

Hiding Information in VQ Index Tables with Reversibility

ZhiHui Wang¹, KuoNan Chen², ChinChen Chang³, and MingChu Li⁴

¹Department of Software, Dalian University of Technology, DaLian, China
wangzhihui1017@yahoo.cn

²Department of Computer Science and Information Engineering, National Chung Cheng University, Chiayi, Taiwan
ckn95p@gmail.com

³Department of Information Engineering and Computer Science, Feng Chia University, Taichung, Taiwan
alan3c@gmail.com

⁴Department of Software, Dalian University of Technology, DaLian, China
li_mingchu@yahoo.com

Abstract—In this paper, we propose a simple and efficient reversible information-hiding scheme for VQ index tables. One secret bit can be embedded into one index pair if this pair passes the predefined restriction. Even if the index pair does not pass the test, it can still carry one secret bit if an additional indicator (a preserved index value) is attached. The experimental results show that our proposed schemes can accomplish lossless recovery of VQ index tables, and the hiding capacity is satisfactory.

Index Term—information hiding; lossless recovery; VQ index

I. INTRODUCTION

Data transmission via the Internet has become highly convenient because of the great improvement in network technologies. Although we can share files and communicate with each other quickly and easily via the Internet, Internet use for communication is accompanied by two problems: security and bandwidth availability. Since the Internet is a public environment, any malicious attacker can intercept and get information he or she is not intended to have, a serious problem, especially when the information concerns national security. Still, the great advantages of transmitting data via the Internet have led to its widespread use. With the huge number of Internet users, bandwidth availability is another issue that merits further study.

Several effective encryption systems, such as DES [1], RSA [2], and AES [3], do a good job of protecting secret information. However, these systems' ciphertext is singular and can attract attention from malicious attackers. In the image processing field, the most commonly used technology for improving the security of data is to embed secret information in the cover image [4-6]. These image data-hiding schemes aim to hide as much secret information as well as possible and to minimize the quality degradation of the embedded image. Since the cover image and the embedded image in these systems are indistinguishable, the embedded image does not draw the attention of eavesdroppers, making it more likely that the secret information can be transmitted safely.

The other notable problem is the availability of Internet bandwidth. One of the solutions is compressing data to reduce the size of a transmission so more files can be transmitted during a fixed time. Vector quantization

(VQ) [7], one of the compression schemes for images, is satisfactory in terms of compression ratio and is effective in retaining good visual quality of the embedded images. Many studies have focused on embedding secret information into the VQ indices [8-10]. For example, in 1999, Lin and Wang proposed a watermarking scheme for embedding secret information into VQ indices [9] by clustering codewords in a codebook into pairs based on their similarity. When a block in the cover image is being compressed as a codeword, instead of selecting only one similar codeword, a pair of codewords is chosen as candidates, with one of the two codewords used to carry bit 1, and the other one to embed bit 0. The hiding capacity of this scheme is one bit per block. In 2008, Chiang and Tsai proposed another embedding scheme for VQ indices based on similar concepts of codeword clustering [8]. In their scheme, codewords are clustered in a codebook into 4-, 3-, 2-, and 1-member clusters, based on a predefined similarity threshold. Each 4-member cluster can carry 2 secret bits, a pair of 3-member clusters can carry 3 secret bits, 1 secret bit can be embedded into each 2-member cluster, and all 1-member clusters are unembeddable. Compared to the scheme in [9], the hiding capacity of this system is much improved. However, although these schemes are simple and efficient, they cannot restore the original VQ indices after the secret bits are extracted.

In this paper, we propose a reversible information-hiding scheme based on VQ indices. The secret bits are embedded into VQ index tables using indicators if necessary. The experimental results show that the proposed scheme can reverse the VQ indices correctly with satisfactory hiding capacity.

II. RELATED WORKS

In an unpublished paper [11], Chang, Chen, and Wang proposed a reversible information-hiding scheme for VQ index tables. In their scheme, they preserved the first and last codewords in a codebook to act as indicators for embedding secret information. If the secret bit is equal to 0, the first index will be attached to the front of the original index and, if the last index is found, secret bit 1 is embedded in the compression code. The procedure is illustrated as follows, where s is secret bits pattern, and F and L are the first and the last indices in a codebook, respectively.

Procedure Indicator ()

Input: a pair of VQ indices (i_1, i_2)

Output: a pair of watermarked VQ indices (j_1, j_2)

- 1) If $s = 00$, then let $j_1 = F \parallel i_1$ and $j_2 = F \parallel i_2$.
- 2) If $s = 01$, then let $j_1 = F \parallel i_1$ and $j_2 = L \parallel i_2$.
- 3) If $s = 10$, then let $j_1 = L \parallel i_1$ and $j_2 = F \parallel i_2$.
- 4) If $s = 11$, then let $j_1 = L \parallel i_1$ and $j_2 = L \parallel i_2$.

In this way, two secret bits can be embedded into a pair of VQ indices with two additional VQ indices added into the compression result.

For example, if the original index values of an image are listed as 30, 50, 45, and 60. By composing two indices as a pair, two pairs are created: (30, 50), and (45, 60). Here, 0110 are four secret bits, $F = 1$, and $L = 256$. To embed these four secret bits in the index pairs, the original index pairs are expanded as (1, 30, 256, 50), and (256, 45, 0, 60). After recomposing the index results pair by pair, the final index pairs are (1, 30), (256, 50), (256, 45), and (0, 60). On the receiver end, if the index values 1 and 256 are found in the result, the secret bits 0 and 1 are extracted, respectively. Except 1 and 256, the other index values are used to recover the VQ-coded image.

III. THE PROPOSED SCHEME

In our proposed scheme, we compress the original image using a reduced codebook where three indices are preserved from use in the VQ compression processes. When the VQ index table is generated according to the reduced codebook, each two VQ indices are joined as a pair for further processing. If an index pair passes the predefined restriction, it can carry one secret bit without any side effect; otherwise, an indicator (one of the preserved indices) is attached in front of the index pair to make this pair embeddable for carrying 1 secret bit. Thus, the tradeoff is that one VQ index is added into the compression results.

A. The Embedding Processes

To choose the three preserved indices in the codebook, we compress the original image by using traditional VQ compression. Through analysis with the VQ index table (the compression code), we find the two indices that are the least frequently used. We rearrange all indices in the codebook and set these two indices as 0 and 255. The indices 0 and 255 are named F and L , respectively. The third preserved index is selected from index 1 to 254 by using a pseudo-random-number generator with a private key K , and the selected result is named I . The secret information is transformed into its binary representation s in advance, and we compress the original image again by using the modified codebook from which the three indices (F , L , and I) are preserved from use. After the new VQ index table (the compression code) is generated, each two indices are composed as a pair (i_1, i_2). All index pairs are analyzed to determine the number of pairs n_1 in which $i_1 > i_2$, and the number of pairs n_2 in which $i_1 < i_2$. By comparing n_1 and n_2 , we decide whether the embedding process is Case A or B: If $n_1 \geq n_2$, then the embedding

process falls into Case A; otherwise, the embedding process falls into Case B. Finally, the watermarked VQ index table is generated after all secret bits are embedded into it. For each index pair, the embedding procedure is illustrated as follows.

Input: L, I, n_1, n_2 , a pair of indices (i_1, i_2), and a secret bit

s

Output: a watermarked index pair (j_1, j_2)

```

If ( $n_1 \geq n_2$ ) // Case A
  {If ( $i_1 > i_2$ )
    {If ( $s = 1$ )
      {( $j_1, j_2$ ) = ( $i_1, i_1 - i_2$ );}
    }
    Else if ( $s = 0$ )
      {( $j_1, j_2$ ) = ( $i_1 - i_2, i_1$ );}
    }
  }
  Else if ( $i_1 = i_2$ )
    {If ( $s = 1$ )
      {( $j_1, j_2$ ) = ( $i_1, I$ );}
    }
    Else if ( $s = 0$ )
      {( $j_1, j_2$ ) = ( $I, i_1$ );}
    }
  }
  Else if ( $i_1 < i_2$ )
    {If ( $s = 1$ )
      {( $j_1, j_2$ ) = ( $L, i_1, i_2$ );}
    }
    Else if ( $s = 0$ )
      {( $j_1, j_2$ ) = ( $L, i_2, i_1$ );}
    }
  }
Else if ( $n_1 < n_2$ ) // Case B
  {If ( $i_1 < i_2$ )
    {If ( $s = 1$ )
      {( $j_1, j_2$ ) = ( $i_1, i_2 - i_1$ );}
    }
    Else if ( $s = 0$ )
      {( $j_1, j_2$ ) = ( $i_2 - i_1, i_1$ );}
    }
  }
  Else if ( $i_1 = i_2$ )
    {If ( $s = 1$ )
      {( $j_1, j_2$ ) = ( $i_1, I$ );}
    }
    Else if ( $s = 0$ )
      {( $j_1, j_2$ ) = ( $I, i_1$ );}
    }
  }
  Else if ( $i_1 > i_2$ )
    {If ( $s = 1$ )
      {( $j_1, j_2$ ) = ( $L, i_1, i_2$ );}
    }
    Else if ( $s = 0$ )
      {( $j_1, j_2$ ) = ( $L, i_2, i_1$ );}
    }
  }

```

Assume the codebook size is 256, $I = 5$, $L = 255$, $s = 1101$ and that we know the embedding process falls into Case A by analyzing all indices. We use an index segment to illustrate the embedding processes (marked with thick border in Figure 1), in which the index pairs are (90, 30), (50, 60), (70, 70) and (254, 1). For the first index pair (90, 30) with its corresponding secret bit 1, the watermarked index pair (j_1, j_2) is equal to (90, 60) by applying the proper embedding rules: (j_1, j_2) = ($i_1, i_1 - i_2$). Secret bit 1 is embedded into the second index pair (50, 60) by attaching L in front of it. The watermarked result is (255, 50, 60). The third index pair is (70, 70), where the corresponding secret bit is 0. By applying (j_1, j_2) = (I, i_1), we get the watermarked result as (5, 70). For the last index pair (254, 1) with secret bit 1, the watermarked

253), which we determine by judging the unprocessed index 254. By the same process, we know the secret bit embedded here is 1 and the original index pair is (254, 1).

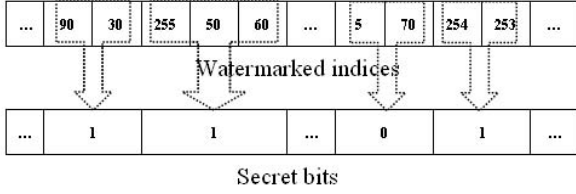


Figure 3. The processes of extracting secret bits

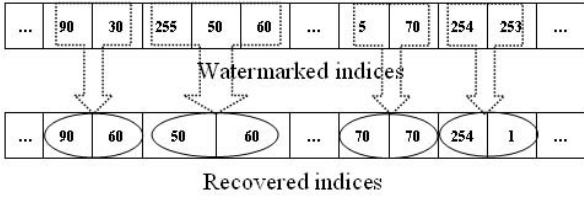


Figure 4. The indices-recovery processes

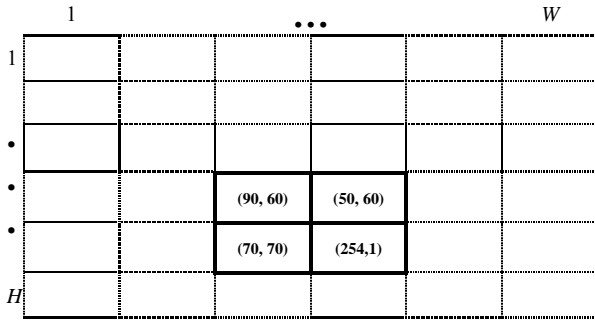


Figure 5. The recovered index table segment

Finally, the secret bits 1101 are extracted precisely, and the original index values are recovered correctly. The secret-bits extraction processes are shown in Figure 3, and the indices-recovery processes are shown in Figure 4. The recovered indices can be rearranged as shown in Figure 5. Since the VQ index table in Figure 5 is identical to that in Figure 1, it shows that our proposed scheme can reverse the VQ indices precisely.

C. Discussion

Assume the codebook size is N . In the proposed scheme, the two least frequently used indices, F and L , are set as the index 0 and index $N-1$, respectively, in the codebook. However, F is never used in both the embedding and the extracting processes; we just want to preserve index 0 from use in the VQ compression because we usually use the permutation of (x, d) or (d, x) to embed secret bit in the index pair, where d is the difference of the two indices in this pair. If an index pair $(x, 0)$ or $(0, x)$, where $x = 0$ to $N-2$, occurs in the VQ index table, we cannot embed any secret bits into it because $d = x$. In other words, we cannot find the second kind of permutation for (x, x) .

In our proposed scheme, L is used as an indicator that is attached in front of an index pair in the embedding process. On the receiver end, if L is discovered, we bind the following two indices as a pair for further processing. Therefore, L is an indicator for the receiver end. Imagine that, instead of setting L to index $N-1$, we set L to index $N-2$ in the embedding processes. Assume the embedding process falls into Case A, $s = 0$, and the index pair $(N-1, 1)$ occurs in the VQ index table. The watermarked index pair would be $(N-2, N-1)$ according to the embedding processes. However, when the receiver gets this watermarked index pair, the first index $N-2$ would be misjudged as an indicator. This is why we set L to index $N-1$.

IV. EXPERIMENTAL RESULTS

Four grey-level VQ-coded images—Lena, Jet, Pepper, and Toys, shown in Figures 6 (a) to (d), respectively—are used in our experiments. Their size is 512×512 , and they are created by using VQ compression with the modified codebook. To show the reversibility of our proposed scheme, the reversed VQ-coded images are shown in Figures 7 (a) to (d). Comparing Figures 6 and 7, it shows that our proposed scheme can reverse the VQ indices precisely.

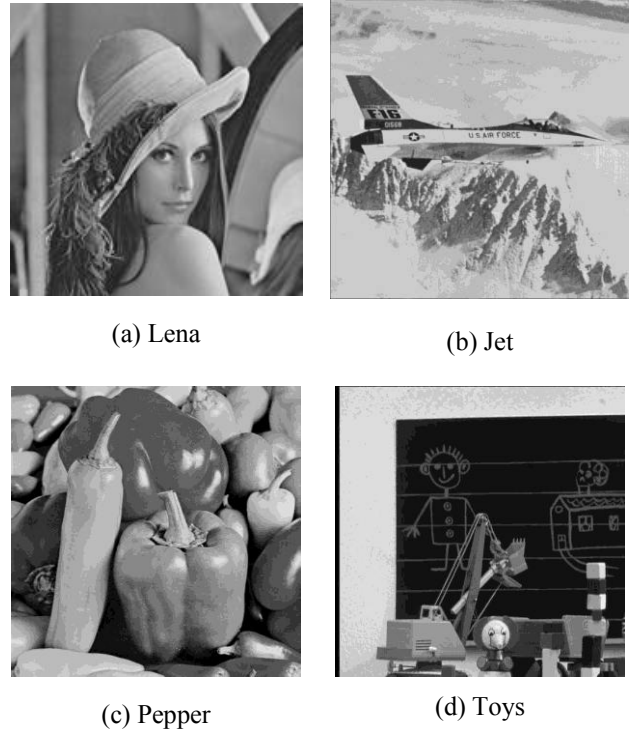


Figure 6. (a), (b), (c) and (d) are the original VQ-coded images

Table I lists the $PSNR$ values of the VQ-coded images and the reversed ones. Here, the peak signal-to-noise ratio ($PSNR$) is used to measure the image quality. The equation for $PSNR$ is:

$$PSNR = 10 \times \log_{10} \frac{255^2}{MSE}, \quad (1)$$



Figure 7. (a), (b), (c) and (d) are the reversed VQ-coded images

where MSE (mean square error) is defined as:

$$MSE = \left(\frac{1}{h \times w} \right) \sum_{i=1}^{h \times w} (X_i - X'_i)^2, \quad (2)$$

where h and w are the height and width of the image, respectively; and X_i and X'_i are the cover pixel value and stego pixel value, respectively.

It is obvious that the $PSNR$ values of reversed VQ-coded images are exactly the same as those of the original ones, further illustrating that our proposed scheme has complete reversibility.

TABLE I. THE COMPARISONS OF THE VQ-CODED IMAGES AND REVERSED VQ-CODED IMAGES

Images	Visual qualities (PSNR)	
	VQ-coded Image	Reversed VQ-coded Image
Lena	32.248 dB	32.248 dB
Jet	30.582 dB	30.582 dB
Pepper	31.407 dB	31.407 dB
Toys	31.156 dB	31.156 dB

TABLE II. THE NUMBER OF PAIRS THAT FALL INTO CASE B

Codebook size	Images	Number of index pairs embedded without indicators (percent)
512	Lena	4909 (60.0 %)
	Jet	5147 (62.8 %)
	Pepper	5319 (64.9 %)
	Toys	6268 (76.5 %)
256	Lena	5185 (63.3 %)
	Jet	5813 (71.0 %)
	Pepper	5604 (68.4 %)
	Toys	6506 (79.4 %)

In Table II, we show the percentages (the number of pairs embedded without indicators / the number of all pairs) of index pairs that need not any indicator for secret bits embedding in different codebook sizes. The higher percentages of this kind of index pairs, the less additional size would be added to the compression code. As we can observe from Table II, the percentages are all higher than 60%.

In Table III, we compare the hiding capacity (bits) and the bit rate between [11] and our proposed scheme. The hiding capacity used here means the number of secret bits that can be embedded into the original VQ index tables. And the bit rate is defined as follows:

$$\frac{\text{The Size of Compressed Results(bits)}}{\text{The Size of The Original Image (pixels)}} \quad (3)$$

The lower bit rate means that the smaller compressed result we have. In other words, the lower bit rate means that the compressed result has lower size expansion. By observing from Table III, it shows that our proposed scheme is superior to [11] in both hiding capacity and bit rate.

TABLE III. COMPARISONS BETWEEN SCHEME [11] AND OUR PROPOSED SCHEME

Images	Methods	Scheme [11]	The proposed scheme
Lena	Capacity	4524	8192
	Bit rate (bpp)	0.7239	0.5918
Jet	Capacity	3379	8192
	Bit rate (bpp)	0.7938	0.5742
Pepper	Capacity	4749	8192
	Bit rate (bpp)	0.7101	0.5790
Toys	Capacity	5102	8192
	Bit rate (bpp)	0.6886	0.5515

V. CONCLUSIONS

This paper proposes an efficient reversible data-hiding scheme for VQ index tables. The experimental results demonstrate that our proposed scheme can achieve the aim of reversibility of VQ indices. The hiding capacity is also satisfactory and superior to that in [11]. In the future, we will try to reduce the expansion size of the compression code while simultaneously improving the hiding capacity.

REFERENCES

- [1] National Institute of Standards & Technology, "Data encryption standard (DES)," Federal Information Processing Standards Publication, January 1977, vol. 46.
- [2] R. L. Rivest, A. Shamir and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," Communications of the ACM, vol. 21, no. 2, February 1978, pp. 120-126.
- [3] National Institute of Standards & Technology, "Announcing the advanced encryption standard (AES)," Federal Information Processing Standards Publication, vol. 197, no. 1, 2001.
- [4] W. L. Tai, C. M. Yeh, and C. C. Chang: "Reversible data hiding based on histogram modification of pixel

- differences," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 6, June 2009, pp. 906-910.
- [5] C. C. Lin, W. L. Tai, and C. C. Chang: "Multilevel reversible data hiding based on histogram modification of difference images," *Pattern Recognition*, vol. 41, no. 12, December 2008, pp. 3582-3591.
- [6] Y. P. Hsieh, C. C. Chang, and L. J. Liu, : "A two-codebook combination and three-phase block matching based image-hiding scheme with high embedding capacity," *Pattern Recognition*, vol. 41, no. 10, October 2008, pp. 3104-3113.
- [7] Y. Linde, A. Buzo, and R. M. Gray, An algorithm for vector quantization design, *IEEE Transactions on Communications*, vol. 28, January 1980, pp. 84-95.
- [8] Y. K. Chiang and P. Tsai, "Steganography using overlapping codebook partition," *Signal Processing*, vol. 88, no. 5, May 2008, pp. 1203-1215.
- [9] Y. C. Lin, C. C. Wang, "Digital images watermarking by vector quantization," *Proceedings of National Computer Symposium*, 1999, pp. 76-87.
- [10] S. C. Shie, S. D. Lin, and C. M. Fang, "Adaptive data hiding based on SMVQ prediction," *IEEE Transactions on Information and Systems*, vol. E89-D, no. 1, 2006, pp. 358-362.
- [11] C. C. Chang, Y. H. Chen, Z. H. Wang, and M. C. Li, "A reversible data embedding scheme based on Chinese remainder theorem for VQ index tables," submitted to 2009 ISECS International Colloquium on Computing, Communication, Control, and Management (CCCM 2009), Sanya China, August 2009.