ISSN: 2321-7782 (Online)
Volume 2, Issue 3, March 2014
International Journal of Advance Research in Computer Science and Management Studies
Research Article / Paper / Case Study
Available online at: www.ijarcsms.com

Application of SMVQ Scheme to JDHC Technique in Enhancing Quality of a Digital Image

Eva Edward[1]
PG Scholar
SCAD Engineering College
Tirunelveli – India

J.A.Jevin[2]
Asst.Professor
FX Engineering College
Tirunelveli – India

Abstract: Data hiding in our proposal is applied to achieve the goals of hiding the secret data into a Side-Match Vector Quantization (SMVQ) compressed image and lossless reconstruction of the original image. The secret data is hidden in compressed codes of the cover image during the encoding process of SMVQ such that the interceptors will never capture the secret information. Together the functions of data hiding and image compression can be integrated together into one single module. Complex blocks are used here to control the visual distortion and error diffusion caused by the progressive compression.. It helps provide higher hiding capacity and maintain the size of the encoded image same as that of the original image. We use the original image captured on comparison with the trusted image in applying the combination of both the global and local features. In our experimental results, the image quality of color host image with the secret data embedded is better compared with other methods. This scheme has an overall satisfactory performance for hiding capacity, compression ratio and decompression quality.

Keywords: Data Hiding, SMVQ, Image compression, Lossless reconstruction, Complex blocks.

I. Introduction

Data hiding involves embedding significant data into various forms of digital media such as text, audio, image and video secretly. It has been widely used in applications of copyright protection, fingerprinting and secret communication. The purpose of data hiding techniques is different from that of traditional cryptography or watermarking techniques. Cryptography encrypts messages into meaningless data while watermarking is utilized to protect the copyright. Data hiding technique covers the secret information with the host media as camouflage and is considered as an extension of traditional cryptography.

Data hiding in images

Data hiding in images presents a variety of challenges that arise due to the way the human visual system (HVS) works and the typical modifications that images undergo. Additionally, still images provide a relatively small host signal in which to hide data. A fairly typical 8-bit picture provides approximately 40 kilobytes of data space in which to work. This is equivalent to only around 5 seconds of telephone-quality audio or less than a single frame of NTSC television. Also, it is reasonable to expect that still images will be subject to operations ranging from simple affine transforms to nonlinear transforms such as cropping, blurring, filtering, and lossy compression. Practical data-hiding techniques need to be resistant to as many of these transformations as possible. Despite these challenges, still images are likely candidates for data hiding. There are many attributes of the HVS that are potential candidates for exploitation in a data-hiding system, including our varying sensitivity to contrast as a function of spatial frequency and the masking effect of edges (both in luminance and chrominance).

Data hiding in an image involves embedding a large amount of secret information into a cover image with minimal perceptible degradation of image quality. However, the hiding capacity for secret data and the distortion of the cover image are a tradeoff since more hidden data always results in more degradation on the visual quality of the cover image. Moreover, when data hiding is implemented on the compressed domain of image, the hiding capacity and the visual quality of cover images can be further restricted. During the last decade, vector quantization (VQ) has emerged as an efficient method in image compression. One specific feature of VQ is that high compression ratios are possible with relatively small block sizes.

With the rapid development of Internet technology, people can transmit and share digital content with each other conveniently. In order to guarantee communication efficiency and save network bandwidth, compression techniques can be implemented on digital content to reduce redundancy, and the quality of the decompressed versions should also be preserved. Nowadays, most digital content, especially digital images and videos, are converted into the compressed forms for transmission. Another important issue in an open network environment is how to transmit secret or private data securely. Even though traditional cryptographic methods can encrypt the plaintext into the cipher text , the meaningless random data of the cipher text may also arouse the suspicion from the attacker. To solve this problem, information hiding techniques have been widely developed in academic and industry, which can embed secret data into the cover data imperceptibly. Due to the prevalence of digital images on the Internet, how to compress images and hide secret data into the compressed images efficiently in-depth study.

Applications

Two important uses of data hiding in digital media are to provide proof of the copyright, and assurance of content integrity. Therefore, the data should stay hidden in a host signal, even if that signal is subjected to manipulation as degrading as filtering, re- sampling, cropping, or lossy data compression. Other applications of data hiding, such as the inclusion of augmentation data, need not be invariant to detection or removal, since these data are there for the benefit of both the author and the content consumer. Thus, the techniques used for data hiding vary depending on the quantity of data being hidden and the required invariance of those data to manipulation. Since no one method is capable of achieving all these goals, a class of processes is needed to span the range of possible applications.

II. Existing Studies

Among the many image compression techniques that have been proposed Vector Quantization is one of the popular. In 2003, Du and Hsu proposed an adaptive data hiding method for VQ compressed images [18], which can vary the embedding process according to the amount of hidden data. In this method, the VQ codebook was partitioned into two or more sub code books, and the best match in one of the sub code books was found to hide secret data. In order to increase the embedding capacity, a VQ-based data-hiding scheme by codeword clustering technique was proposed in [19]. The secret data were embedded into the VQ index table by codeword-order-cycle permutation. Inspired by [18], [19], Lin *et al*. adjusted the pre-determined distance threshold according to the required hiding capacity and arranged a number of similar code words in one group to embed the secret sub-message. The search-order coding (SOC) algorithm was proposed by Hsieh and Tsai, which can be utilized to further compress the VQ index table and achieve better performance of the bit rate through searching nearby identical image blocks following a spiral path [21]. Some steganographic schemes were also proposed to embed secret data into SOC compressed codes [22].

However, in all of the above mentioned schemes, data hiding is always conducted after image compression, which means the image compression process and the data hiding process are two independent modules on the server or sender side. Under this circumstance, the attacker may have the opportunity to intercept the compressed image without the watermark information embedded, and the two independent modules may cause a lower efficiency in applications. Thus, in this work, we not only focus on the high hiding capacity and recovery quality, but also establish a joint data-hiding and compression (JDHC) concept and integrate the data hiding and the image compression into a single module seamlessly, which can avoid the risk of the attack from interceptors and increase the implementation efficiency. The proposed JDHC scheme in this paper is based on SMVQ and image in painting. On the sender side, except for the blocks in the leftmost and topmost of the image, each of the other residual

 ISSN: 2321-7782 (Online)

blocks in raster-scanning order can be embedded with secret data and compressed simultaneously by SMVQ or image in painting adaptively according to the current embedding bit. VQ is also utilized for some complex residual blocks to control the visual distortion and error diffusion caused by the progressive compression. After receiving the compressed codes, the receiver can segment the compressed codes into a series of sections by the indicator bits. According to the index values in the segmented sections, the embedded secret bits can be extracted correctly, and the decompression for each block can be achieved successfully.

The rest of this paper is organized as follows. Section III describes the proposed concept, SMVQ Technique in detail. Experimental results and analysis are provided in Section IV, Conclusions are briefed in Section V .

III. Proposed System

In the proposed scheme, rather than two separate modules, only a single module is used to realize the two functions, i.e., image compression and secret data embedding, simultaneously. The image compression in our JDHC scheme is based mainly on the SMVQ mechanism. According to the secret bits for embedding, the image compression based on SMVQ is adjusted adaptively by incorporating the image in painting technique. After receiving the secret embedded and compressed codes of the image, one can extract the embedded secret bits successfully during the image decompression.

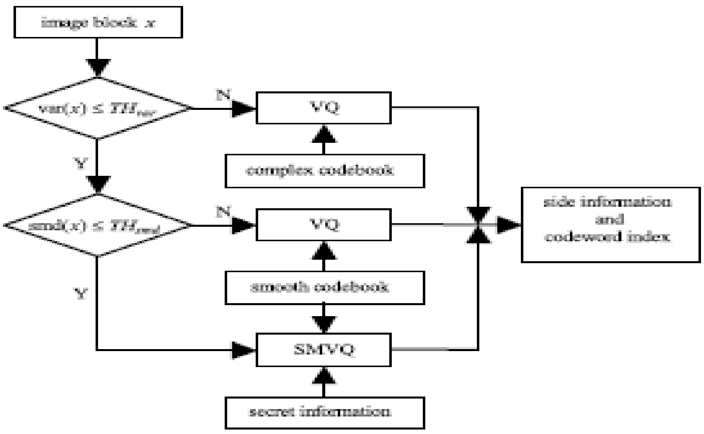

Fig.1. Block Diagram of SMVQ

A. Image Compression Technique

In our scheme, the sender and the receiver both have the same codebook _ with *W* code words, and each codeword length is *n*2. Denote the original uncompressed image sized $M \times N$ as **I**, and it is divided into the non-overlapping $n \times n$ blocks. For simplicity, we assume that *M* and *N* can be divided by *n* with no remainder. Denote all *k* divided blocks in raster scanning order as **B***i, j*, where $k = M \times N/ n2$, $i = 1,2, . \;, M/n$ and j = 1,2, .*N/n*. Before being embedded, the secret bits are scrambled by a secret key to ensure security. The blocks in the leftmost and topmost of the image **I**, i.e.,**B***i*,1(*i* = 1, 2, . . . , *M/n)* and **B**1, *j (j* = 2, 3, . . . , *N/n)*, are encoded by VQ directly and are not used to embed secret bits. The residual blocks are encoded progressively in raster scanning order, and their encoded methods are related to the secret bits for embedding and the correlation between their neighboring blocks. The block diagram of the processing for each residual block is illustrated in Fig 1.

B. Secret Data Embedding Algorithm

The algorithm of secret data embedding into the JPEG compressed image includes the following steps.

1. Apply entropy decoding to the JPEG compressed image. For each block, *Step* 2 and *Step* 3 are then executed.

2. Let *F* be the quantized DCT block with *F*(*i*, *j*) denoting the (*i*, *j*)th entry of *F*, where 0 _ *i*, *j* < 8. For each |*F*(*i*, *j*)| > 1, calculate *E*(*i*, *j*) in Eq. 3. Embed the secret data with length *E*(*i*, *j*) in the LSBs of *F*(*i*, *j*).

3. If the block is a uniform block, a 0 bit is embedded in the last AC coefficient. Otherwise, a 1 bit is embedded in the last AC coefficient.

4. Put the modified quantization table in the header of the JPEG file, and then apply JPEG entropy encoding. This way, the stego-image is produced.

In our scheme, a PDE-based image inpainting method using the fluid dynamics model is adopted [34]. Denote $\mathbf{B}\chi$ as the region including the current block $\mathbf{B}x,y$ that needs compression by inpainting and the available neighboring region of $\mathbf{B}x,y$. Let $B\chi\ (\xi,\ \eta)$ be the gray value of $\mathbf{B}\chi$ in the coordinate (ξ, η). The Laplacian $B\chi\ (\xi,\ \eta)$ is used as a smoothness measure of the region $\mathbf{B}\chi$. By analogizing the inpainting process as the fluid flowing and imitating the practice of a traditional art professional in the manual retouching, details in the unknown region may be created through propagating the available information in the surrounding areas into the unknown region along isophote directions. The field of isophote is defined as:

$$\nabla^{\perp} B_{\chi}(\xi, \eta) = \left(-\frac{\partial}{\partial \xi}\mathbf{i} + \frac{\partial}{\partial \eta}\mathbf{j}\right) B_{\chi}(\xi, \eta). \qquad (1)$$

Where **i** and **j** are unit directional vectors. Clearly, variations in image gray values are minimal along the isophote directions.Having finished the inpainting process, $\nabla\perp B\chi\ (\xi,\ \eta)$ should be normal to the gradient of the smoothness $B\chi(\xi,\ \eta)$:

$$\nabla[\Delta B_{\chi}(\xi, \eta)] \cdot \nabla^{\perp} B_{\chi}(\xi, \eta) = 0. \qquad (2)$$

The scalar product in the above equation indicates projection of the smoothness change onto the direction of isophote. If we let the projection value be equal to the change of image gray values with respect to time t, the following PDE can be acquired.

$$\frac{\partial}{\partial t} B_{\chi}(\xi, \eta) = \nabla[\Delta B_{\chi}(\xi, \eta)] \cdot \nabla^{\perp} B_{\chi}(\xi, \eta), \quad \forall(\xi, \eta) \in \mathbf{B}_{x,y} \qquad (3)$$

By using the finite difference method, we can obtain a discretized iteration algorithm to solve the PDE. Information propagation of this inpainting model finishes until the gray values in $\mathbf{B}x,y$ reach stable state. Consequently, the recovered effectively without serious blurring on edges.Consequently, when $s = 1$, in order to indicate that block $\mathbf{B}x,y$ is processed by inpainting and differentiate from the index λ produced by SMVQ, the index value R occupying log2$(R + 1)$ bits is used as the compressed code of $\mathbf{B}x,y$ $(R > \lambda)$. For simplicity, we assume that log2$(R + 1)$ is an integer and _log2 R_ ≡ log2$(R + 1)$. After the current block $\mathbf{B}x,y$ is processed, the following block in raster-scanning order repeats the above procedure. Note that each processed block should be substituted with its corresponding decompressed result, i.e., VQ codeword, SMVQ codeword, or inpainting result, for the success of progressive mechanism. The whole procedure of image compression and secret data embedding finishes until all residual blocks are processed. Then, the compressed codes of all image blocks are concatenated and transmitted to the receiver side.

C. Data Extracting Algorithm

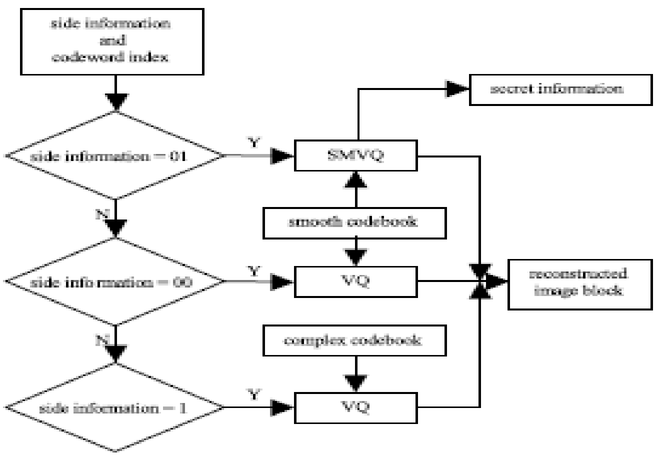

Fig.2. Block Diagram of Data extracting algorithm

The block diagram of the proposed data extracting algorithm is illustrated in Fig.2. The details of the extracting algorithm at the receiver are given as follows.

Input: the smooth codebook $Y0 = \{y00, y01, \ldots, y0p-1\}$, the complex codebook $Y1 = \{y10, y11, \ldots, y1q-1\}$, and the received bit stream (including side information and VQ indices). Relative parameters: the side-match state code book size r.

Step 1. Let the currently processed (decoded) image block be xi, extract the first bit from the received bit stream.

Step 2. If the extracted bit is '0', then the master codebook for xi is codebook $Y0$ and examine the next bit in the received bit stream. Otherwise, the master codebook for xi is codebook $Y1$ and go to step 4.

Step 3. If the next bit is '0', remove this next bit and extract the succeeding log2 p bits from the received bit stream. Use the log2 p bits and codebook $Y0$ to decode block xi by VQ. Otherwise, remove this next bit and extract the succeeding log2 r bits from the received bit stream. Generate the side match state codebook $S\ i$ from codebook $Y0$, and decode xi, using the log2 r bits based on SMVQ. Collect the log2 r bits as part of the secret bit stream.

Step 4. Extract the succeeding log2 q bits from the received bit stream. Use the log2 q bits and codebook $Y1$ to decode block xi by VQ directly.

Step 5. If there exists image blocks to be processed, go to step 1.

Step 6. Merge the collected bits to obtain the whole secret bit stream.

Fig.3. Six standard test images

Therefore, besides the image compression, the proposed scheme can achieve the function of data hiding that can be used for covert communication of secret data. The sender can transmit the secret data securely through the image compressed codes, and the receiver can extract the hidden secret data effectively from the received compressed codes to complete the process of covert communication. Additionally, because the secret data extraction in our scheme can be conducted independently with the decompression process, the receiver obtains the secret bits at any time if he or she preserves the compressed codes. The proposed scheme can also be used for the integrity authentication of the images, in which the secret bits for embedding can be regarded as the hash of the image principle contents. The receiver can calculate the hash of the principle contents for the decompressed image, and then compare this calculated hash with the extracted secret bits (embedded hash) to judge the integrity of the received compressed codes and the corresponding decompressed image. If the two hashes are equal, it means the image is authenticated. Otherwise, the received compressed codes must be tampered.

IV. Experimental Results and Analysis

Experiments were conducted on a group of gray-level images to verify the effectiveness of the proposed scheme. In the experiment, the sizes of the divided non-overlapping image blocks were 4 × 4, i.e., $n = 4$. Accordingly, the length of each codeword in the used VQ codebooks was 16. The parameter R was set to 15. Six standard, 512 × 512 test images, i.e., Lena, Airplane, Lake, Peppers, Sailboat, and Tiffany, are shown in Figure 3. Besides these six standard images, the uncompressed color image database (UCID) that contains 1338 various color images with sizes of 512 × 384 was also adopted. The luminance components of the color images in this database were extracted and used in the experiments. The performances of compression ratio, decompression quality, and hiding capacity for the proposed scheme were evaluated. All experiments were implemented on a computer with a 3.00 GHz AMD Phenom II processor, 2.00 GB memory, and Windows 7 operating system, and the programming environment was Matlab 7.

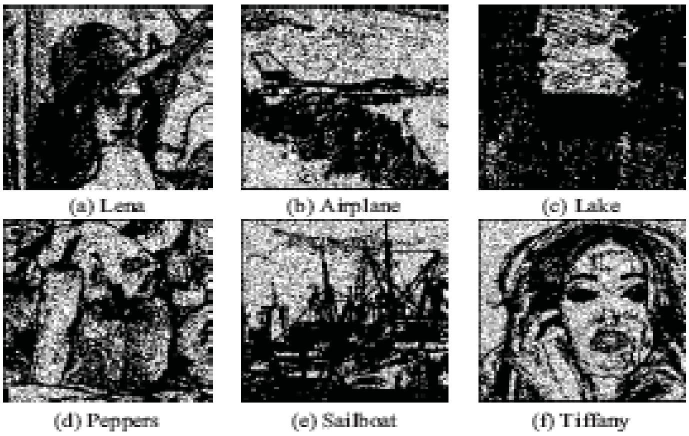

Fig.4. Labels of Image blocks with different Thresholds(T)

In the proposed scheme, the hiding capacity and the visual quality of cover images are mainly affected by the three parameters, the variance threshold *THvar*, the side match distortion threshold *THsmd*, and the side-match state code book size *r*. These parameters are adjusted based on the amount of hidden data and the characteristic of cover image. They can be used as keys for secret data extraction as well. If *THvar* is set with a larger value, more blocks will be treated as smooth blocks and, consequently, more secret data can be hidden into a cover image. However, since more blocks are directly predicted by the proposed scheme, the visual quality of cover image will be degraded. If *THsmd* is given as a larger value, more smooth blocks will be selected for hiding data. Therefore, the hiding capacity increases and the visual quality decrease for the cover image. If *r* is assigned to be a larger value, more code words are included into the state codebook and the selected smooth blocks will be encoded (predicted) more randomly. Accordingly, the visual quality of cover image degrades while the hiding capacity increases.

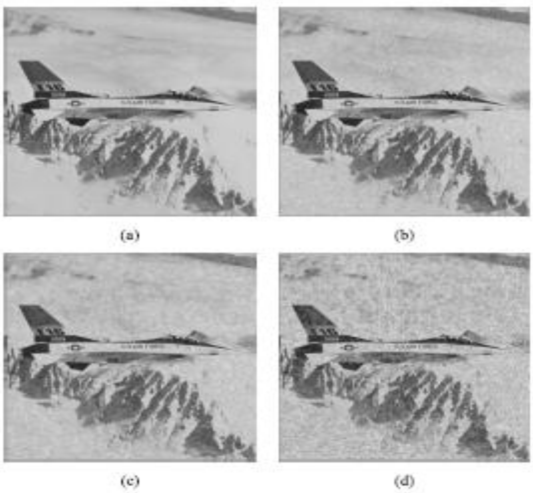

Fig.5. Hiding results with various Hidden data

Besides, we employed the peak signal-to-noise ratio (PSNR) as a measure of the stego-image quality. It is defined as follows:

$$\text{PSNR} = 10 \times \log_{10} \frac{255^2}{MSE} \text{dB} \qquad (4)$$

Where *MSE* is the mean-square error. For an *N* × *N* image, its *MSE* is defined as,

$$MSE = \left(\frac{1}{\text{N}}\right)^2 \times \sum_{i=1}^{N}\sum_{j=1}^{N}\left(x[i,j] - \bar{x}[i,j]\right)^2. \qquad (5)$$

Here, *x*[*i, j*] and ¯*x*[*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 stego-image preserves the original image quality better.

Our method employs the capacity factors α to control the level of embedding capacity. Users can adjust it to balance between the image quality (PSNR) and the embedding capacity. If the capacity factor is selected as a large number, then the embedding capacity can be raised, but the cost is that the compression ratio of the image gets low. Through quite a number of experiments, the capacity factor α is finally selected for uniform blocks, and $1.2 * \alpha$ for non-uniform blocks. They apply to a wide variety of images. In addition, we also conducted some experiments to show the flexibility of our method.

	Capacity Factor (α)	Embedded data (bits)	Compressed file (bytes)	Embedding Capacity	PSNR (dB)
JPEG	–	–	87891	–	45.85
Jpeg–Jsteg	–	67000	87552	9.56%	44.85
Chang *et al.*	–	212992	88912	29.9%	33.71
Proposed method	0.18	68871	89436	9.63%	43.97
	0.36	107035	89235	15.0%	43.06
	0.5	122032	89199	17.1%	42.53
	0.75	134437	89185	18.8%	41.70
	1.0	142820	89160	20.0%	40.84

Table.1.Comparison of embedding capacity in Lena

	Capacity Factor (α)	Embedded data (bits)	Compressed file (bytes)	Embedding Capacity	PSNR (dB)
JPEG	–	–	45259	–	41.15
Jpeg–Jsteg	–	32868	45060	9.11%	40.07
Chang *et al.*	–	212992	89018	29.9%	28.16
Proposed method	0.2	35678	48137	9.26%	39.38
	0.4	53532	48137	13.9%	38.12
	0.6	58254	48101	15.1%	37.32
	0.8	65457	48222	17.0%	36.29
	1.0	66363	48213	17.2%	35.77

Table.2.Comparison of embedding capacity in Airplane

Experimental results in Tables 1 are for the Lena image compressed by JPEG with a *Q* factor of 5. Tables 2 is for the Jet image which compressed by JPEG with a *Q* factor of 5. We selected a proper capacity factor so that our embedding capacity is about the same as that of Jpeg–Jsteg, and the results showed that the stego-image quality of our method was as good as that of Jpeg–Jsteg. However, the size of the compressed file produced by our method was a bit larger than that of Jpeg–Jsteg because we embedded one bit in the last AC component to indicate the block type. But the size of the compressed file did not expand when the embedding capacity increased. Generally speaking, our experimental results show that the proposed method is able to achieve the embedding capacity of around 20% of the stego-image with little or no noticeable degradation of image quality when the compression ratio is low. Of course the embedding capacity is lower when the compression ratio increases. We can find from Tables 1 and 2 that, compared with the standard VQ method, the proposed scheme can achieve the comparable visual quality of decompressed images and obtain greater compression ratios. The standard SMVQ method has the exactly same compression ratio with the proposed scheme. However, the standard SMVQ method can not carry secret information within its compressed codes. Our scheme not only can carry a large amount of secret bits within the compressed codes as shown in Table I, but also achieves higher decompression quality than the standard SMVQ method due to the satisfactory recovery property of image inpainting. Although the scheme in [27] can embed more secret bits than our scheme, our scheme has better performance of compression ratio and significantly higher decompression quality than [27]. Furthermore, the proposed scheme can realize data-hiding and image compression simultaneously in a single module, i.e., joint data hiding and compression.

V. Conclusion

In this paper, a high capacity data hiding method is proposed. Our method embeds a joint data-hiding and compression scheme by using SMVQ and PDE-based image inpainting. The blocks, except for those in the leftmost and topmost of the image, can be embedded with secret data and compressed simultaneously, and the adopted compression method switches between SMVQ and image inpainting adaptively according to the embedding bits. VQ is also utilized for some complex blocks to control the visual distortion and error diffusion. On the receiver side, after segmenting the compressed codes into a series of sections by the indicator bits, the embedded secret bits can be easily extracted according to the index values in the segmented sections, and the decompression for all blocks can also be achieved successfully by VQ, SMVQ, and image inpainting. Ours is an adaptive data hiding method with which one can adjust capacity factor to balance between the image quality and the embedding capacity dynamically. Furthermore, the proposed method is securer than most of its predecessors.

Experimental results show that our method indeed provides acceptable image quality and adjustable embedding capacity. High embedding capacity of around 20% of the JPEG compressed image size is achieved with little noticeable degradation of image quality when the compression ratio is low. The proposed method is very practical for most image files that are stored and transmitted in the JPEG format.

References

1. W. B. Pennebaker and J. L. Mitchell, The JPEG Still Image Data Compression Standard. New York, NY, USA: Reinhold, 1993.
2. D. S. Taubman and M. W. Marcellin, JPEG2000: Image Compression Fundamentals Standards and Practice. Norwell, MA, USA: Kluwer, 2002.
3. A. Gersho and R. M. Gray, Vector Quantization and Signal Compression. Norwell, MA, USA: Kluwer, 1992.
4. N. M. Nasrabadi and R. King, “Image coding using vector quantization: A review,” IEEE Trans. Commun., vol. 36, no. 8, pp. 957–971, Aug. 1988.
5. Announcing the Advanced Encryption Standard (AES), National Institute of Standards & Technology, Gaithersburg, MD, USA, Nov. 2001.
6. R. L. Rivest, A. Shamir, and L. Adleman, “A method for obtaining digital signatures and public-key cryptosystems,” Commun. ACM, vol. 21, no. 2, pp. 120–126, 1978.
7. F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn, “Information hiding survey,” Proc. IEEE, vol. 87, no. 7, pp. 1062–1078, Jul. 1999.
8. C. D. Vleeschouwer, J. F. Delaigle, and B Macq, “Invisibility and application functionalities in perceptual watermarking: An overview,” Proc. IEEE, vol. 90, no. 1, pp. 64–77, Jan. 2002.
9. C. C. Chang, T. S. Chen, and L. Z. Chung, “A steganographic method based upon JPEG and quantization table modification,” Inf. Sci., vol. 141, no. 1, pp. 123–138, 2002.
10. H. W. Tseng and C. C. Chang, “High capacity data hiding in JPEGcompressed images,” Informatica, vol. 15, no. 1, pp. 127–142, 2004.
11. P. C. Su and C. C. Kuo, “Steganography in JPEG2000 compressed images,” IEEE Trans. Consum. Electron., vol. 49, no. 4, pp. 824–832, Nov. 2003.
12. W. J. Wang, C. T. Huang, and S. J. Wang, “VQ applications in steganographic data hiding upon multimedia images,” IEEE Syst. J., vol. 5, no. 4, pp. 528–537, Dec. 2011.
13. Y. C. Hu, “High-capacity image hiding scheme based on vector quantization,” Pattern Recognit., vol. 39, no. 9, pp. 1715–1724, 2006.
14. 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 Recognit., vol. 41, no. 10, pp. 3104–3113,2008.
15. C. H. Yang and Y. C. Lin, “Fractal curves to improve the reversible data embedding for VQ-indexes based on locally adaptive coding,” J. Vis.Commun. Image Represent., vol. 21, no. 4, pp. 334–342, 2010.
16. Y. Linde, A. Buzo, and R. M. Gray, “An algorithm for vector quantization design,” IEEE Trans. Commun., vol. 28, no. 1, pp. 84–95, Jan. 1980.

17. C. C. Chang and W. C. Wu, "Fast planar-oriented ripple search algorithm for hyperspace VQ codebook," IEEE Trans. Image Process., vol. 16, no. 6, pp. 1538–1547, Jun. 2007.
18. W. C. Du and W. J. Hsu, "Adaptive data hiding based on VQ compressed images," IEE Proc. Vis., Image Signal Process., vol. 150, no. 4, pp. 233–238, Aug. 2003.
19. C. C. Chang and W. C. Wu, "Hiding secret data adaptively in vector quantisation index tables," IEE Proc. Vis., Image Signal Process., vol. 153, no. 5, pp. 589–597, Oct. 2006.
20. C. C. Lin, S. C. Chen, and N. L. Hsueh, "Adaptive embedding techniques for VQ-compressed images," Inf. Sci., vol. 179, no. 3, pp. 140–149, 2009.
21. C. H. Hsieh and J. C. Tsai, "Lossless compression of VQ index with search-order coding," IEEE Trans. Image Process., vol. 5, no. 11, pp. 1579–1582, Nov. 1996.
22. C. C. Lee, W. H. Ku, and S. Y. Huang, "A new steganographic scheme based on vector quantisation and search-order coding," IET Image Process., vol. 3, no. 4, pp. 243–248, 2009.
23. S. C. Shie and S. D. Lin, "Data hiding based on compressed VQ indices of images," Comput. Standards Inter., vol. 31, no. 6, pp. 1143–1149, 2009.
24. C. C. Chang, G. M. Chen, and M. H. Lin, "Information hiding based on search-order coding for VQ indices," Pattern Recognit. Lett., vol. 25, no. 11, pp. 1253–1261, 2004. 978 IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 23, NO. 3, MARCH 2014
25. T. Kim, "Side match and overlap match vector quantizers for images," IEEE Trans. Image Process., vol. 1, no. 2, pp. 170–185, Apr. 1992.
26. C. C. Chang, W. L. Tai, and C. C. Lin, "A reversible data hiding scheme based on side match vector quantization," IEEE Trans. Circuits Syst. Video Technol., vol. 16, no. 10, pp. 1301–1308, Oct. 2006.
27. C. C. Chen and C. C. Chang, "High capacity SMVQ-based hiding scheme using adaptive index," Signal Process., vol. 90, no. 7, pp. 2141–2149, 2010.
28. L. S. Chen and J. C. Lin, "Steganography scheme based on side match vector quantization," Opt. Eng., vol. 49, no. 3, pp. 0370081–0370087, 2010.
29. S. C. Shie and J. H. Jiang, "Reversible and high-payload image steganographic scheme based on side-match vector quantization," Signal Process., vol. 92, no. 9, pp. 2332–2338, 2012.
30. C. F. Lee, H. L. Chen, and S. H. Lai, "An adaptive data hiding scheme with high embedding capacity and visual image quality based on SMVQ prediction through classification codebooks," Image Vis. Comput., vol. 28, no. 8, pp. 1293–1302, 2010.
31. P. Tsai, "Histogram-based reversible data hiding for vector quantisationcompressed images," IET Image Process., vol. 3, no. 2, pp. 100–114, 2009.
32. Z. X. Qian and X. P. Zhang, "Lossless data hiding in JPEG bitstream," J. Syst. Softw., vol. 85, no. 2, pp. 309–313, 2012.
33. Z. C. Ni, Y. Q. Shi, N. Ansari, and W. Su, "Reversible data hiding," IEEE Trans. Circuits Syst. Video Technol., vol. 16, no. 3, pp. 354–362, Mar. 2006.
34. M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," in Proc. 27th Int. Conf. Comput. Graph. Interact. Tech., New Orleans, LA, USA, Jul. 2000, pp. 417–424.
35. C. Qin, S. Wang, and X. Zhang, "Simultaneous inpainting for image structure and texture using anisotropic heat transfer model," Multimedia Tools Appl., vol. 56, no. 3, pp. 469–483, 2012.

AUTHOR(S) PROFILE

Mrs.Eva Edward completed her Bachelorof Technology(Information Technology) in Jeya Engineering college,Chennai in 2012 and pursuing her Masters in Engineering(Computer science) in SCAD Engineering College, Tirunelveli and would be awarded the degree in 2014.

Her interests in research include Image processing and medical imaging.

Mr.Jevin completed his Bachelor of Engineering(Computer science) in KSR college of Engineering and Technology,Tiruchengode in 2006 and completed his Masters in Engineering(Computer science) in University Department, Anna university, Tirunelveli in 2012

He is working as an Assistant Professor in Francis Xavier Engineering College, Tirunelveli from 2013.

His research interests include Medical imaging and Cyber security.

