x+1)(x-2)}{-6}[m, n-1] + \frac{(x+2)(x+1)(x-1)}{12}[m+1, n-1] \tag{6b}$$

$$M = \frac{P_h(x) + P_v(x)}{2} \tag{7}$$

During all of above steps, the lost index will be estimated and recovered. Figure 2 shows the steps of recovery process and the Figure 3 shows the whole system configuration.

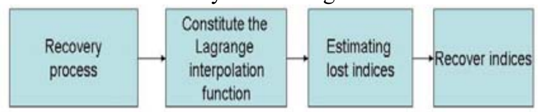

Figure 2 Recovery process diagram.

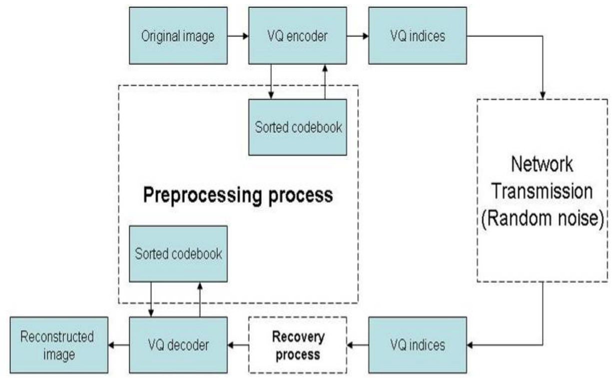

Figure 3 System diagram.

3. SIMULATION RESULTS

We select the gray level images Lenna, Tiffany, and Boat, whose size is 512×512 to be the test images. The codebook size is 256 and set the threshold ε = 128 to classify the codebook as described in the previous section. The quality measure for reconstructed image is the peak signal-noise rate (PSNR):

$$PSNR = 10 \times \log_{10}\left(\frac{255^2}{MSE}\right) \ (dB). \tag{8}$$

Where the MSE:

$$MSE = \left(\frac{1}{M \times N}\right) \sum_{i=1}^{M} \sum_{j=1}^{N} (y_{ij} - \hat{y}_{ij})^2 \tag{9}$$

$M \times N$ is the image size and y_{ij} and $\hat{y}_{ij}$ denote the pixel value at the location (i, j) of original and reconstructed images, respectively.

In order to show the performance, we compare to other methods. Table 3 lists the simulation results at different lost-rate and the PSNR values of reconstructed image using different methods for image recovery. In this table, each reconstructed image at difference lost-rate has two PSNR values (compared with original image and compared with VQ reconstructed image at no data lost). Figure 4 illustrates simulation results of Table 3. Random recovery is to recovery those indices by random padding. Notice that, we all know low-pass filter be used to eliminate the random noise. Therefore, we also use it for test. By the observation of Table 3 and Figure 4, we can see that the proposed method in image recovery is efficiently.

The quality of using low-pass filter, one-way, and two-way Lagrange interpolations in low data lost rates are not so obvious, but in high data lost rates or lost is in the edge blocks, the two-way Lagrange interpolation is more efficient.

Table 3 The quality of reconstructed image (Lenna) expressed in dB with different methods and different lost -rates.

VQ indices lost-rate (%)	0	0.1	0.5	5	10
Non recovery	30.154	29	26.31	18.438	15.802
Random recovery	30.154	29.557	28.342	21.705	19.042
Low-pass filter	30.154	30.097	29.822	27.331	25.150
One-way Lagrange	30.154	30.105	29.828	26.921	23.701
Two-way Lagrange	30.154	30.134	30.0	28.418	27.215

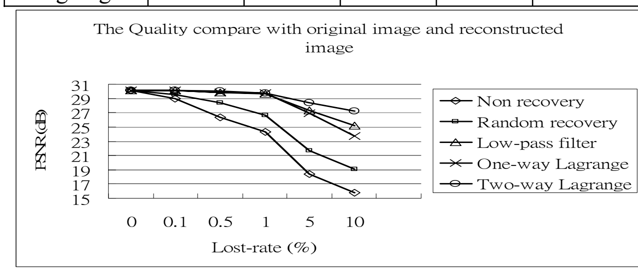

Figure 4 The quality of reconstructed image (Lenna) with different methods and different lost-rates.

Figure 5, Figure 9, and Figure 12 show the original test images (Lenna, Boat, and Tiffany). Figure 6 shows the Lenna VQ reconstructed image without any index lost. Figure 7(a), Figure 8(a), Figure 10(a), Figure 11(a), Figure 13(a) and Figure 14(a) show the test images at difference data lost (1% and 5%). Figure 7(b), Figure 8(b), Figure 10(b), Figure 11(b), Figure 13(b), and Figure 14(b) show the test images with different data lost rates using random selecting indices recovery to reconstruct images. Figure 7(c), Figure 8(c), Figure 10(c), Figure 11(c), Figure 13(c), and Figure 14(c) show the test images at different data lost rates using low-pass filter to estimate lost indices and recover them. Figure 7(d), Figure 8(d), Figure 10(d), Figure 11(d), Figure 13(d), and Figure 14(d) show the test images with different data lost rate and use two-way Lagrange interpolation to estimate lost indices and recover them. In addition, notice that the time to recover one image is about 0.31 milliseconds. The table 4 shows the time of recover one hundred images or retransmission one hundred VQ encoded images.

Our results show that our proposed method in high data lost-rate is powerful than other methods. In other method, the high data lost rate causes the quality of reconstructed image can not accepted. And the estimating time of the proposed method is quiet fast. This result shows our method could be used in time requirement application.

Table 4 Compare the waste time of recovery images and retransmission images

One hundred 512×512 gray level images	
Using two-way Lagrange to recovery	Retransmission (T1 network : 1.544Mbps)
31 milliseconds	8290seconds
Bit rate = 0.5	Bit rate = 0.5

Figure 5 Original Lenna image.

Figure 6 The VQ reconstructed image (PSNR = 30.15dB)

Figure 7(a) Lost-rate 1%.

Figure 7(b) Random indices recovery at lost-rate 1%. PSNR = 26.622dB

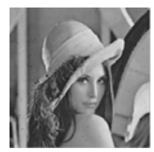

Figure 7(c) Low-pass filter recovery at lost-rate 1%. PSNR =29.607dB.

Figure 7(d) Two-way Lagrange recovery at lost-rate 1%. PSNR=29.735dB.

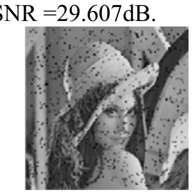

Figure 8(a) Lost-rate 5%.

Figure 8(b) Random indices recovery at lost-rate 5%. PSNR = 21.705dB

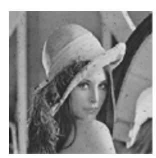

Figure 8(c) Low-pass filter recovery at lost-rate 5%. PSNR =27.331dB.

Figure 8(d) Two-way Lagrange recovery at lost-rate5%. PSNR=28.418dB.

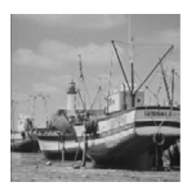

Figure 9 Original Boat image.

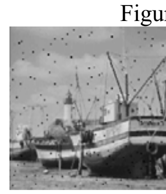

Figure 10(a) Lost-rate 1%.

Figure 10(b) Random indices recovery at lost-rate 1%.

PSNR = 25.180dB

Figure 10(c) Low-pass filter recovery at lost-rate 1%. PSNR =27.761dB.

Figure 10(d) Two-way Lagrange recovery at lost-rate1%. PSNR=27.934dB.

Figure 11(a) Lost-rate 5%.

Figure 11(b) Random indices recovery at lost-rate 5%. PSNR = 20.753dB

Figure 11(c) Low-pass filter recovery at lost-rate 5%. PSNR =26.092dB.

Figure 11(d) Two-way Lagrange recovery at lost-rate 5%. PSNR=27.101dB.

Figure 12 Original Tiffany image.

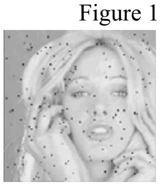

Figure 13(a) Lost-rate 1%.

Figure 13(b) Random indices recovery at lost-rate 1%. PSNR = 25.305dB

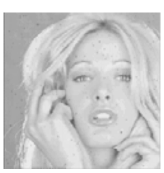

Figure 13(c) Low-pass filter recovery at lost-rate 1%. PSNR =28.801dB.

Figure 13(d) Two-way Lagrange recovery at lost-rate 1%. PSNR=29.436dB.

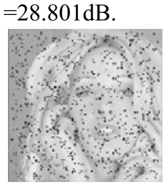

Figure 14(a) Lost-rate 5%.

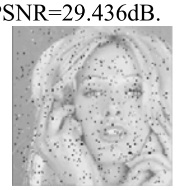

Figure 14(b) Random indices recovery at lost-rate 5%. PSNR = 19.485dB

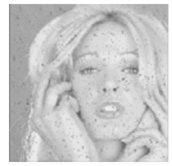

Figure 14(c) Low-pass filter recovery at lost-rate 5%. PSNR =26.215dB.

Figure 14(d) Two-way Lagrange recovery at lost-rate 5%. PSNR=28.934dB.

4. CONCLUSION

We present an efficient data recovery method for VQ encoded image transmission in this paper. During data transmission, if the data lost happens, we usually request the sender to transmit these data again. If the image or video has the limitation of the period of validity, re-transmitting the data wastes of time. Hence, using the received correct data to estimate and recover the lost data is efficient in time-constraint situation. In our simulation result, using the correct indices to constitute a Lagrange interpolating polynomial and estimate the lost indices to reconstruct image. The proposed method has the property of fast processing and the reconstructed images have visual acceptable quality. Therefore, these results show our method can efficiently improve the transmission quality when the time-constraint requirement is needed.

REFERENCES

[1] K. R. Rao and J. J. Hwang, Techniques and Standards for Image Video, and Audio Coding, Prentice Hall, Englewood Cliffs, NJ, 1996.

[2] K. Sayood, Introduction to Data Compression, Morgan Kaufmann, 2000.

[3] A. Gersho and R. M. Gray, “Vector Quantization and Signal Compression,” Kluwer, Norwell, MA, 1992.

[4] W. B. Kleijn and K. K. Paliwal, “Speech Coding and Synthesis. Amsterdam,” Elsevier, The Netherlands, 1995.

[5] N. M. Nasabadi and R. A. King, “Image coding using vector quantization: A review,” IEEE Trans. Commun. 36(8), pp.957-971, 1988.

[6] Y. Linde, A. Buzo, and R. M. Gray, 1980, “An algorithm for vector quantization design,” IEEE Trans. Commun. COM-28(1), pp. 84-95.

[7] Yung-Gi Wu, Kuo-Lun Fan, “Fast vector quantization image coding by mean value predictive algorithm,” Journal of Electronic Imaging, 13(2), pp. 324-333, 2004.

[8] M. Abramowitz and I. A. Stegun, “Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables 9th,” Dover, New York, pp. 878-879 and 883, 1972.

[9] H. Jeffreys and Jeffreys, “Methods of Mathematical Physics 3rd,” Cambridge University Press, England, pp. 260-261, 1988.

[10] http://mathworld.wolfram.com/LagrangeInterpolatingPolynomial.html