Information Sciences 181 (2011) 3379–3396

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Tabu search based multi-watermarks embedding algorithm with multiple description coding

Hsiang-Cheh Huang [a,*], Shu-Chuan Chu [b], Jeng-Shyang Pan [c], Chun-Yen Huang [c], Bin-Yih Liao [c]

[a] National University of Kaohsiung, 700 University Rd., Kaohsiung 811, Taiwan, ROC
[b] School of Computer Science, Engineering and Mathematics, Flinders University of South Australia, Australia
[c] National Kaohsiung University of Applied Sciences, 415 Chien-Kung Rd., Kaohsiung 807, Taiwan, ROC

ARTICLE INFO

Article history:
Received 5 February 2005
Received in revised form 2 April 2011
Accepted 6 April 2011
Available online 15 April 2011

Keywords:
Watermarking
Error resilience
Multiple description coding
Vector quantization
Optimization
Tabu search

ABSTRACT

Digital watermarking is a useful solution for digital rights management systems, and it has been a popular research topic in the last decade. Most watermarking related literature focuses on how to resist deliberate attacks by applying benchmarks to watermarked media that assess the effectiveness of the watermarking algorithm. Only a few papers have concentrated on the error-resilient transmission of watermarked media. In this paper, we propose an innovative algorithm for vector quantization (VQ) based image watermarking, which is suitable for error-resilient transmission over noisy channels. By incorporating watermarking with multiple description coding (MDC), the scheme we propose to embed multiple watermarks can effectively overcome channel impairments while retaining the capability for copyright and ownership protection. In addition, we employ an optimization technique, called tabu search, to optimize both the watermarked image quality and the robustness of the extracted watermarks. We have obtained promising simulation results that demonstrate the utility and practicality of our algorithm.

1. Introduction

Digital watermarking [21,34,43], in conjunction with encryption [1,25], is a useful solution for digital rights management (DRM) systems. It embeds secret information into the digital contents to protect the intellectual property [20,42] or the ownership of the original multimedia sources [4,11]. Typical watermarking schemes embed the watermark by altering coefficients related to the original source in some specific domain, including the spatial-domain methods [39], transform-domain techniques using discrete cosine transform (DCT) [12,24], discrete wavelet transform (DWT) [19] and discrete Fourier transform (DFT) [40], or VQ domain schemes [27,36]. These schemes have been popular research topics in the last decade.

There are many metrics to measure the effectiveness of a watermarking algorithm. From the algorithm design viewpoint, the three most critical requirements are: *watermark imperceptibility*, *watermark robustness*, and *watermark capacity*. Although these requirements are all very desirable, as pointed out in the literature [3,28,30,50], they influence, or even conflict, with the remaining requirements. Fixing one dimension, the other two conflict with each other, and some tradeoff or compromise must be reached [29]. The tradeoff relationships can affect all three parameters.

* Corresponding author. Tel.: +886 918 952075.
E-mail address: huang.hc@gmail.com (H.-C. Huang).

0020-0255/$ - see front matter © 2011 Elsevier Inc. All rights reserved.
doi:10.1016/j.ins.2011.04.007

(1) Watermark imperceptibility refers to whether the viewer can perceive the existence of an embedded watermark. To make the watermark imperceptible, two situations need to be considered.
   (a) The number of watermark bits embedded must be less than a certain amount to make the watermark imperceptible. The theoretical bound for this amount, called the watermark capacity, is described in item (3) below, and is derived in the literature [23]. Enhanced reliability can be expected by embedding multiple watermarks into the original multimedia [41]. Conversely, embedding fewer bits implies less robustness in the watermarking algorithm.
   (b) Imperceptible watermarking indicates the least modification of the original media. The watermarked image quality is supposed to be within the just noticeable distortion (JND) region [17]. In this regard, the commonly employed scheme in the spatial domain is to embed the watermark into the least significant bits (LSB); in the frequency domain, including the DCT and DWT domains, people tend to embed the watermark into the higher frequency band coefficients. This approach is motivated by the fact that most of the energy in the multimedia content, such as an image, is concentrated in the lower frequency band coefficients [15]. Hence, embedding the watermark bits into higher frequency band coefficients results in less modification to the original source. However,this technique renders the watermark vulnerable to common image processing steps such as low-pass filtering (LPF).

(2) Watermark robustness refers to the capability of the watermarked media to withstand intentional or unintentional media processing, called *attacks*, including filtering, transcoding, resizing, or rotation. There are benchmarks to examine the watermark robustness objectively, such as Stirmark [37]. It is generally agreed that robustness plays an important role in the design of a watermarking algorithm. Heuristically, improving robustness requires embedding the watermark into the most significant bits (MSB) in the spatial domain or the lower frequency band coefficients in the transform domain. However, this process can seriously degrade the watermarked image quality and alert others about the existence of the watermark. Consequently, to satisfy the tradeoff between watermark imperceptibility and watermark robustness, the watermark is embedded into the "middle frequency bands" in the transform domain [13].

(3) Watermark capacity is determined by the number of bits embedded in the original media; that is, the size of the watermark. Generally speaking, when more bits can be embedded, the algorithm is supposed to be more robust; however, under such a condition, the quality of watermarked media must be degraded, and hence, the existence of the watermark becomes more perceptible. Authors in [3,23] derived theoretical bounds for watermark capacity. For image watermarking, the watermark capacity is generally a constant size. Thus, only watermark imperceptibility and watermark robustness need to be considered for the design of the algorithm.

After considering the three fundamental requirements for watermarking, techniques for optimizing non-linear functions with multiple variables [5,45] can be considered to search for the optimized outcome. In this paper, we fix the watermark capacity, and employ tabu search [9,14] to find a tradeoff between watermark imperceptibility and watermark robustness. Tabu search is an evolutionary algorithm, characterized by the use of a flexible memory. It is able to eliminate local minima and to search areas beyond a local minimum. Some research papers and applications concentrate on designing watermarking algorithms with tabu search for audio signals [44] and images [2,6,22,31], which are similar to the goal for optimization presented in this paper.

Before going into more details of our proposed algorithm, we would like to briefly summarize the existing methods [2,6,22,31,44] to design a watermarking algorithm with tabu search. These methods are closely related to the main theme of this paper. We also discuss common items and differences, in addition to the advantages and disadvantages, in the design of each algorithm.

In [44], the authors present a wavelet-based watermarking algorithm for audio signals with tabu search. Both [44] and this paper share similar concepts in designing the fitness for optimizing the different goals. On the one hand, the audio quality and the robustness of the watermarking algorithm under intentional attacks are considered in [44]. The authors use tabu search to design a robust audio watermarking algorithm that preserves good quality in the watermarked audio signal, and yields better robustness for the extracted watermark. On the other hand, we consider image quality and watermark robustness under unintentional attacks in this paper. The concepts for designing our fitness function with tabu search are similar to those presented in [44]. However, we introduce another major theme in this paper, called "multiple description coding" (MDC). In our work, image quality is enhanced by both tabu search and error-resilient coding, in the form of MDC. Background knowledge of MDC will be covered in Section 3.

In [2], the author proposes a wavelet-based robust watermarking algorithm for still images. The author claims that the algorithm can cope with JPEG compression and cropping attacks. Tabu search is employed to find the region- of-interest (ROI) of the original image, and then the watermark is embedded into the ROI portion of the image with the wavelet transform. In comparison with our paper, both the concept for designing the fitness function, and the goal for developing the watermarking algorithm are different. Though both [2] and this paper use tabu search, the goals to be optimized are significantly different.

In [22], the authors propose a VQ-based robust watermarking algorithm suitable for transmitting watermarked VQ indices over a binary symmetric channel (BSC). The authors integrate their watermark embedding scheme into the codebook design problem. They employ two conventional schemes for VQ codebook design with tabu search: the index assignment (IA) scheme and the energy allocation (EA) scheme. In this paper, we employ an existing codebook trained by the well-known LBG algorithm. Tabu search is employed to optimize both the watermarked image quality and the watermark

robustness. Like our approach, [22] also use tabu search to train the parameters in the fitness function. However, the methodologies for designing the algorithms in the two papers are dissimilar.

In [6] and [31], the authors propose a multiple watermarking scheme for embedding two watermarks: one in the spatial domain and the other in the transform domain. They employ tabu search to develop algorithms for image-based robust watermarking by embedding one visible watermark in the spatial domain and one imperceptible watermark in the transform domain. Both the watermarked image quality and the capability to resist the JPEG attack are optimized with tabu search. Because we use a different algorithm, VQ-based invisible watermarking with the error-resilient capability of MDC, we infer that the algorithms proposed in [6,31] are completely different from our method.

We will briefly compare the existing VQ-based watermarking algorithms to the one we proposed. In this paper, we propose an innovative, VQ-based image watermarking algorithm that is suitable for error-resilient transmission over noisy channels. A review of the research literature related to digital watermarking indicated that only a few authors have concentrated on the error-resilient transmission of watermarked media [38,47]. Most watermarking related literature focuses on how to resist intentional attacks. For practical implementations with VQ-based watermarking, existing methods are compared to the proposed algorithm in this paper. We conduct researches in transmitting watermarked images over binary symmetric channels (BSC) [35] and packet-loss channels [32,33]. In general, researchers use the BSC or packet-loss channel to simulate the transmission of baseband signals. In [32,33] and in this paper, algorithms for transmission over packet-loss channels are developed. Because these papers [32,33,35] focus on transmitting the watermarked image over lossy channels, this could be regarded as a new branch for watermarking research.

In light of the discussions above, we would like to point out that despite superficial similarities between our proposed algorithm and prior research work, there are still significant innovations presented in this paper. Besides using tabu search for optimization, we provide an additional method, MDC, to help protect the ownership of the original image, and simultaneously retain the reconstructed image. Our proposed algorithm combines watermarking and error-resilient coding, and the results have led to promising results. Based on the experience of our previous works, we employ tabu search in this paper to obtain an optimized solution suitable for the transmission of watermarked images over lossy channels.

This paper is organized as follows. We describe the fundamental concepts of VQ and MDC in Section 2 and Section 3, respectively. Section 4 concentrates on quantization-based MDC, which is an integration between the concepts in Sections 2 and 3. Section 5 and Section 6 present the watermark embedding and extraction algorithms. We demonstrate an example in Section 7 to improve understanding of our proposed algorithm. Optimization of our algorithm with tabu search is described in Section 8. We also study two related algorithms published in the literature and briefly describe them in Section 9. Simulation results are shown in Section 10, and comparisons between our algorithm and those in Section 9 are presented. Finally, we conclude this paper in Section 11.

2. Fundamentals of vector quantization

Vector quantization [8], one of the important techniques in multimedia compression, has received considerable attention since the 1980s. As an extension to scalar quantization, vector quantization works on vectors of raw data. A vector can be a small block of image data, for example, the grey-level values of a 4×4 pixel image block forms a 16-dimensional vector. Fig. 1 gives a block diagram illustration of the operation of vector quantization compression.

The original image $\boldsymbol{X}$ is composed of the combination of all of the input vectors, $\boldsymbol{X}_k$, $\forall k$. The codeword search process looks for a "nearest codeword," c_i, from the codebook for the given input vector $\boldsymbol{X}_k$ with the Euclidean distance measure. The codebook $\boldsymbol{C}$ with size L is composed of L elements, or the codewords with the representation $\boldsymbol{C} = \{c_0, c_1, \ldots, c_{L-1}\}$.

The codebook size, or the number of codewords in a codebook, is a tradeoff between the reconstructed image quality and the compression rate. The codewords in the codebook decide the subsequent compression distortion. A dedicated procedure requires to generation the appropriate codebook.

3. Background of multiple description coding and its generic model

During transmissions of data, loss of data is inevitable due to channel error or packet loss in various types of transmission channels. In contrast to the conventional schemes such as progressive transmission, multiple description coding (MDC) offers an alternative method for the effective delivery of compressed multimedia information.

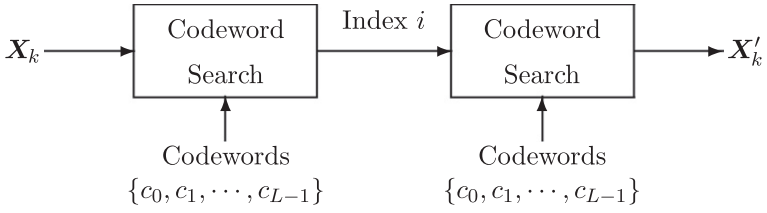

Fig. 1. A block diagram for vector quantization. All the L codewords form a codebook of size L, that is, $\boldsymbol{C} = \{c_0, c_1, \ldots, c_{L-1}\}$. $\boldsymbol{X}_k$ means the vector, or a small block in the original image $\boldsymbol{X}$, and $\boldsymbol{X}'_k$ is the reconstructed vector.

MDC is an error-resilient coding technique, which can be referred to as a source coding method for a channel whose end-to-end performance includes uncorrected erasures. This channel is encountered in a packet communication system that has effective error detection but does not have the features that permit the retransmission of incorrect or lost packets. MDC uses diversity to overcome channel impairments so that a decoder that receives an arbitrary subset of the channels may reproduce a useful reconstruction [16]. Information-theoretic issues of MDC have been studied extensively since the early 1980s [7,51]. In multiple description (MD) coders, the same source material is coded into several pieces of data, called *descriptions*, such that each description can be decoded independently to obtain minimum fidelity. This information is also combined with other descriptions to achieve a better quality. The goal of MDC, and channel coding in general, is the making effective transmission of data, and MDC offers a totally different perspective from that of channel coding [26].

MDC is suitable for transmission over noisy channels with long bursts of errors. To gain robustness of the loss in spite of descriptions, MDC must sacrifice some compression efficiency while still retaining the capability for error resilience. Therefore, correlations between descriptions should be intentionally induced to achieve this goal. Fig. 2 depicts the generic model for MD source coding with two channels and three decoders. The Encoder is denoted by α_0. Decoder 0, denoted by β_0, is called the *central decoder*, and Decoders 1 and 2, are denoted by β_1 and β_2 respectively, are the *side decoders*. The Euclidean distance between $\boldsymbol{X}$ and $\hat{\boldsymbol{X}}^{(0)}$ is the *central distortion*, while the errors between $\boldsymbol{X}$ and $\hat{\boldsymbol{X}}^{(i)}, i = 1, 2,$ are the *side distortions*. It suggests a situation in which there are three separate users or three classes of users, which could arise when broadcasting on two channels. The same abstraction holds if there is a single user that can be in one of three states depending on which descriptions are received. Generally speaking, if we extend the number of transmission channels in Fig. 2 to P, there will be $(2^P - 1)$ receivers that decode with different number of descriptions received and reconstruct the image with different quality levels.

In addition to theoretical research, it is also important to devise practical designs to make MDC applicable to the situation depicted in Fig. 2. Practical applications and implementations of MDC emerged in the 1990s. Two major categories for MDC applications are: (i) *quantization based* schemes, such as Multiple Description Scalar Quantization (MDSQ) [46] and Multiple Description Vector Quantization (MDVQ) [10], and (ii) *transform-domain based* schemes, called Multiple Description Transform Coding (MDTC) [48,49]. In this paper, we focus on quantization based MD schemes for watermarking. Operations and realizations of quantization-based MDC will be described in Section 4. For quantization-based MDC, redundancies induced between descriptions are controlled inherently. Thus, only the watermarked image quality and the robustness should be taken care of, and this is the motivation to use MDVQ for watermarking. On the other hand, the idea for watermarking with MDTC is basically the same as that with MDVQ. For MDTC, correlations between different descriptions are controlled by the users at the encoder. After transmission, two descriptions are composed to reconstruct one 8×8 block at the decoder. Greater correlation between descriptions leads to better resilience to channel error in the reconstructed image, at the expense of degraded performance in compression. Taking watermark embedding into account, three parameters, including (a) the watermarked image quality, (b) the robustness, and (c) the correlation coefficient between descriptions should all be considered. Because one 8×8 block in the original image corresponds to one correlation coefficient, the design of the fitness function may become a difficult task. The vast quantity of correlation coefficients may impair the convergence of the training with tabu search. Thus, watermarking with MDTC is beyond the scope of this paper because the design and implementation of the algorithm need to be performed by other means. We will concentrate on watermarking with quantization-based MDC in this paper.

4. Quantization-based multiple description coding

Applications of MDC focus on error concealment and error resilience. In this paper, we introduce the idea of applying MDC with watermarking schemes to cover both the reconstructed image quality after reception, and the ownership of the original image.

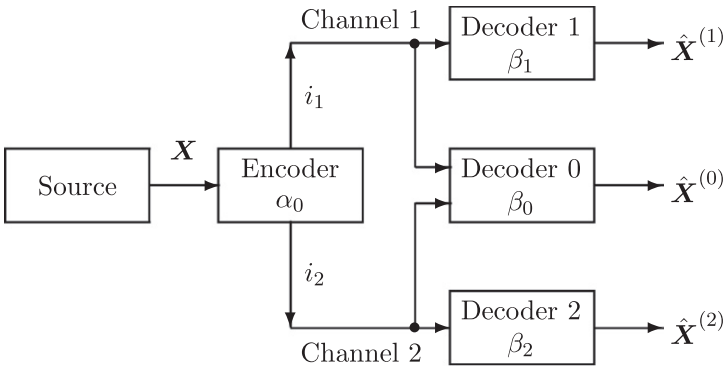

Fig. 2. The generic model for MD source coding with two channels and three receivers. The general case has P channels and $(2^P - 1)$ receivers.

Fig. 2 is the generic structure of MDC, which can also be applied to quantization based multiple-description (MD). For example, the first practical design is based on scalar quantization. Employing different quantization levels in the MD structure is a straightforward solution. In addition, MDSQ is flexible in that it allows a designer to choose the relative importance of the central distortion and each side distortion. The basic structure for MDSQ with two descriptions is illustrated in Fig. 3. The input $\boldsymbol{X}$ is first quantized into a scalar i, using the scalar quantization function α. Then, the encoder produces a pair of quantization indices (i_1,i_2) from each scalar sample i. This step is called "index assignment." In [46], the author described in detail how to perform quantization using an invertible function and introduced a convenient way to visualize the encoding operation. The method to turn the scalar sample into the indices by carrying out the index assignment process was also developed. The encoding is first decomposed into two steps:

$$\alpha_0 = l \circ \alpha. \tag{1}$$

The initial encoder α is a regular scalar quantizer. That is, it partitions the real line into cells that are each intervals. The index assignment l employs the index produced by the ordinary quantizer α, and uses the resulting encoder α_0 to produce the pair of indices (i_1,i_2). After transmission of the indices over different channels with mutually independent breakdown probabilities p_1 and p_2, the three decoders produce estimates from the received indices: $(\tilde{i}_1,\tilde{i}_2)$ for Decoder 0, $\tilde{i}_1$ for Decoder 1, and $\tilde{i}_2$ for Decoder 2, respectively. The index assignment must be invertible in order for the central decoder to recover the output of α. The visualization technique is to write l^{-1}, forming the index assignment matrix. Therefore, the β_1 and β_2 decoder mappings are indicated by the row and column positions in the MDSQ index assignment in Fig. 4. The action of β_0 is implicit. By performing the inverse quantization process, the output of the central decoder, which is the reconstructed sample $\widehat{\boldsymbol{X}}^{(0)}$, has a low central distortion. The side decoders output the reconstructions $\widehat{\boldsymbol{X}}^{(1)}$ and $\widehat{\boldsymbol{X}}^{(2)}$ with somewhat higher side distortions.

The breakdown probabilities for the two channels, p_1 and p_2, should be considered as the "packet-loss" probabilities. The meaning and usage of p_1 and p_2 are consistent with other MDC research papers [48,49].

Fig. 4 provides a simple example of MDSQ to help visualize the encoding operation in the *index assignment* portion of Fig. 3. This example has a codebook size of $L = 8$. In the design of an MD scalar quantizer, one can optimize α_0, β_0, β_1, and β_2 very easily as in Fig. 3. The optimization of the index assignment l is very difficult. Instead of addressing the exact optimal index assignment problem, the author in [46] presented several heuristic techniques. For example, the output for the nested index assignment is close to the best possible performance. The dimension of the matrix is denoted by P. This equals the number of descriptions, or the number of channels available for transmission in Fig. 2. We will set $P = 2$ in this paper. Thus, for $P = 2$, the descriptions of the MDSQ can be interpreted as the row and column indices of a matrix, where the codewords, or respectively, their indices, are placed. The basic ideas are to number the index assignment matrix from the upper-left corner to the lower-right corner and to fill in from the main diagonal outward. A set of index pairs are constructed from those that lie on the main diagonal and on the $2m$ diagonals closest to the main diagonal. The parameter m is called *spread*. The index assignment, shown in Fig. 4, is called the "nested index assignment," where the row and column indices, i_1, and i_2, are transmitted over two independent channels.

The cells of the encoder α are used in increasing values of i, and are numbered from j_0 to j_7 in Fig. 4(a), and j_0 to j_{21} in Fig. 4(b), respectively. Fig. 4(a) is an index assignment scheme with a spread $m = 0$. Only eight samples, or $j_0,j_1,\ldots,j_7$, are the valid scalar samples for transmission. Generally speaking, an n-bit sample can be represented by $\log_2(n)$-bit strings. The eight samples can be represented by 3-bit strings. Thus, if j_3 is the scalar sample to be transmitted, then after the index assignment step l, we obtain $i_1 = 011$ and $i_2 = 011$ represented in their binary forms. The central distortion is the quantization error between the input and the quantized samples. This configuration with a spread of $m = 0$ can be regarded as repetition of samples. It indicates that a total of 6 bits will be received if no channel breakdown occurs. Consequently, 6 bits need to be transmitted over two different channels to describe 3 bits of information. This produces a redundancy of $(\frac{6-3}{3})\times$

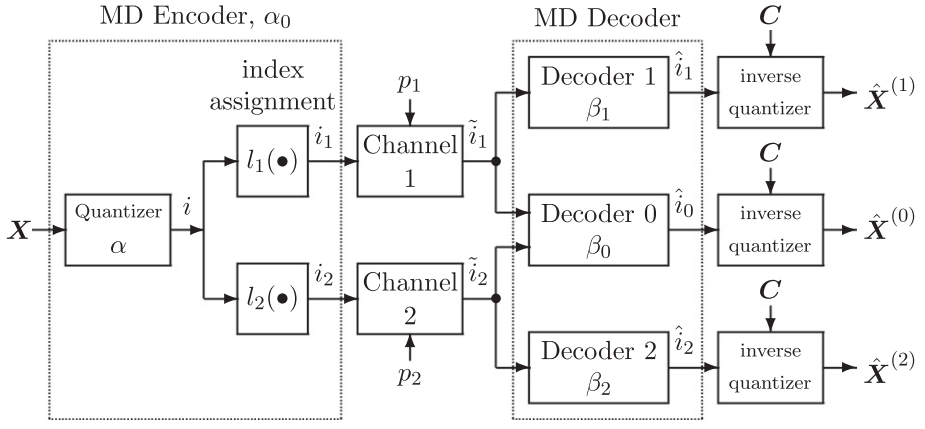

Fig. 3. The structure for MDSQ for two descriptions over two independent channels with mutually independent breakdown probabilities.

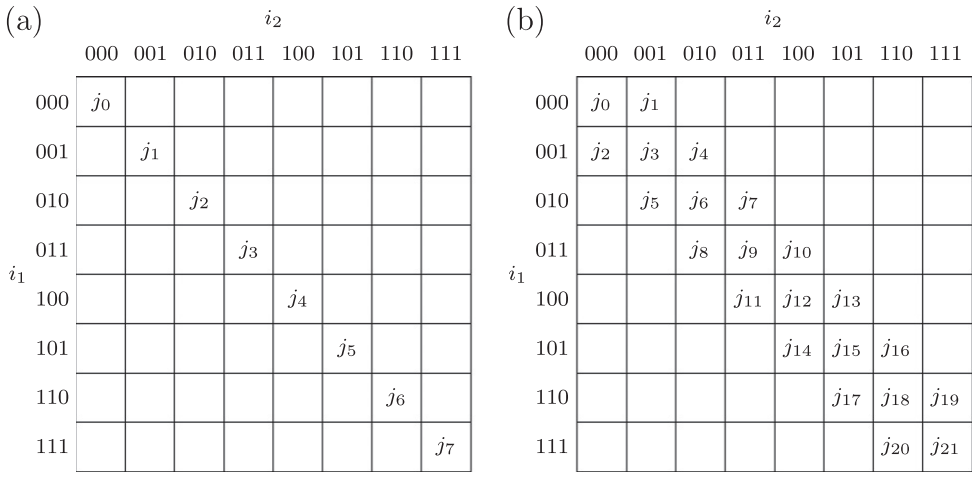

Fig. 4. The illustrations of the nested index assignment in MDSQ for two channels with codebook size $L = 8$. (a) With spread $m = 0$. (b) With spread $m = 1$.

$100\% = 100\%$. In decoding the received descriptions, if both are obtained, as depicted in [46] by calculating the conditional expectation, the reconstructed image decoded from both descriptions can be obtained. The conditional probabilities for receiving both descriptions in Fig. 4(a) are:

$$p(j_t|i_1 = 011, i_2 = 011) = \begin{cases} 1, & \text{if } t = 3, \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

Because the conditional probability for transmitting j_3 is 1.0, given the received conditions, we determine that the transmitted index is j_3. If one of the channels breaks down, say, Channel 1, $i_2 = 011$ is received. Using Fig. 4(a) and calculating the conditional probability at the decoder as indicated in Eq. (2), we visualize the column containing '011'. We can then determine that the transmitted scalar is j_3 with a probability of 1.0. This is equivalent to when both descriptions are received. In this circumstance, the central distortion is the same as the side distortion at a cost of 100% redundancy when the spread is $m = 0$ in MDSQ.

Fig. 4(b) is an index assignment scheme with a spread of $m = 1$. There are only 22 samples, or $j_0, j_1, \ldots, j_{21}$, which are the valid scalar samples for transmission. This shows that the quality of side reconstructions is represented by the small ranges of values in any row or any column depending on the received description from any one channel. An index assignment matrix with a higher fraction of occupied cells leads to a quantizer pair with lower redundancy. From the viewpoint of practical implementation, the 22 samples can be represented by $\lceil \log_2(22) \rceil$-bit, or 5-bit strings, where $\lceil \bullet \rceil$ indicates a ceiling function. If j_7 is the scalar sample to be transmitted, after the index assignment step l, we obtain $i_1 = 010$ and $i_2 = 011$. By doing this and when both descriptions are received, the transmitted sample j_7 is determined with a probability of 1.0 with Fig. 4(b). That is,

$$p(j_t|i_1 = 010, i_2 = 011) = \begin{cases} 1, & \text{if } t = 7, \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

and the redundancy is reduced to $\left(\frac{6 - \lceil \log_2(22) \rceil}{\lceil \log_2(22) \rceil}\right) \times 100\% = 20.00\%$. If one of the channels breaks down, say, Channel 1, only $i_2 = 011$ is received. With the aid of Fig. 4(b), we visualize the column containing '011' and estimate that there are three possible candidates. These are j_7, j_9, and j_{11}, which may have been transmitted, with the conditional probabilities:

$$p(j_t|i_2 = 011) = \begin{cases} \frac{1}{2m+1}, & \text{if } t = 7 \text{ or } 9 \text{ or } 11, \\ 0, & \text{otherwise,} \end{cases} \tag{4}$$

where m denotes the spread. The side distortion would be larger than the central distortion, because the side distortion is the error between the transmitted j_7 and the conditional expectation in Eq. (4), $\frac{1}{3}(j_7 + j_9 + j_{11})$. Comparing this to the case when $m = 0$, the redundancy is greatly reduced, while the side distortion is somewhat increased.

It is a straightforward task to extend MDSQ to MDVQ, though the index assignment of MDVQ is more difficult than MDSQ. Fig. 5 demonstrates the MDVQ structure with two descriptions. Here, by following Fig. 1, the input $\boldsymbol{X}_k$ denotes the small block or code vector. For example, $\widehat{\boldsymbol{X}}_k$ denotes the 4×4 block described in Section 2, for VQ operation. The reconstruction $\widehat{\boldsymbol{X}}_k^{(0)}$ from the central decoder has less distortion than that from each of the side decoders, $\widehat{\boldsymbol{X}}_k^{(1)}$ or $\widehat{\boldsymbol{X}}_k^{(2)}$. In this paper, we follow the MDSQ in [46] and MDVQ algorithm in [10], and devise a robust multi-watermarking algorithm suitable for both error resilient transmission and ownership protection.

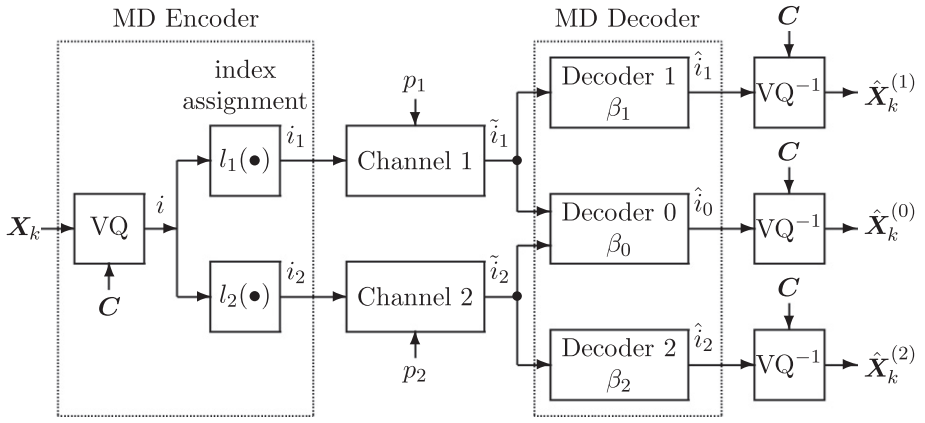

Fig. 5. The structure for MDVQ for two descriptions over two independent channels with mutually independent breakdown probabilities.

5. The watermarking algorithm for embedding two watermarks

We propose our watermarking algorithm for embedding two watermarks with VQ and MDC. Embedding more than one watermark has been an interesting topic in the literature. The structure is demonstrated in Fig. 6. Our goals are twofold. The first is to contribute to the error-resilience for the transmission of a watermarked image. Our second goal is to provide ownership protection capability with watermarking.

Let the input image be $\boldsymbol{X}$ with a size $M \times N$. We perform the VQ operation first to train the codebook for $\boldsymbol{X}$. The codebook has a size L. The codebook $\boldsymbol{C} = \{c_0, c_1, \ldots, c_{L-1}\}$ is obtained. Each index therein is represented by a $\lceil \log_2 L \rceil$-bit binary string, where $\lceil \bullet \rceil$ means a ceiling function. In operating the VQ procedure, $\boldsymbol{X}$ is divided into non-overlapping blocks $\boldsymbol{X}_k$ with size $\frac{M}{M_W} \times \frac{N}{N_W}, 0 \leqslant k \leqslant M_W \cdot N_W - 1$, then each $\boldsymbol{X}_k$ finds its nearest codeword c_i in the codebook $\boldsymbol{C}$, and the index i is assigned to $\boldsymbol{X}_k$. The steps above follow the conventional VQ procedures.

Let the watermarks for embedding be $\boldsymbol{W}_1 = \{W_{1,0}, W_{1,1}, \ldots W_{1,M_W \cdot N_W - 1}\}$ and $\boldsymbol{W}_2 = \{W_{2,0}, W_{2,1}, \ldots W_{2,M_W \cdot N_W - 1}\}$, both having sizes $M_W \times N_W$. Each element in $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$ represents one watermark bit to be embedded into $\boldsymbol{X}_k$. Embedding of the two watermarks will now be described in Section 5.1 and Section 5.2.

5.1. Embedding the first watermark

When embedding the first watermark $\boldsymbol{W}_1$, we split $\boldsymbol{C}$ into two sub-codebooks $\boldsymbol{C}' = \left\{c'_0, c'_1, \ldots c'_{\frac{L}{2}-1}\right\}$ and $\boldsymbol{C}'' = \left\{c''_0, c''_1, \ldots c''_{\frac{L}{2}-1}\right\}$. We denote this in the "codeword selection" portion in Fig. 6. We see that $\boldsymbol{C}' \bigcup \boldsymbol{C}'' = \boldsymbol{C}$ and $\boldsymbol{C}' \bigcap \boldsymbol{C}'' = \emptyset$.

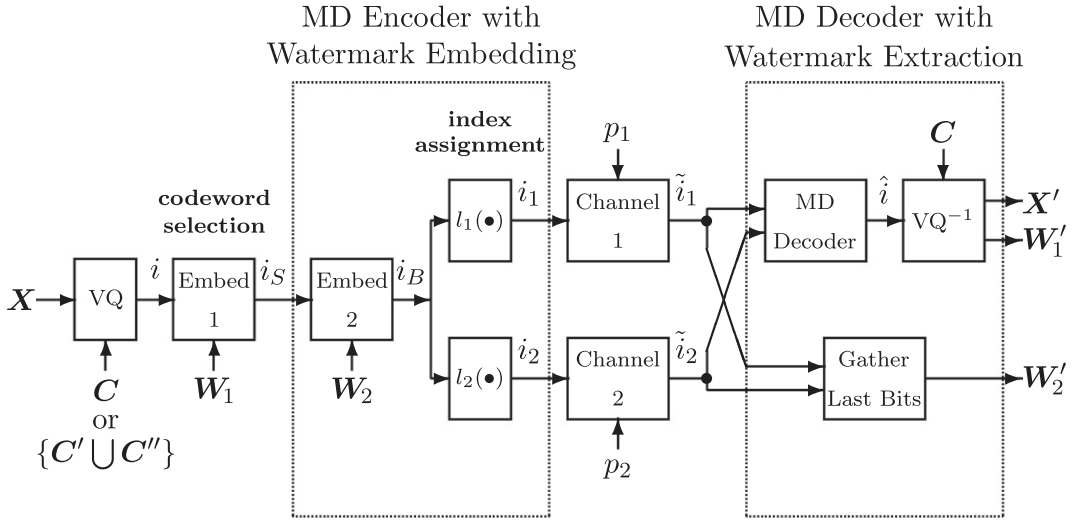

Fig. 6. The structure for embedding two watermarks with two descriptions for transmission in MDC. The two independent channels have mutually independent breakdown probabilities.

We employ tabu search to split $\boldsymbol{C}$ into $\boldsymbol{C}'$ and $\boldsymbol{C}''$ and will search for the tradeoff between watermark imperceptibility and watermark robustness using a properly selected fitness function, as described in Section 8. For one index in $\boldsymbol{C}'$, there is a one-to-one corresponding counterpart, with the same subscript, in $\boldsymbol{C}''$. For example, let c'_t denote the index for the current block $\boldsymbol{X}_k, t \in [0, \frac{L}{2} - 1]$. When embedding the first watermark, the output index i_S, which denotes the index containing ***Single*** watermark for representing $\boldsymbol{X}_k$, is generated according to the value of the watermark bit $W_{1,k}$:

$$i_S = \begin{cases} c'_t, & \text{if } W_{1,k} = 0, \\ c''_t, & \text{if } W_{1,k} = 1, \end{cases} \quad t \in \left[0, \frac{L}{2} - 1\right], \quad k \in [0, M_W \cdot N_W - 1]. \tag{5}$$

Next, i_S is fed into the MD encoder in Fig. 6 to embed the second watermark.

5.2. Embedding the second watermark

Embedding the second watermark is a modification from our previous work in [32]. This is a two-step process. First, we shift the watermarked index i_S to the left by one bit. Second, we tag the watermark bit $W_{2,k}$ and move it to the end of the shifted index. That is,

$$i_B = (i_S << 1) + W_{2,k}, \quad k \in [0, M_W \cdot N_W - 1], \tag{6}$$

where i_B denotes the index containing ***Both*** watermarks. Next, we make use of the MDSQ algorithms in Fig. 4(b) for index assignment. The index assignments in Fig. 6 are $i_1 = l_1(i_B)$ and $i_2 = l_2(i_B)$. They map the quantizer output index i_B to the two descriptions i_1 and i_2.

After completing the embedding of $W_{1,k}$ and $W_{2,k}$ in $\boldsymbol{X}_k$, i_1 and i_2 are transmitted over two memoryless and mutually independent channels with erasure probabilities p_1 for Channel 1, and p_2 for Channel 2, respectively. This procedure is completed when all of the blocks $\boldsymbol{X}_k$, $k \in [0, M_W \cdot N_W - 1]$, in the original image $\boldsymbol{X}$ are processed and the resulting indices are transmitted.

6. The watermarking algorithm for extracting two watermarks

In this section, we describe the corresponding schemes to extract the two embedded watermarks. This is an inverse procedure to the embedding process. We first extract the second embedded watermark and then extract the first embedded one.

6.1. Extracting the second watermark and obtaining the watermarked reconstruction containing $\boldsymbol{W}_1$

At the decoder side in Fig. 6, the first step is to determine the outcome i'_B from the received indices i'_1 and i'_2 with the MDSQ decoder by doing the inverse of the index assignment process, l^{-1}. Then, i'_B is shifted to the right by one bit to smooth away the effects of watermark embedding,

$$i'_S = (i'_B >> 1). \tag{7}$$

Next, the decoder performs a table look-up process on i'_S to obtain the codeword $\tilde{c}_i, 0 \leqslant i \leqslant L - 1$. We then find the block $\boldsymbol{X}'_k$ that contains the watermark bit $W_{1,k}$. By gathering all of the blocks $\boldsymbol{X}'_k$, $0 \leqslant k \leqslant M_W \cdot N_W - 1$, we obtain the watermarked reconstruction $\boldsymbol{X}'$, which contains the watermark $\boldsymbol{W}_1$.

When extracting $\boldsymbol{W}_2$, we perform the estimation using the received indices to determine the value of the watermark bits by using

$$W'_{2,k} = i'_B \bmod 2, \quad k \in [0, M_W \cdot N_W - 1], \tag{8}$$

where "mod" denotes the modulus operation. The concept for extracting $W'_{2,k}$ is straightforward. At the receiver, with the descriptions received, the index i'_B could be estimated with MDC. Next, by performing the inverse operation to the embedding procedure Fig. 6, the last bit in the index i'_B is retrieved, which is the operation described in Eq. (8). Because the embedding of the second watermark is carried out by tagging the watermark bit into the index in Eq. (6), we can perform the reverse operation to obtain the extracted watermark bit $W'_{2,k}$.

Fig. 7 is a demonstration of how $W'_{2,k}$ is extracted. By calculating the conditional probabilities with two descriptions in MDC, one of the following conditions will be satisfied.

(1) If both descriptions for one block $\boldsymbol{X}_k$ are received, the resulting index decoded using MDSQ can then be determined uniquely as shown in Fig. 7. By visualizing the intersection between the row of the received $\tilde{i}_1$, and the column of the received $\tilde{i}_2$, the estimated watermark bit $W'_{2,k}$ is extracted by removing the last bit from i'_B using Eq. (8).
(2) Because of the error concealment capability for index assignment in MDSQ, when only one description is received, the block can be partially reconstructed. The watermark bit must be determined from several possible indices assigned in the MDSQ row or column matrix [46].
   (a) If Channel 1 breaks down, then $\tilde{i}_2$ will be received. As illustrated in Fig. 7, we choose the column containing $\tilde{i}_2$ and infer that the transmitted description should be one of the several possible indices in the column. We use a majority vote to estimate the watermark bit $W'_{2,k}$ by checking whether the subscripts of the possible indices are odd or

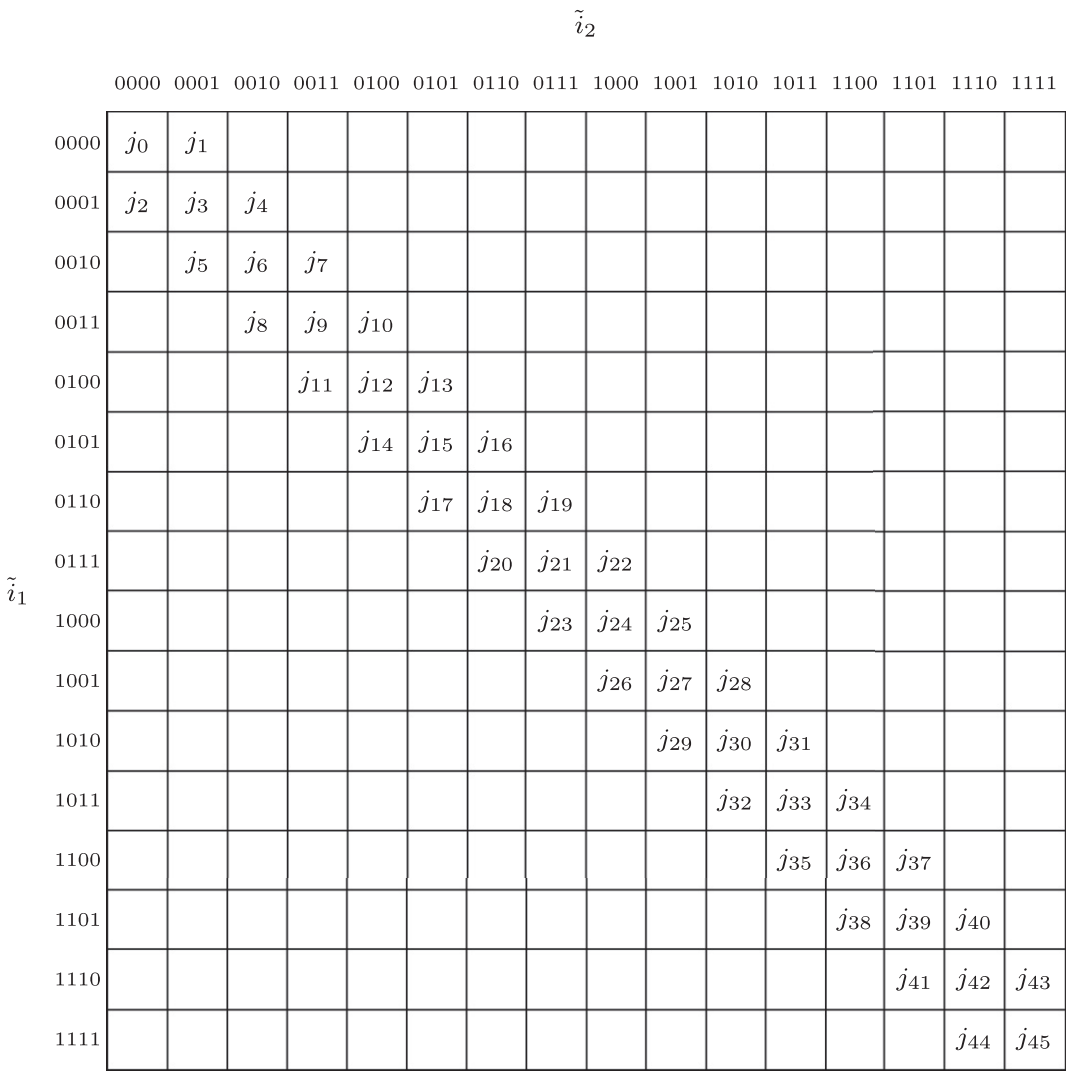

Fig. 7. An example of the combination of MDC and watermarking, an extension to Fig. 4(b) with spread $m = 1$.

even. If there are more odd indices, we set $W'_{2,k} = 1$. Otherwise, we set $W'_{2,k} = 0$ with the modulus operation in Eq. (8). After checking, if there are equal numbers of 0's and 1's in the MDSQ matrix Fig. 7, we randomly assign the watermark bit. Finally, we calculate the conditional expectation from the possible indices, and produce the reconstructed block $\boldsymbol{X}'_k$.

(b) If Channel 2 breaks down, then $\tilde{i}_1$ will be received. By visualizing Fig. 7, we choose the row containing $\tilde{i}_2$ and infer that the transmitted description should be one of the several possible indices on the row. With the same procedures in the previous case, we can obtain the extracted watermark bit $W'_{2,k}$ and the reconstructed block $\boldsymbol{X}'_k$.

(3) If no description is received, the value of the watermark bit $W'_{2,k}$ is randomly assigned. With no received information, the block $\boldsymbol{X}'_k$ cannot be reconstructed, and the luminance of every pixel in that block is set to 128, which is the average value of the 8-bit per pixel grey level images.

By gathering all of the extracted watermark bits $W'_{2,k}$, we obtain the extracted watermark $\boldsymbol{W}'_2$. By gathering all of the reconstructed blocks $\boldsymbol{X}'_k$, we obtain the reconstructed image $\boldsymbol{X}'$, which contains the first watermark. We proceed with watermark extraction as shown in Section 6.2 to extract the first watermark embedded from the received descriptions.

6.2. Extracting the first watermark

After obtaining the codeword i'_S in Section 6.1, we are prepared to extract $\boldsymbol{W}_1$. Assuming that $i'_S = \tilde{c}_i$, we examine whether the codeword $\tilde{c}_i$ belongs to the sub-codebook $\boldsymbol{C}'$ or $\boldsymbol{C}''$, and then extract the watermark bit $W'_{1,k}$. This can be estimated with Eq. (9), which is an inverse operation of Eq. (5):

$$W'_{1,k} = \begin{cases} 0, & \text{if } \tilde{c}_i \in \boldsymbol{C}', \\ 1, & \text{if } \tilde{c}_i \in \boldsymbol{C}'', \end{cases} \quad i \in [0, L-1], \quad k \in [0, M_W \cdot N_W - 1]. \tag{9}$$

By gathering all of the extracted watermark bits $W'_{1,k}$, we obtain an estimate of the first embedded watermark $\boldsymbol{W}'_1$.

7. An example for watermarking in Section 5 and Section 6

Fig. 7 is the example that follows Fig. 4(b). Here the codebook size is $L = 8$, and each codeword is 3-bit in length. Consequently, when using the watermarking scheme in Eq. (6), Fig. 7 is a direct extension of Fig. 4(b). This is because each watermarked codeword is 4 bits in length. By following conventional VQ techniques, the codebook with a size of $2^4 = 2L = 16$ is trained in advance to deal with watermark embedding. From another perspective, for watermarking purposes, the effective size of the codebook for reconstructing the compressed image is halved. The watermarked image quality with MDC should be somewhat degraded.

In this example, the original image $\boldsymbol{X}$ employed has a size of 512×512, and the two watermarks, $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$ both have sizes 128×128. With VQ encoding, the original image is divided into 4×4 blocks $\boldsymbol{X}_k$, $k \in \left[0, \frac{512\times512}{4\times4} - 1\right]$, and each block is represented by one codeword c_i, where $i \in [0, L-1]$. For watermarking, two bits can be embedded into one block, one from $\boldsymbol{W}_1$ and the other from $\boldsymbol{W}_2$.

Without loss of generality, we assume that $W_{1,k} = 0$ and $W_{2,k} = 1$ are the two watermark bits to be embedded into $\boldsymbol{X}_k$. With the algorithm given in Section 5.1 and Eq. (5), and after searching for the nearest codeword in $\boldsymbol{C}'$, we conclude that the resulting codeword containing the first watermark is $i_S = 101$. According to Eq. (6), we embed the second watermark and obtain $i_B = 1011$. Because the binary form of 1011 has a decimal form of 19, we conclude that j_{19} is the codeword to be transmitted in Fig. 6. Referring to Fig. 7, the two descriptions for transmission are $i_1 = 0110$ and $i_2 = 0111$.

To extract the watermark using both descriptions, or $P = 2$ in Fig. 2, there are four possible cases.

(1) If both descriptions are received, then $i_1 = 0110$ and $i_2 = 0111$ can be used to extract the watermark bit $W_{2,k}$ and reconstruct the image. By performing the inverse operation of index assignment l^{-1} in Eq. (1) and computing the conditional probability, we can determine exactly that the codeword j_{19} is transmitted with a probability of 1.0, and thus, the watermark bit is $W_{2,k} = 1$. In reconstructing the block from the received descriptions, we use the look-up table, and determine $\widehat{\boldsymbol{X}}_k^{(0)} = j_{19}$. We employ the spatial domain representations of j_{19} to represent the reconstructed block $\boldsymbol{X}'_k$.

(2) If Channel 1 breaks down, then only $i_2 = 0111$ is received. By performing the inverse operation of the index assignment l^{-1}, we find that j_{19}, j_{21}, and j_{23} are the three possible candidates transmitted. Because 19, 21, and 23 are odd numbers, and by using a majority vote, the embedded bit can be estimated as $W_{2,k} = 1$. In reconstructing the received image, we need to calculate the conditional probability using the given conditions:

$$p(j_t | i_2 = 0111) = \begin{cases} \frac{1}{2m+1}, & \text{if } t = 19 \text{ or } 21 \text{ or } 23, \\ 0, & \text{otherwise}. \end{cases} \tag{10}$$

where m denotes the spread. Here, we set $m = 1$. Thus, with Eq. (10), the reconstructed vector is the conditional expectation, represented by $\widehat{\boldsymbol{X}}_k^{(2)} = \frac{1}{3}(j_{19} + j_{21} + j_{23})$. We use the spatial domain representations of j_{19}, j_{21}, and j_{23} to calculate their average to complete the reconstruction of $\widehat{\boldsymbol{X}}^{(2)}$.

(3) If Channel 2 breaks down, then only $i_1 = 0110$ is received. By computing the inverse operation of index assignment, we find that j_{17}, j_{18}, and j_{19} are the three possible candidates transmitted. Using a majority vote, the embedded bit can be estimated to be $W_{2,k} = 1$. In some situations where there are two candidates, we use random selection to determine the watermark bit. In reconstructing the received image, we need to calculate the conditional probability at the given conditions:

$$p(j_t | i_1 = 0110) = \begin{cases} \frac{1}{2m+1}, & \text{if } t = 17 \text{ or } 18 \text{ or } 19, \\ 0, & \text{otherwise}. \end{cases} \tag{11}$$

Again, m denotes the spread, and we set $m = 1$ here. Thus, with Eq. (11), the reconstructed vector is the conditional expectation represented by $\widehat{\boldsymbol{X}}_k^{(1)} = \frac{1}{3}(j_{17} + j_{18} + j_{19})$.

(4) If both channels break down, no description is received. We randomly choose '0' or '1' to represent the extracted watermark bit $W_{2,k}$. In addition, because no description is received, nothing can be reconstructed, and therefore, we set the luminance of every pixel in that block to 128 for the 8-bit per pixel grey level images.

8. Optimization with tabu search

8.1. Tabu search fundamentals

Tabu search is a meta-heuristic approach, characterized by the use of a flexible memory. It is able to escape local minima and to search areas beyond a local minimum [9,14].

Tabu search overcomes the local optima problem by using an evaluation function, or a fitness function, that chooses the highest evaluation solution at each iteration. The building blocks of tabu search are stated as follows:

Forbidding strategy: This strategy is employed to avoid cycling problems by classifying certain moves as forbidden, or *tabu*. To prevent the cycling problem, it is sufficient to check whether a solution has been previously visited. Alternatively, this can be approximated by whether the solution has been visited during the last T_S iterations. T_S is normally named the *tabu list length* or *tabu list size*. With the help of an appropriate value of T_S, the likelihood of cycling effectively vanishes.
Aspiration criteria and tabu restrictions: An aspiration criterion is applied to make a tabu solution that is not a forbidden state. Each solution must be of sufficient quality and able to prevent cycling. A solution is acceptable if the tabu restrictions are satisfied. However, a tabu solution is also assumed acceptable if an aspiration criterion applies, regardless of the tabu status. We also make use of tabu restrictions to avoid repetitions but not reversals. A tabu restriction is typically activated only when its attributes occur within a limited number of iterations prior to the present iteration, or if they have occurred with a certain frequency over a larger number of iterations. The appropriate use of aspiration criteria can significantly improve the performance of a tabu search.
Freeing strategy: The freeing strategy is used to decide which solutions can be removed from the tabu list. This strategy removes tabu restrictions of the solutions so that they can be reconsidered during future steps of the search. The attributes of a tabu solution remain on the tabu list for a duration of T_S iterations.
Intermediate and long-term learning strategies: These strategies are implemented with intermediate and long-term memory functions. Their operations are to record good features of a selected number of moves generated during the execution of the algorithm.
Short-term strategy or overall strategy: This strategy manages the interplay between the different strategies listed above. A candidate list is a sub-list of the possible moves which are generally problem dependent.
The best-solution strategy: This strategy selects an admissible solution from the current solutions if it yields the greatest improvement or the least distortion in the cost function, provided that the tabu restrictions and aspiration criteria are satisfied.
Termination: A stopping criterion terminates the tabu search procedure either after a specified number of iterations, or if the currently best solution has not improved for a given number of iterations.

Using the background information presented in this section, we are able to apply the tabu search algorithm to digital watermarking to obtain an optimized outcome between watermark imperceptibility and watermark robustness.

8.2. Watermarking with tabu search

In Section 5.1, the main problem when embedding the first watermark is how to split the codebook $\boldsymbol{C}$ into two sub-codebooks $\boldsymbol{C}'$ and $\boldsymbol{C}''$. The result for splitting $\boldsymbol{C}$ will not only influence the watermark imperceptibility and the robustness of the first watermark, but it will also affect the robustness of the second watermark. All of the problems can be optimized using tabu search [9] by offering the properly chosen fitness function. Using the fundamentals of tabu search described in Section 8.1, and the watermarking requirements described in Section 1, we can consider both the imperceptibility of the watermarked image, represented by Peak Signal-to-Noise Ratio (PSNR), and the robustness of the extracted watermarks, represented by Bit Correct Rates (BCR), for optimization. The fitness function with this system is:

$$f_i = \mathrm{PSNR}_i + \lambda_1 \cdot \mathrm{BCR}_{1,i} + \lambda_2 \cdot \mathrm{BCR}_{2,i}, \tag{12}$$

where, f_i denotes the fitness score in the ith iteration, PSNR_i,$\mathrm{BCR}_{1,i}$ and $\mathrm{BCR}_{2,i}$ denote Peak Signal-to-Noise Ratio (PSNR) of the watermarked image, and Bit Correct Rates (BCR) of the two extracted watermarks, respectively. Objectively, higher PSNR values lead to better imperceptibility, and higher BCR values lead to better robustness. Thus, the goal for optimization is to maximize the fitness score f_i.

Because the PSNR values are generally many times larger than the BCR values, we include λ_1 and λ_2 to represent the weighting factors to balance the effects of PSNR and BCR. The objective is to maximize f_i in our watermarking system. The PSNR in the ith iteration between the original and watermarked images can be represented by

$$\mathrm{PSNR}_i = 10 \cdot \log_{10}\left(\frac{255^2}{\frac{1}{M\cdot N}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}\left(X(m,n) - X_i'(m,n)\right)^2}\right), \tag{13}$$

where $X(m,n)$ and $X_i'(m,n)$ denote the pixel values at position (m,n) of the original image $\boldsymbol{X}$ and the watermarked image $\boldsymbol{X}_i'$ in the ith iteration, where $M \cdot N$ denotes the image size. The BCR between the embedded and extracted watermarks can be defined by

$$\mathrm{BCR}_i = \left(\frac{1}{M_W \cdot N_W}\sum_{b=0}^{M_W\cdot N_W - 1}\overline{\left(w_{b,i} \oplus w_{b,i}'\right)}\right) \cdot 100\%, \tag{14}$$

where $w_{b,i}$ and $w_{b,i}'$ represent the embedded watermark bit and the extracted one in the ith iteration, $M_W \cdot N_W$ denotes the watermark size, $\oplus$ indicates the "exclusive-or operation," and the line above the exclusive-or operation means the "not" operation in logic design.

Given the preliminaries in Section 1, we fix the watermark capacity so that both the watermark imperceptibility and watermark robustness improve after tabu search optimization. The parameters employed in the tabu search are:

- there are 20 candidate solutions trained for each iteration;
- the tabu list length T_S is set to 10;
- the weighting factors are set to $\lambda_1 = \lambda_2 = 10$;
- the aspiration value is set to 40, with $PSNR_i \geqslant 26$, $BCR_{1,i} \geqslant 0.7$, and $BCR_{2,i} \geqslant 0.7$ in Eq. (12);
- watermark embedding and extraction are performed in every training iteration to obtain the updated $PSNR_{i+1}$, $BCR_{1,i+1}$, and $BCR_{2,i+1}$ for the next iteration;
- the number of total training iterations is set to 100.

The above parameters are chosen carefully. In this paper, we choose 20 candidates for training with tabu search. After considering the computation time, the memory consumption, and the convergence rate in tabu search, we choose 20 candidates for each training iteration based on the fitness function. If we choose too many candidates, the computation time per iteration will increase and memory allocation might become a problem. In contrast, if we choose too few candidates, the output result might only be a locally optimal one, which is a problem generally encountered in optimization. We have found 20 candidates to be a reasonable number for training with tabu search.

Moreover, we set the tabu list length $T_S = 10$ by considering the tradeoff between the computation time and the convergence rate in the optimization process. We also discovered that PSNR values are many times larger than the BCR values. To compensate for this, we use the weighting factors, λ_1 and λ_2, which need to be included in Eq. (12), the fitness function, to weight the effects from the PSNR, representing the image quality, and the BCR, representing the watermark robustness.

9. Other related watermarking algorithms

To demonstrate the effectiveness of the proposed algorithm, we compare two VQ-based watermarking algorithms reported in the literature. After searching major databases, we found that VQ-based watermarking constitutes about 2% of the total paper counts among all watermarking-related papers since the mid-1990s. The two papers here were carefully chosen for comparison with our proposed algorithm. The two watermarking algorithms are described as follows.

9.1. Review of watermarking algorithm in [27]

The algorithm in [27] is one of the pioneering VQ-based watermarking methods proposed in literature. The authors trained the codebook $\boldsymbol{C}$ with a size of L in advance. Then, $\boldsymbol{C}$ is partitioned into N groups, $\{\boldsymbol{G}_0, \boldsymbol{G}_1, \ldots, \boldsymbol{G}_{N-1}\}$, where

(1) $\boldsymbol{C} = \bigcup_{i=0}^{N-1} \boldsymbol{G}_i$;
(2) $\bigcap_{i=0}^{N-1} \boldsymbol{G}_i = \emptyset$;
(3) $\boldsymbol{G}_i = \{\boldsymbol{c}_0^i, \boldsymbol{c}_1^i\}, i \in [0, N-1]$.

For a given input vector $\boldsymbol{X}_k$, we assume that the codeword $\boldsymbol{c}_t^p \in \boldsymbol{G}_p, p \in [0, N-1], t \in \{0, 1\}$, is the nearest codeword. To embed the corresponding watermark bit $w \in \{0, 1\}$ in $\boldsymbol{X}_k$, the jth codeword of $\boldsymbol{G}_p$ is the output as the watermarked vector $\boldsymbol{X}'_k$:

$$j = (t + w) \bmod \|\boldsymbol{G}_p\|, \tag{15}$$

$$\boldsymbol{X}'_k = \boldsymbol{c}_j^p, \tag{16}$$

where $j \in [0, \|\boldsymbol{G}_p\|]$, and $\|\boldsymbol{G}_p\|$ denotes the number of codewords contained in group $\boldsymbol{G}_p$, and "mod" means the modulus operation. For embedding only one bit into each vector, $\|\boldsymbol{G}_p\| = 2$.

After all of the watermark bits have been embedded into the corresponding vectors, the output vectors are pieced together to form the watermarked image, $\boldsymbol{X}'$. In addition, due to the embedding strategy employed, this method requires the original cover image to be presented during extraction, or else the hidden information cannot be obtained. This is a fatal disadvantage for the practical application of this algorithm.

9.2. Review of watermarking algorithm in [36]

The algorithm described in this sub-section is an improvement on some existing schemes for VQ-based watermarking reported in the literature [18]. The trained codebook has a size of L, which is an even number. Then, $\boldsymbol{C} = \{c_0, c_1, \ldots, c_{L-1}\}$ is employed for vector quantization. In [36], the authors propose a method to partition the codebook according to the watermarking bits '0' or '1' to be embedded. They divide $\boldsymbol{C}$ into the odd-indexed and even-indexed sub-codebooks $\boldsymbol{C}_o$ and $\boldsymbol{C}_e$, with $\boldsymbol{C}_o = \{c_1, c_3, c_5, \ldots, c_{L-1}\}$ and $\boldsymbol{C}_e = \{c_0, c_2, c_4, \ldots, c_{L-2}\}$. Thus, $\boldsymbol{C}_o \cup \boldsymbol{C}_e = \boldsymbol{C}$ and $\boldsymbol{C}_o \cap \boldsymbol{C}_e = \emptyset$.

To embed the watermark, if the codeword for the current block is c_k, $k \in [0, L-1]$, and if the watermark bit is '0', $\boldsymbol{C}_e$ is adopted for watermark embedding because 0 is an even number, and the nearest codeword in $\boldsymbol{C}_e$ is found to replace the

original codeword. If the watermark bit is '1', $\boldsymbol{C}_o$ is used, and the same scheme is applied for embedding watermark bit '1'. Finally, with the codewords in $\boldsymbol{C}_o$ and $\boldsymbol{C}_e$, the watermarked image $\boldsymbol{X}'$ is reconstructed.

To extract the watermark, the codebook plays an essential role. Suppose that the watermarked image $\boldsymbol{X}'$ is transmitted over the packet loss channel, and the received image is denoted by $\boldsymbol{X}''$. On the receiver side, the same sub-codebooks $\boldsymbol{C}_o$ and $\boldsymbol{C}_e$ are employed to extract the watermark. The authors use table look-up to find the VQ indices of the received image $\boldsymbol{X}''$. For every block in the vector quantization, if the index belongs to $\boldsymbol{C}_o$, then the extracted watermark bit is determined to be '0'; if not, the bit is '1'. By gathering all of the extracted watermark bits, the extracted watermark $\boldsymbol{W}'$ can be reconstructed.

9.3. Comparisons between our algorithm and those in Sections 9.1 and 9.2

As we stated in Section 1, imperceptibility, robustness, and capacity are the three important requirements in designing a watermarking algorithm. We quantitatively compare our algorithm to those presented above, based on the three requirements. Our results indicate that our algorithm outperforms the two algorithms reported in the literature [27,36].

First, we compare the watermark capacity. The main difference between our algorithm and those in [27,36] is that our algorithm can embed two binary watermarks, each with a watermark capacity of $128 \times 128 = 16384$ bits, or $128 \times 128 \times 2 = 32768$ bits in total. Only one binary watermark (having size $128 \times 128 = 16384$ bits) can be embedded by using either [27] or [36]. Consequently, we can embed twice the number of bits, or twice the capacity.

Next, we compare the robustness and imperceptibility criteria between our algorithm and those in [27,36]. These can be compared objectively with the results presented in Section 10. For the robustness metric, the Hamming distance (HD) is employed in [27], and the normalized cross-correlation (NC) is used in [36], respectively. For the imperceptibility metric, both papers employ the Peak Signal-to-Noise Ratio (PSNR) for making comparisons. Better robustness is indicated by either smaller HD or larger NC values. Better imperceptibility is indicated by larger PSNR values. Because the robustness metrics differ among our paper and those in [27,36], we determine to use the commonly employed Bit Correct Rates (BCR) for making comparisons in Section 10. Readers are encouraged to refer to Section 10 for detailed comparisons. From the results presented in Table 1 through Table 4 in Section 10, it is evident that our algorithm has better performance.

To summarize, by comparing quantitative measures of the three requirements for a watermarking algorithm, we conclude that our algorithm yields better results than those in [27,36]. We attribute this performance to the innovations in our algorithm design compared to existing published algorithms.

10. Simulation results

In our simulations, we use the test image, `Lena`, with size 512×512, as the original source. We have embedded a watermark with size 128×128, shown in Fig. 8. The original source is divided into 4×4 blocks for VQ compression, which also meets the number of bits required for watermark embedding. The codebook sizes are $L = 512$ and $L = 1024$, and indices therein are represented by 9-bit or 10-bit strings, respectively.

Two quantities are considered to evaluate our proposed algorithm. We employ the watermarked image quality as the first metric for evaluation. The watermarked image quality, measured by the Peak Signal-to-Noise Ratio (PSNR) between the watermarked image $\boldsymbol{X}'$ and the original $\boldsymbol{X}$ can be calculated by using Eq. (13). $\boldsymbol{X}'$ is reconstructed from the received descriptions which are transmitted over two mutually independent, erasure channels. The second metric, the Bit Correct Rates (BCR), of the two extracted watermarks are employed to evaluate the robustness of the algorithm. They can be calculated with Eq. (14). Generally speaking, we aim for a high PSNR value, indicating that the watermarked image quality is preserved, and high BCR values in the extracted watermarks, indicating watermark robustness. We also make comparisons with other existing VQ-based watermarking schemes [27,36] described in Section 9. Simulations show the practicality and usefulness of our method.

As depicted in Fig. 6, only the watermarked VQ codewords are transmitted over the noisy channels. Therefore, attacking schemes such as low-pass filtering, or those employed in the Stirmark benchmark [37], are not applicable to our scheme. Therefore, we only use the situations where the descriptions can be transmitted over mutually independent channels.

Table 1
Comparisons of watermarked image quality under different channel erasure probabilities for the codebook size of $L = 512$ after considering watermark embedding.

Channel erasure prob.		PSNR with our method (in dB)	PSNR with [27] (in dB)	PSNR with [36] (in dB)
p_1	p_2			
0	0	30.74	30.46	30.64
0.1	0.1	28.15	28.14	28.56
0.25	0.25	24.39	24.37	24.40
0.5	0.5	19.88	19.93	19.86
0	1	26.19	25.92	25.41
1	0	26.13	26.04	25.48

Table 2
Comparisons of watermark robustness under different channel erasure probabilities for the codebook size of $L = 512$ after considering watermark embedding.

Channel erasure probability		BCR with our method (in %)		BCR with [27] (in %)	BCR with [36] (in %)
p_1	p_2	W_1'	W_2'	W_1'	W_1'
0	0	100	100	100	100
0.1	0.1	94.12	93.60	90.48	94.90
0.25	0.25	87.50	85.64	78.47	87.19
0.5	0.5	74.30	72.08	62.88	74.69
0	1	81.03	68.23	75.72	79.72
1	0	67.00	69.71	66.38	66.52

Table 3
Comparisons of watermarked image quality under different channel erasure probabilities for the codebook size of $L = 1024$ after considering watermark embedding.

Channel erasure prob.		PSNR with our method (in dB)	PSNR with [27] (in dB)	PSNR with [36] (in dB)
p_1	p_2			
0	0	32.74	31.84	32.20
0.1	0.1	28.35	28.47	28.88
0.25	0.25	24.27	24.33	24.34
0.5	0.5	19.78	19.77	19.79
0	1	25.10	24.88	25.17
1	0	25.04	24.72	25.17

Table 4
Comparisons of watermark robustness under different channel erasure probabilities for the codebook size of $L = 1024$ after considering watermark embedding.

Channel erasure probability		BCR with our method (in %)		BCR with [27] (in %)	BCR with [36]
p_1	p_2	W_1'	W_2'	W_1'	W_1'
0	0	100	100	100	100
0.1	0.1	94.27	93.44	90.77	94.76
0.25	0.25	86.52	84.78	79.06	85.77
0.5	0.5	73.28	71.34	62.66	72.42
0	1	76.61	70.78	75.89	80.94
1	0	68.56	70.26	66.66	68.13

Simulations with different channel erasure probabilities are presented in Fig. 9 and Fig. 10 with a codebook size of $L = 512$, and in Fig. 11 and Fig. 12 with a codebook size of $L = 1024$, respectively. We also use tables to make comparisons with the results obtained from other existing algorithms. In Table 1 and Table 2, we present the watermarked PSNR values and the BCR values of the extracted watermarks with a codebook size of $L = 512$. Table 3 and Table 4 show their counterparts with the codebook size of $L = 1024$.

In Fig. 9, p_1 and p_2 denote the erasure probabilities with Channel 1 and Channel 2. Fig. 9(a) shows the extracted watermarks under error-free transmission. These are identical to the watermark embedded in Fig. 8. In Fig. 9(b)–(d), they represent the results after transmission over lightly- to heavily-erased channels. The BCR values are still high and the extracted watermarks are recognizable even when $p_1 = p_2 = 0.5$. In Fig. 10, we demonstrate the case when one of the channels experiences a total breakdown. When Channel 2 breaks down, the first watermark is recognizable in Fig. 10(a), while the second cannot be distinguished. When Channel 1 breaks down, as depicted in Fig. 10(b), we obtain similar results.

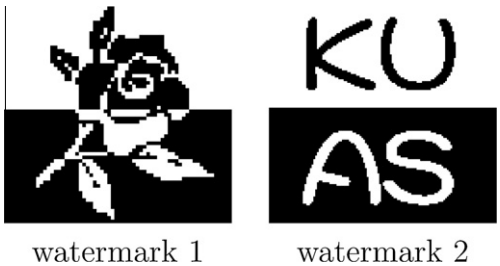

Fig. 8. The two watermarks used for embedding in our simulation. Both have sizes 128×128. Watermark 1 shows a flower, while watermark 2 denotes the characters KUAS, representing the author's affiliation.

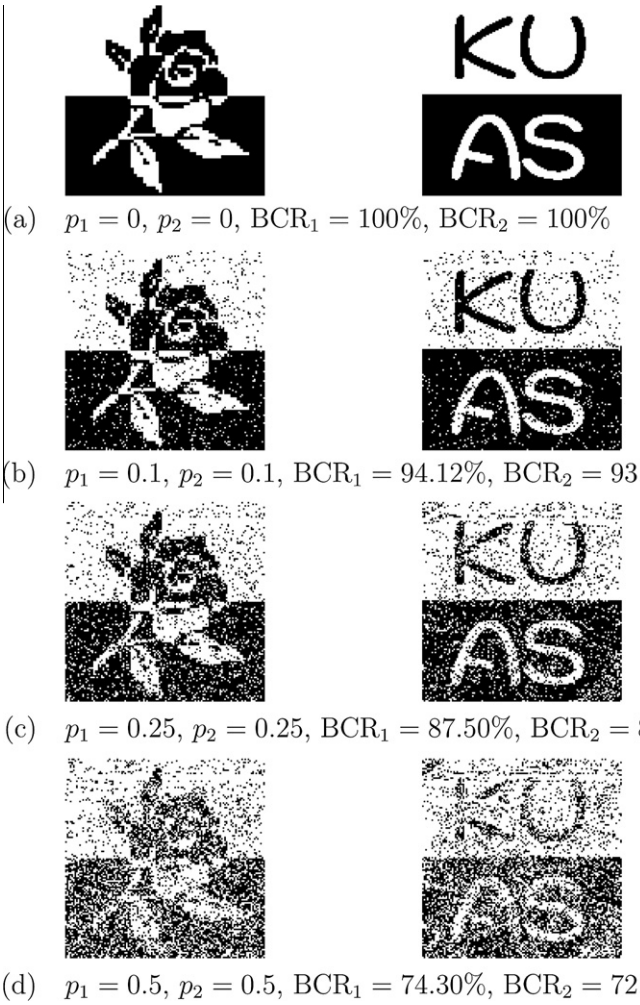

Fig. 9. The two extracted watermarks under different channel erasure probabilities, with a codebook size of $L = 512$.

In Table 1 and Table 2, the PSNR and BCR values under different erasure probabilities are indicated. Comparisons with the results in [27] and [36] are also made. The PSNR values in Table 1 show error-resilient capabilities with MDVQ under severely erased channels. The PSNR values with our algorithm outperform others in most cases. Under the error-free condition, the watermarked PSNR using our algorithm has the best performance. In addition, the BCR values shown in Table 2 are acceptable even with heavily erased channels, and the corresponding watermarks can all be subjectively recognized. In only one situation when one of the channels breaks down completely, is it not possible to recognize the second watermark. More importantly, using our algorithm, we can embed twice as much watermark capacity compared to the algorithms reported in [27] and [36]. Thus, while only one watermark can be extracted using the prior algorithms, the corresponding BCR values using our algorithm are better than those in [27] and [36].

In Fig. 11, similar comparisons can be made by following those given in Fig. 9. Fig. 11(a) shows the extracted watermarks during error-free transmission. These are identical to those embedded in Fig. 8. In Fig. 11(b)–(d), they represent the results when transmitting over the lightly to heavily erased channels. The BCR values are still high and the extracted watermarks are recognizable even with $p_1 = p_2 = 0.5$. When comparing these results with Fig. 9(b)–(d), although the BCR values are a little inferior, all the extracted watermarks are recognizable. In Fig. 12, when one of the channels fails, both of the first watermarks are recognizable, while the second watermarks can only be partially distinguished.

In Table 3 and Table 4, PSNR and BCR values under different erasure probabilities are indicated. Comparisons with results from [27] and [36] are also made. The PSNR values in Table 3 show the error-resilient capabilities using MDVQ under severely erased channels. The PSNR values using our algorithm outperform others in most cases, and under the error-free condition, the watermarked PSNR using our algorithm performs best. In comparison with Table 1, with a larger codebook size in Table 3, we obtain better PSNR values. In addition, the BCR values in Table 2 are acceptable even with heavily erased channels. Again, with our algorithm, we can embed twice as much watermark capacity compared to those in [27] and [36]. Only

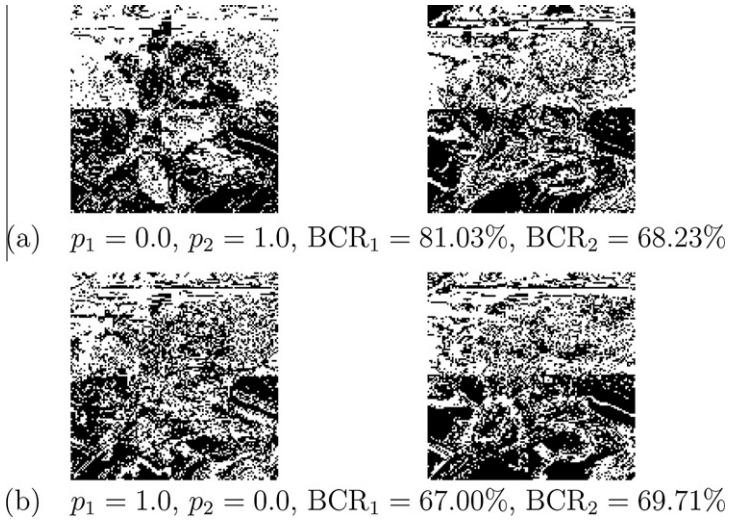

Fig. 10. The two extracted watermarks for the case when one channel suffers a total breakdown, with a codebook size of $L = 512$.

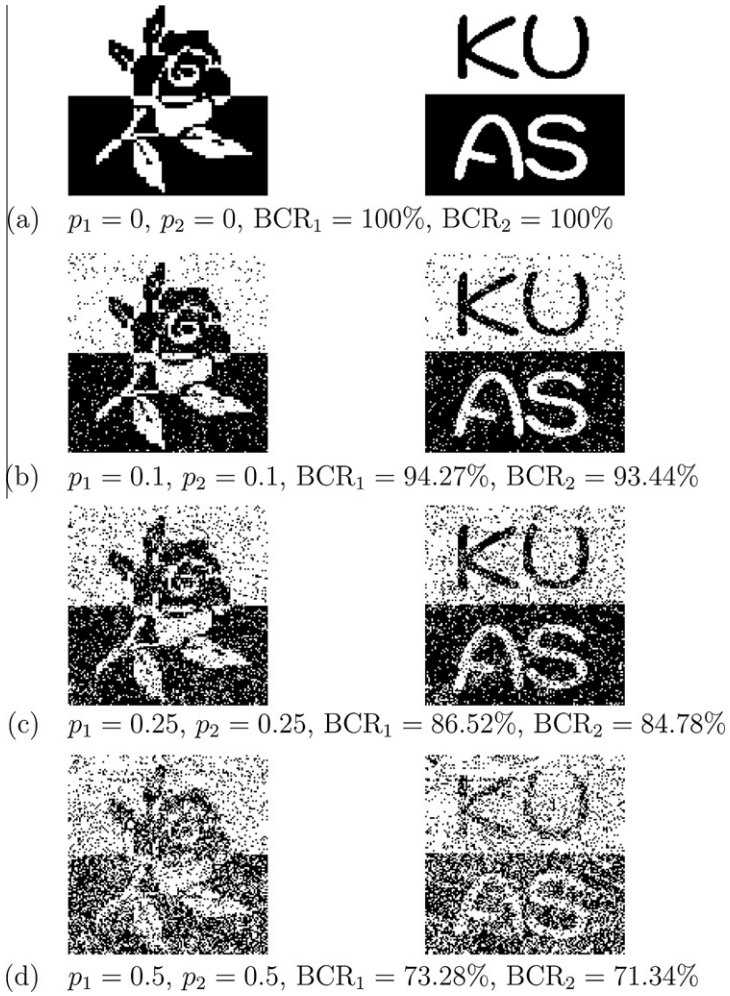

Fig. 11. The two extracted watermarks under different channel erasure probabilities, for a codebook size of $L = 1024$.

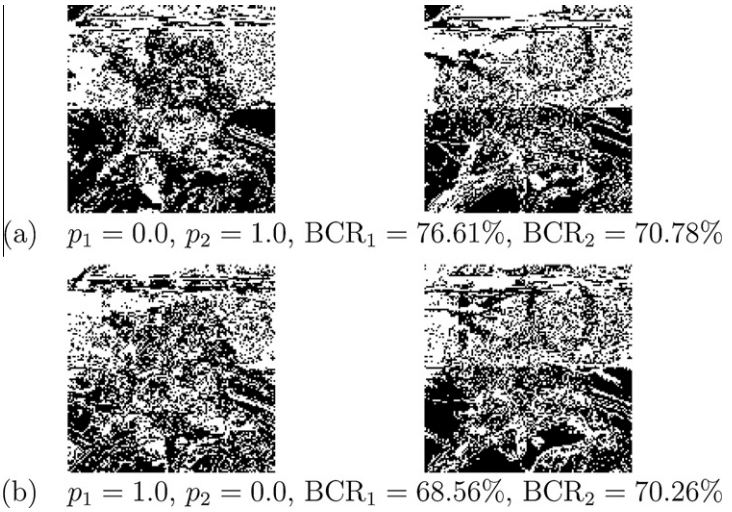

Fig. 12. The two extracted watermarks for the case when one channel suffers a total breakdown, for a codebook size of $L = 1024$.

one watermark can be extracted with existing algorithms, and the corresponding BCR values with our algorithm are better than [27] and [36].

In summary, under a wide range of channel erasure probabilities, the results using our proposed algorithm demonstrate both the effective transmission of watermarked images, and the acceptable robustness of the extracted watermarks. Compared with the existing schemes, we have doubled the amount of watermark capacity embedded with our algorithm; our algorithm also performs better on imperceptibility and robustness measures.

11. Conclusions

In this paper, we proposed an innovative scheme for VQ-based image multi-watermarking with multiple description coding (MDC), which is suitable for transmission over noisy channels. We modified the MDVQ and MDSQ index assignments for watermark embedding and extraction. By incorporating this with MDC, we obtained promising results. We also presented discussions and made comparisons between our algorithm and others previously published, and we point out the superiority of our algorithm. Simulation results indicate that our watermarking algorithm is more robust and more resilient with respect to combat with channel noise under both lightly and heavily erased channels. In addition, in comparison with existing VQ-based algorithms in the literature, our algorithm performed better than others in both the watermark imperceptibility, shown by PSNR, and the watermark robustness, shown by BCR. Therefore, our algorithm is not only innovative for research, but also suitable for practical implementation.

Acknowledgement

This work was supported by National Science Council (Taiwan, ROC) under Grant No. NSC 93-2219-E-009-006 and NSC 95-2221-E-390-034. The authors would also like to give their gratitude to the Editor-in-Chief and anonymous reviewers for their valuable comments.

References

[1] M. Ak, K. Kaya, K. Onarlıoglu, A.A. Selçuk, Efficient broadcast encryption with user profiles, Inform. Sci. 180 (6) (2010) 1060–1072.
[2] P. Artameeyanant, Tabu searching for watermarking robust against compression and cropping, in: SICE-ICASE International Joint Conference, 2006, pp. 4461–4463.
[3] M. Barni, F. Bartolini, A. De Rosa, A. Piva, Capacity of full frame DCT image watermarks, IEEE Trans. Image Process. 9 (8) (2000) 1450–1455.
[4] C.C. Chang, P.Y. Pai, C.M. Yeh, Y.K. Chan, A high payload frequency-based reversible image hiding method, Inform. Sci. 180 (11) (2010) 2286–2298.
[5] F.C. Chang, H.C. Huang, A refactoring method for cache-efficient swarm intelligence algorithms, Information Sciences, (doi:10.1016/j.ins.2010.02.025).
[6] T.D. Chen, N.N. Chen, Optimization of multi-purpose watermarking algorithm based on tabu search, Comput. Eng. Appl. 42 (36) (2006) (in Chinese).
[7] A.A. El Gamal, T.M. Cover, Achievable rates for multiple descriptions, IEEE Trans. Inform. Theory 28 (6) (1982) 851–857.
[8] A. Gersho, R.M. Gray, Vector Quantization and Signal Compression, Kluwer Academic Publishers, Boston, MA, 1992.
[9] F. Glover, M. Laguna, Tabu Search, Kluwer Academic Publishers, Boston, MA, 1997.
[10] N. Görtz, P. Leelapornchai, Optimization of the index assignments for multiple description vector quantizers, IEEE Trans. Commun. 51 (3) (2003) 336–340.
[11] H.C. Huang, Y.H. Chen, Genetic fingerprinting for copyright protection of multicast media, Soft Comput. 13 (4) (2009) 383–391.
[12] H.C. Huang, Y.H. Chen, S.C. Chen, Copyright protection for image files with EXIF metadata, Int. J. Innovative Comput. Inform. Control 5 (7) (2009) 1903–1909.
[13] H.C. Huang, C.M. Chu, J.S. Pan, The optimized copyright protection system with genetic watermarking, Soft Comput. 13 (4) (2009) 333–343.

[14] H.C. Huang, S.C. Chu, J.S. Pan, Z.M. Lu, A tabu search based maximum descent algorithm for VQ codebook design, J. Inform. Sci. Eng. 17 (5) (2001) 753–762.
[15] H.C. Huang, W.C. Fang, Metadata-based image watermarking for copyright protection, Simul. Modell. Pract. Theory 18 (4) (2010) 436–445.
[16] A. Ito, S. Makino, Designing side information of multiple description coding, J. Inform. Hiding Multimedia Signal Process. 1 (1) (2010) 10–19.
[17] N.S. Jayant, J.D. Johnston, R.J. Safranek, Signal compression based on models of human perception, Proc. IEEE 81 (10) (1993) 1385–1422.
[18] M. Jo, H.D. Kim, A digital image watermarking scheme based on vector quantisation, IEICE Trans. Inform. Syst. E85-D (6) (2002) 1054–1056.
[19] X. Kang, J. Huang, Y.Q. Shi, Y. Lin, A DWT–DFT composite watermarking scheme robust to both affine transform and JPEG compression, IEEE Trans. Circuits Syst. Video Technol. 13 (8) (2003) 776–786.
[20] S. Katzenbeisser, F. Petitcolas, Information Hiding – Techniques for Steganography and Digital Watermarking, Artech House, Norwood, MA, 2000.
[21] R.H. Koenen, J. Lacy, M. Mackay, S. Mitchell, The long march to interoperable digital rights management, Proc. IEEE 92 (6) (2004) 883–897.
[22] C. Lin, J.S. Pan, Robust VQ-based digital image watermarking for noisy channel, in: First International Conference on Innovative Computing, Information and Control, 2006, pp. 673–676.
[23] C.Y. Lin, S.F. Chang, Watermarking capacity of digital images based on domain-specific masking effects, in: International Conference on Information Technology: Coding and Computing, 2001, pp. 90–94.
[24] Q. Liu, A.H. Sung, M. Qiao, Z. Chen, B. Ribeiro, An improved approach to steganalysis of JPEG images, Inform. Sci. 180 (9) (2010) 1643–1655.
[25] Z. Liu, Y. Hu, X. Zhang, H. Ma, Certificateless signcryption scheme in the standard model, Inform. Sci. 180 (3) (2010) 452–464.
[26] P. Loo, N. Kingsbury, Watermark detection based on the properties of error control codes, IEE Proc. Vision, Image Signal Process. 150 (2) (2003) 115–121.
[27] Z.M. Lu, S.H. Sun, Digital image watermarking technique based on vector quantisation, IEE Elec. Lett. 36 (4) (2000) 303–305.
[28] B. Macq, J. Dittmann, E.J. Delp, Benchmarking of image watermarking algorithms for digital rights management, Proc. IEEE 92 (6) (2004) 971–984.
[29] S. Moskowitz, What is acceptable quality in the application of digital watermarking: trade-offs of security, robustness and quality, in: International Conference on Information Technology: Coding and Computing, 2002, pp. 80–84.
[30] P. Moulin, M.K. Mıhçak, A framework for evaluating the data-hiding capacity of image sources, IEEE Trans. Image Process. 11 (9) (2002) 1029–1042.
[31] J.S. Pan, C.S. Cheng, B.Y. Liao, C.H. Hsu, K.C. Huang, Optimization of multipurpose watermarking algorithm, in: 2004 IEEE Asia-Pacific Conference on Circuits and Systems, 2004, pp. 601–604.
[32] J.S. Pan, Y.C. Hsin, H.C. Huang, K.C. Huang, Robust image watermarking based on multiple description vector quantisation, IEE Elect. Lett. 40 (22) (2004) 1409–1410.
[33] J.S. Pan, C.Y. Huang, H.C. Huang, B.Y. Liao, K.C. Huang, Multiple description multi-watermarking with tabu search approaches, Visual Commun. Image Process. Proc. SPIE (5960) (2005) 1455–1467.
[34] J.S. Pan, H.C. Huang, L.C. Jain (Eds.), Intelligent Watermarking Techniques, World Scientific Publishing Company, Singapore, 2004.
[35] J.S. Pan, M.T. Sung, H.C. Huang, B.Y. Liao, Robust VQ-based digital watermarking for the memoryless binary symmetric channel, IEICE Trans. Fundam. E-87A (7) (2004) 1839–1841.
[36] J.S. Pan, F.H. Wang, H.C. Huang, L.C. Jain, Improved schemes for VQ-based image watermarking, IEEE Int. Symp. Consum. Electron. (2003). paper no: ISCE03011.
[37] F.A.P. Petitcolas, Stirmark benchmark 4.0, 2004. <http://www.petitcolas.net/fabien/watermarking/stirmark/>.
[38] A. Piva, F. Bartolini, M. Barni, Managing copyright in open networks, IEEE Internet Comput. 6 (3) (2002) 18–26.
[39] C.I. Podilchuk, W.J. Zeng, Image-adaptive watermarking using visual models, IEEE J. Sel. Areas Commun. 16 (4) (1998) 525–539.
[40] M. Ramkumar, A.N. Akansu, On the design of data hiding methods robust to lossy compression, IEEE Trans. Multimedia 6 (6) (2004) 947–951.
[41] C.S. Shieh, H.C. Huang, F.H. Wang, J.S. Pan, An embedding algorithm for multiple watermarks, J. Inform. Sci. Eng. 19 (2) (2003) 381–395.
[42] H.J. Shiu, K.L. Ng, J.F. Fang, R.C.T. Lee, C.H. Huang, Data hiding methods based upon DNA sequences, Inform. Sci. 180 (11) (2010) 2196–2208.
[43] R. Sion, M. Atallah, S. Prabhakar, Rights protection for relational data, IEEE Trans. Knowl. Data Eng. 16 (12) (2004) 1509–1525.
[44] N. Sriyingyong, K. Attakitmongcol, Wavelet-based audio watermarking using adaptive tabu search, in: 2006 1st International Symposium on Wireless Pervasive Computing, 2006, pp. 1–5.
[45] R. Toscano, P. Lyonnet, A new heuristic approach for non-convex optimization problems, Inform. Sci. 180 (10) (2010) 1955–1966.
[46] V.A. Vaishampayan, Design of multiple description scalar quantizers, IEEE Trans. Inform. Theory 39 (3) (1993) 821–834.
[47] A.O. Waller, G. Jones, T. Whitley, J. Edwards, D. Kaleshi, A. Munro, B. MacFarlane, A. Wood, Securing the delivery of digital content over the Internet, Electron. Commun. Eng. J. 14 (5) (2002) 239–248.
[48] Y. Wang, M.T. Orchard, V.A. Vaishampayan, A.R. Reibman, Multiple description coding using pairwise correlating transforms, IEEE Trans. Image Process. 10 (3) (2001) 351–366.
[49] Y. Wang, A.R. Reibman, M.T. Orchard, H. Jafarkhani, An improvement to multiple description transform coding, IEEE Trans. Signal Process. 50 (11) (2002) 2843–2854.
[50] K. Yamamoto, M. Iwakiri, Real-time audio watermarking based on characteristics of PCM in digital instrument, J. Inform. Hiding Multimedia Signal Process. 1 (2) (2010) 59–71.
[51] Z. Zhang, T. Berger, New results in binary multiple descriptions, IEEE Trans. Inform. Theory 33 (4) (1987) 502–521.