Engineering Applications of Artificial Intelligence 97 (2021) 104049

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

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

Digital watermarking with improved SMS applied for QR code

Jeng-Shyang Pan [a,*], Xiao-Xue Sun [a], Shu-Chuan Chu [a,b], Ajith Abraham [c], Bin Yan [d]

[a] College of Computer Science and Engineering, Shandong University of Science and Technology, Qingdao 266590, China
[b] College of Science and Engineering, Flinders University, 1284 South Road, Clovelly Park SA 5042, Australia
[c] Machine Intelligence Research Labs (MIR Labs), Auburn, WA, USA
[d] College of Electronic and Information Engineering, Shandong University of Science and Technology, Qingdao 266590, China

A R T I C L E I N F O

Keywords:
Quick Response code
Watermarking
Discrete Wavelet Transform
Singular Value Decomposition
States of Matter Search

A B S T R A C T

With the rapid development of information technology, infringements have become increasingly serious. Digital watermarking is an effective method to protect information. The current watermarking technology still has room for further improvement in imperceptibility and robustness. This paper proposes an improved watermarking technology using meta-heuristic algorithm. Further, Quick Response code (QR code) is used as a carrier to transmit information. The improved Discrete Wavelet Transform-Singular Value Decomposition (DWT-SVD) is used to hide the watermark into the QR code. Therefore, digital watermarking is realized on the QR code. In the common watermark embedding methods, the digital watermark is related to the embedding strength. How to find a suitable embedding factor and reduce distortion is of great significance to these watermarking algorithms. This paper mainly proposes two novel algorithms based on States of Matter Search (SMS) algorithm to find suitable embedding factors. The first algorithm uses an adaptive parameter to control the movement of particles called the adaptive step States of Matter Search (sSMS). The second algorithm incorporates co-evolutionary matrix to enhance the search capability named Co-evolution States of Matter Search (CSMS). DWT-SVD is updated through two algorithms to acquire optimal embedding strength factors on the QR code watermarking. By adjusting the embedding strength factors, the intensity of the watermark embedded in different frequency domains would be modified. The experimental results have higher PSNR and the QR code can still be decoded by a general decoder. It shows that the proposed approaches are practicable and effective.

1. Introduction

Digital watermarking technology directly embeds some secret information into multimedia content through a certain algorithm. It will not affect the value and use of the original content, and will not be noticed by human intuition. With the development of theoretical research, technology has been developed sharply in practical application, such as copyright protection and information hiding. Digital watermarking technology should be safe, that is, it cannot be tampered with. It should be not easily detected, and will not affect the normal use of the original carrier. It should be robust, which means that after suffering one or more attacks, the watermark can still be extracted completely. The sensitivity means that digital watermarking can determine whether the data has been tampered. Further, the location and the degree of damage can be judged.

The most common watermarking techniques include the Least Significant Bit (LSB), Discrete Cosine Transform (DCT), DWT, SVD, and Discrete Fourier Transform (DFT). Tirkel et al. propose two methods to add a watermark to the LSB of a gray-scale image (Van Schyndel et al., 1994). Subsequently, O'Ruanaidh et al. propose DCT, which is a watermarking scheme based on spread spectrum communication in 1996 (O'Ruanaidh et al., 1996). Lu et al. propose a robust image-based on DCT in 2010 (Lu et al., 2010). DWT and DFT are the most commonly used watermarking techniques (Ganic and Eskicioglu, 2004). Kundur et al. come up with embedding the watermark into the wavelet domain (Kundur and Hatzinakos, 1998). The wavelet transform deals with the image signal and exploits a series of wavelets of different scales to decompose the original function. After the transformation, the coefficients of the original function under different scales of wavelet are obtained. The wavelet transforms algorithm has the advantage of keeping the original anti-filtering and compression attacks.

At present, the Discrete Wavelet Transform (DWT) has been widely exploited in the digital image, video, audio, and other fields. So far, there are various improved DWT algorithms that have effectively improved the robustness and image quality of embedded watermark. But they are still not insufficient to resist multiple attacks. SVD is

* Corresponding author.
E-mail addresses: jspan@cc.kuas.edu.tw (J.-S. Pan), Xues1123@163.com (X.-X. Sun), jan.chu83@gmail.com (S.-C. Chu), ajith.abraham@ieee.org (A. Abraham), yanbinhit@hotmail.com (B. Yan).

https://doi.org/10.1016/j.engappai.2020.104049
Received 27 May 2020; Received in revised form 24 September 2020; Accepted 25 October 2020
Available online 24 November 2020

a special matrix transformation. It is also widely used in watermark processing (Liu and Tan, 2002; Mohammad et al., 2008). Mohammad et al. present a new semi-blind reference watermarking scheme in 2009 (Bhatnagar and Raman, 2009). A new combination method DWT-SVD appears, which combined DWT with SVD.

Computational intelligence algorithms are mostly stemmed from biological behaviors in nature, and most of them have strong optimization ability, which is a powerful tool to solve problems in actual scenarios (Corchado et al., 2010; Pan et al., 2020; Kim et al., 2007). Computational intelligence includes three basic areas: fuzzy computing, neural networks, and evolutionary computing. In the last few years, a good deal of naturally inspired computation methods is proposed. They have their expansibilities and have been widely used in engineering and daily life. Classical examples are genetic algorithm (GA) by simulating the genetic evolution in the world (Abraham et al., 2006), particle swarm optimization (PSO) by simulating bird swarm movement (Kennedy and Eberhart, 1995). There are many intelligent optimization algorithms proposed in nature, such as multiverse optimizer (MVO) (Mirjalili et al., 2016), pigeon-inspired algorithm (PIO) (Duan and Qiao, 2014; Tian et al., 2020), differential evolution algorithm (DE) (Hu et al., 2014; Das et al., 2008; Meng et al., 2019), cat swarm algorithm (CSO) (Chu et al., 2006; Tsai et al., 2008), flower pollination algorithm (Nabil, 2016; Nguyen et al., 2019), QUasi-Affine TRansformation Evolution (QUATRE) (Pan et al., 2016; Du et al., 2020; Sun et al., 2020), gray wolf algorithm (GWO) (Emary et al., 2015; Hu et al., 2019) and so on.

Generally speaking, the objective is to get a globally optimal solution under the given conditions in many algorithms. SMS is proposed by Erik Cuevas et al. which is inspired by the physical principles of thermal-energy motion mechanism in 2014 (Cuevas et al., 2014). Compared with other evolutionary algorithms, it improves balance and has better searchability. The relationship between exploration and exploitation is adjusted in our new approach. In this paper, we mainly study more scientific methods based on adaptive step SMS (sSMS) and significative Co-evolution States of Matter Search (CSMS). CSMS has a wonderful property with fewer hardware demands. It effectively improves the balance between exploration and exploitation.

The embedding factors in the watermarking technique affect the imperceptibility and robustness of the image. They are studied to find the optimal values for QR code watermarking. Ying Yang et al. determine the adaptive embedding factor by establishing a mathematical model on Human Visual System (HVS) characteristics (Yang et al., 2008). C. Patvardhan et al. hide watermark by converting from RGB color space to YCbCr space to exploit characteristics of the HVS in 2018 (Patvardhan et al., 2018). Later, some computational intelligence algorithms are widely applied on watermarking methods, including PSO (Saxena et al., 2018), GA (ElShafie et al., 2008) and quantum-inspired evolutionary algorithm (Samanta et al., 2017).

This paper mainly uses our novel algorithms to obtain a robust QR code watermarking. Before the watermark is embedded, the scrambling algorithm combining Arnold transform and Logistic transform is performed on the watermark. This method not only changes the original appearance of the watermark but also alters its information. To some extent, the security of watermarking is improved, which is conducive to ensure the secondary encryption of this algorithm. The QR code is a practical tool and has a security mechanism (Liu et al., 2019). Lastly, we combine the improved methods with the QR code. It cannot only be decoded correctly but also the watermark can be extracted faultlessly. In addition, the watermarked QR code can resist some attacks. Therefore, this watermarked QR code carries two-level information. It is difficult to find the watermark.

The purposed approach applies information encryption and hides information into a QR code. Our main contribution is to design two novel algorithms to obtain lower distortion and stronger robustness for the QR code watermarking technique. In addition, they have the ability to resist some attacks.

The rest of this paper is organized as follows. Basic concepts of original SMS, encryption methods, DWT, and SVD are introduced in Section 2. Section 3 proposes two novel algorithms based on SMS, including sSMS and CSMS. The optimization process for DWT-SVD and the adaptive embedding strength factors are illustrated in Section 4. The combination with the QR code watermarking is shown as well. Section 5 is the experimental results. Finally, the conclusion is given in Section 6.

2. Related work

In general, the watermark model is organized by two parts, one part is the watermark embedding and the other is the extraction of this watermark. The section provides a theoretical basis for the research. The basis for optimization is also derived.

2.1. States of matter search

The part mainly introduces the original SMS algorithm. It is proposed to find a global optimal result refer to the matter model. It enhances the balance of exploitation and exploration. SMS is a novel search pattern based on a common physical phenomenon. According to the states of matter, the evolutionary process is divided into three phases.

In the gas phase, the molecules are arranged very loosely. The force between the molecules is very small. They move violently in the confined space, despite they have no shape or volume. The molecules of the liquid phase, the arrangement deserves to be called unwound with less force. In this way, the liquid has also energy to move reversely, thus forming a movable model. Differently, molecules are packed in a regular arrangement, and intermolecular forces are obvious. Moreover, the solid has a definite shape and volume. Therefore, molecules just move randomly around their equilibrium positions, and the range of movement is quite small. Simulating the states of matter search, the molecules present disparate motions.

In the first phase, it is a gas state that accounts for 50% of all iterations. It is mainly used for exploration. In the second stage, it is a liquid state and accounts for 40%. There is a transition between exploitation and exploration. Finally, the remaining 10% of all iterations is the last stage, which is a solid-state and emphasizes the exploitation. In SMS, each individual is regarded as a molecule and evolved by simulating the behavior of the thermal-energy mode including some operators. The direction vector operator plays a crucial role in moving. Suppose that a molecule $\vec{X}_i$ of population $\mathbf{X}$ is n-dimensional, which corresponds to the direction vector $\vec{d}_i$. This $\vec{d}_i$ denotes respective direction towards the best position and it is updated by Eq. (1).

$$\begin{cases} \vec{a}_i = \dfrac{(\vec{X}_g - \vec{X}_i)}{\left\| \vec{X}_g - \vec{X}_i \right\|}, \\ d_{i,r}^{t+1} = d_{i,r}^{t} \times (1 - \dfrac{t}{I_{max}}) \times 0.5 + a_{i,r}, \end{cases} \tag{1}$$

where $\vec{a}_i$ is the unitary vector of movement, $\vec{X}_g$ represents the best solution up to now. The t is the current iteration number. The total iteration numbers are I_{max}.

The second operator is a collision that mimics the interactions between molecules. Within a certain range, the molecules collide with each other. If two molecules have a collision their direction vectors would exchange with each other. Hence, the collision radius is needed as:

$$r = \frac{\sum_{r=1}^{m} (b_r^u - b_r^l)}{m} \times \beta,\ \beta \in [0, 1]\,, \tag{2}$$

where b_r^u means the upper bound of r parameter and b_r^l is the lower bound.

Table 1
Control parameters of SMS with different states.

Stage	α	μ	β	T
The first gas stage	[0.8, 1]	0.8	0.8	0.9
The second liquid stage	[0.3, 0.6]	0.4	0.2	0.2
The last solid stage	[0, 0.1]	0.1	0	0

In the last phase, there is a random position operator that embodies the random movement in the search space by introducing a probability technique. The position is generated according to a predefined threshold T as below.

$$x_{i,r}^{t+1} = \begin{cases} b_r^l + rand \times (b_r^u - b_r^l), & \text{if } n_r < T \\ x_{i,r}^t, & \text{if } n_r < 1 - T \end{cases} \tag{3}$$

where the range of the random number n_r is from 0 to 1.

The three states take some uniform steps. Their control parameters are set to disparate at each phase. The general details are given below.

Step 1: Select the best individual from the whole population.

Step 2: Initialize the initial velocity size as Eq. (4) and collision radius is calculated by Eq. (2).

$$v_0 = \frac{\sum_{r=1}^{m}(b_r^u - b_r^l)}{m} \times \mu, \ \mu \in [0, 1]. \tag{4}$$

Step 3: Adjust new position exploiting Eq. (5) based on the direction vector.

$$\begin{cases} v_{i,r}^{t+1} = d_{i,r}^{t+1} \times v_0, \\ x_{i,r}^{t+1} = x_{i,r}^t + v_{i,r}^{t+1} \times rand \times \alpha \times (b_r^u - b_r^l). \end{cases} \tag{5}$$

Step 4: Collide within range and exchange their respective direction vectors.

Step 5: Update randomly position according the threshold T by Eq. (3).

On account of the characteristics of disparate motion modes at each stage, the control parameters should be set variously under Table 1. In the first gas stage, particles without restriction have the most violent movement and the widest collision range. Comparing with the gas state molecules perform less active so that these parameters are set smaller. Because of the strong interaction force of molecules, they can hardly move and collide. Their random position cannot be created as well.

As the evolution of different phase, the learning strategy of individuals changes with the parameters, which efficiently prevents premature convergence. The collision model is helpful to increase the diversity of solutions in a predetermined manner. SMS has good ability to find global optima.

2.2. Digital watermarking

This section introduces the basic theoretical concepts about digital watermarking. It includes the encryption operation on the watermark, DWT, and SVD. The most common watermarking technique is DWT in these common methods.

2.2.1. The watermark encryption

To further protect it, the watermark is usually encrypted before the watermark is embedded. The encryption operation mainly utilizes Arnold transform. It is a periodic image scrambling algorithm that encrypts images by changing their pixel positions. The encryption algorithm is simple and easy to implement, so it is widely used. The transformation is an reversible operation in a $m \times m$ image. It can be recovered by transforming.

Arnold transform has periodicity after the cycle of transformation could be restored to the original image. It is easy to be cracked, that is, there seems to be a certain degree of hidden danger in security. After the coordinates of the image are scrambled by Arnold transform, the pixels are changed by logistic transformation. Definition of logistic mapping is $y_{l+1} = y_l \times \rho \times (1 - y_l)$, where $\rho \in [0, 4]$ and $y_l \in (0, 1)$. $\rho \in [0, 4]$ means the logistic parameter. When $y_l \in [0, 1]$, logistic works in a chaotic state. In the end, logistic sequence is acquired. The generated image is difficult to crack. It also looks very different from the original image. Therefore, the security of the watermark is further enhanced.

2.2.2. DWT-SVD

DWT-SVD is mainly divided into two steps. Firstly, the cover image is transformed into four frequency domains utilizing DWT. Secondly, the watermark is embedded into all sub-bands of the selected low frequency after it is decomposed by SVD.

Wavelet transform is a traditional transformation analysis method. It offers a time-frequency window in which the width changes with the frequency. High compression ratio, fast compression speed, and anti-interference in transmission are the significant advantages of wavelet analysis. Wavelet decomposition is used to decompose the image on different scales. The low-frequency signal contains the characteristics of signals, and the high-frequency information shows the details of the signal. After transformation, the image is decomposed into four quarter-sized sub-graphs: the mid-high frequency detail sub-graph in the horizontal, vertical, and diagonal directions and the low-frequency approximation sub-graph. In subsequent decomposition, the approximation sub-graph is decomposed in the same way into smaller sub-graphs at the next level of resolution. Adopting the technique, we deal with the coefficients after decomposing the image. As Fig. 1 demonstrates how the original picture is segmented by the three-level DWT. LL_3 signifies the optimal approximation of the original image under the maximum scale and minimum resolution, which is determined by the decomposition series.

According to HVS, it is more sensitive in low and middle-frequency regions. DWT is used to perform three-level decomposition of the image. The watermark is embedded in it. The watermarked image is not only robust but also could not be easily removed by signal processing.

SVD is a familiar matrix decomposition method. It is used for image compression. $\mathbf{M}$ is a $a \times b$ real matrix, then there would exist a decomposition as the following formula.

$$\mathbf{M}_{a\times b} = \mathbf{U}_{a\times a} \times \mathbf{C}_{a\times b} \times \mathbf{V}_{b\times b}^T, \tag{6}$$

where $\mathbf{C}$ denotes a diagonal matrix. The number of non-zero value is $min(a, b)$ where $min(a, b)$ means the smaller between a and b. The elements lying in the diagonal is called the singular value of the matrix $\mathbf{M}$. $\mathbf{U}$ and $\mathbf{V}$ are orthogonal matrices.

SVD is used to process the image. There are some attributes for image processing using SVD. The singular value on the obtained matrix $\mathbf{C}$ has good stability. When a small disturbance is added to an image, the singular value does not change significantly. After $\mathbf{C}$ is modified to embed watermark, there will not be a big difference in HVS for people. Significantly, singular value means intrinsic algebraic properties. To enhance the performance of DWT against attack, Ganic and Eskicioglu proposed DWT-SVD. For taking into account the robustness, the watermark is embedded into second level low-frequency sub-band.

3. The designs for improved SMS

In this section, the two schemes of optimization based on SMS are described. For the goal of this design, we take minimization under the restrict conditions as the optimal value in our algorithms.

3.1. sSMS

The first improvement employs the step factor α on SMS. The adaptive α is optimized in the direction vector. From the previous process of SMS, the parameter α is a preset value. It is fixed separately in three stages. sSMS employs an adaptive step to update the new solution set to select the next position.

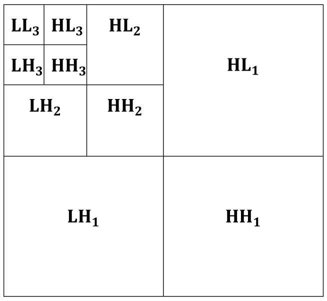


Fig. 1. Three-level DWT image decomposition.

In sSMS, the primary method is to use an adaptive step parameter α. The previous α is selected within the defined range by different states. sSMS is based on simulating a group of particles to find their positions taking advantage of the adaptive step parameter α. The change of α is shown as Eq. (7). The updated α is helpful to find more scientific solution with novel updating rule.

$$\alpha = \alpha_{max} - \frac{t}{I_{max}} \times (\alpha_{max} - \alpha_{min}), \tag{7}$$

where α is a control parameter in finding a better position of the particle. α_{max} and α_{min} are upper and lower bound of α from Table 1. Their values changes along with the gas state, liquid state, and solid state, respectively. So, the new molecules are generated by the direction vector operator as Eq. (8).

$$\begin{cases} \alpha = \alpha_{max} - \dfrac{t}{I_{max}} \times (\alpha_{max} - \alpha_{min}), \\ v_{i,r}^{t+1} = d_{i,r}^{t+1} \times v_0, \\ xt_{i,r}^{t+1} = x_{i,r}^{t} + v_{i,r}^{t+1} \times rand \times \alpha \times (b_r^u - b_r^l). \end{cases} \tag{8}$$

In this method, the collision operator is unchanged. And the random positions are still selected in the same ways as Eq. (9).

$$xt_{i,r}^{t+1} = \begin{cases} b_r^l + rand \times (b_r^u - b_r^l), & \text{if } n_r < T \\ x_{i,r}^t, & \text{if } n_r < 1 - T \end{cases} \tag{9}$$

where n_r is a random value within the range [0, 1]. The threshold T is a prearranged value as Table 1 according to the state of particles. $\vec{X}t_i = (xt_{i,1}, xt_{i,2}, xt_{i,3}, \ldots, xt_{i,r}, \ldots, xt_{i,dim}), i \in [1, ps]$.

In the meanwhile, computing the positions of particles adopts a new update strategy. That is, if a particle has a better fitness value in a new position, it will move to the new location. Otherwise, it will keep its position unchanged. Then, the new molecules are generated by the direction vector operator as follows.

$$\vec{X}_i = \begin{cases} \vec{X}t_i, & \text{if } f(\vec{X}t_i) < f(\vec{X}_i) \\ \vec{X}_i, & \text{if } f(\vec{X}t_i) \geqslant f(\vec{X}_i) \end{cases} \tag{10}$$

where $f(*)$ presents the fitness function.

The improved algorithm applies an adaptive parameter configuration and different movement strategies. Other parameters are the same as Table 1. According to different parameters in three states, there is the general process as Algorithm 1. Therefore, Algorithm 2 shows the whole process of novel sSMS.

Algorithm 1 Pseudo code of general details

```
Calculate collision radius r by Eq. (2)
while t <predefined iteration number or stop criterion do
  /Move with direction vector operator/
  for i = 1 : ps do
    for r = 1 : dim do
      d_{i,r}^{t+1} = d_{i,r}^t × (1 − t/I_max) × 0.5 + a_{i,r}
       Update by Eq. (8)
    end for
  end for
  /Generate collisions/
  for i = 1 : ps do
    for l = 2 : ps do
      if ‖X_i − X_l‖ < r and i ≠ l then
        d_t = d_i
        d_i = d_l
        d_l = d_t
      end if
    end for
  end for
end while
```

Algorithm 2 sSMS-Pc: Pseudo code of sSMS algorithm

Input: Dimension dim, population size ps, and the total number of iteration I_{max}
Output: Optimal solution

```
Initialization: X^0 = (X_1^0, X_2^0, X_3^0, ..., X_i^0, ..., X_pos^0)^T and initial velocity v_0 by Eq. (4)
//Global best selection//
f(X_g) = min(f(X_i)), X_gb = X_i
while t < 0.5 * I_max do
  // Enter the gas state//
  Set specific parameters: α_max = 1, α_min = 0.8, μ = 0.8, β = 0.8 and T = 0.9
  Apply the general details as Algorithm 1
end while
while t < 0.9 * I_max do
  //Enter the liquid state//
  Set specific parameters: α_max = 0.6, α_min = 0.3, μ = 0.4, β = 0.2 and T = 0.2
  Apply the general details as Algorithm 1
end while
while t < I_max do
  //Enter the solid state//
  Set specific parameters: α_max = 0.1, α_min = 0, μ = 0.1, β = 0 and T = 0
  Apply the general details as Algorithm 1
end while
/Random positions by Eq. (9)/
//Positions selection by Eq. (10)//
//The optimal position//
f(X_g) = min(f(X_i)), X_g = X_i
return f(X_g), X_g
```

3.2. CSMS

The second method takes advantage of a co-evolution matrix **M** so that evolution would be more holistic. The optimized algorithm is proposed with the QUATRE method. QUATRE is proposed and changed by Meng et al. (2016, 2018) and Meng and Pan (2018). In virtue of matrix **M** in QUATRE, it realizes the good combination of the balance ability in SMS and QUATRE co-evolutionary learning ability. QUATRE owns preeminent large-scale search capability, whereas it is effortless to trap into local optimum. Particles could move in the form of a matrix and change the learning strategy in SMS. Therefore, CSMS increases searching capability.

In QUATRE, $\mathbf{M}_t$ is defined as a unit lower triangular matrix and through the random transformation into matrix **M**. The operation includes two continuous steps. Firstly, the row elements in $\mathbf{M}_t$ are randomly arranged; in the next moment arrange all rows. In general, ps is bigger than dim. $\overline{\mathbf{M}}$ denotes the inverse matrix of **M** and it could be expressed as Eq. (11) that corresponds to **M**.

$$\mathbf{M}=\begin{bmatrix}1 & 1 & 1 & \\ 1 & & & \\ 1 & 1 & \cdots & 1\\ & \cdots & & \\ 1 & \cdots & 1 & 1\\ & & \vdots & \\ & \cdots & 1 & \\ 1 & & & \end{bmatrix}\xrightarrow[operation]{inverse}\begin{bmatrix} & \cdots & & 1\\ \cdots & 1 & 1 & 1\\ \cdots & & 1 & \\ & \cdots & & \\ & 1 & & \\ & & \vdots & \\ 1 & 1 & \cdots & 1\\ & 1 & 1 & 1\end{bmatrix}=\overline{\mathbf{M}} \tag{11}$$

Applying a novel learning scheme mainly changes the process which is controlled by the direction vector operator using matrix **M**. We employ the adaptive step strategy proposed by sSMS. Therefore, the direction vector of movement is updated by Eq. (12).

$$\begin{cases}\alpha=\alpha_{max}-\dfrac{t}{I_{max}}\times(\alpha_{max}-\alpha_{min}),\\ v_{i,r}^{t+1}=d_{i,r}^{t+1}\times v_0.\end{cases} \tag{12}$$

The positions of all molecules are depicted as Eq. (13) in a novel way.

$$\mathbf{X}^{t+1}=\mathbf{M}\otimes\mathbf{X}^{t}+\overline{\mathbf{M}}\otimes\mathbf{V}, \tag{13}$$

where the operator $\otimes$ represents component-wise multiplication between two matrices. Each element in a matrix multiples the corresponding element of another matrix.[1] Matrix $\mathbf{V}=(\vec{V}_1,\vec{V}_2,\dots,\vec{V}_i,\dots,\vec{V}_{ps})^T$ and $\vec{V}_i=(v_{i,1},v_{i,2},\dots,v_{i,r},\dots,v_{i,dim})$ where $i\in[1,ps]$.

The movement under the collision operator and random position is unchanged. Meantime, the final position is selected as a better solution. This new strategy makes the search procedure avoid repetition and improve the speed of convergence. Due to the form of evolution changes from a single molecule to the whole matrix, the general algorithm is also changed into Algorithm 3. Algorithm 4 gives the integrated procedure of CSMS.

Algorithm 3: Pseudo code of general algorithm.

Calculate collision radius r by Eq. (2)
while t <predefined iteration number **or** stop criterion **do**
  /Move with direction vector operator/
  for $i=1:ps$ **do**
    for $r=1:dim$ **do**
      $d_{i,r}^{t+1}=d_{i,r}^{t}\times(1-\frac{t}{I_{max}})\times 0.5+a_{i,r}$
      Update by Eq. (12)
    end for
  end for
  Eq. (13)
  /Sort molecules from large to small about the fitness function/
  for $i=1:ps/2$ **do**
    Reposition the first half of the molecule.
  end for
  /Generate collisions/
  for $i=1:ps$ **do**
    for $l=2:ps$ **do**
      if $\left\|\vec{X}_i-\vec{X}_l\right\|<r$ and $i\neq l$ **then**
        $\vec{d}_t=\vec{d}_i$
        $\vec{d}_i=\vec{d}_l$
        $\vec{d}_l=\vec{d}_t$
      end if
    end for
  end for
end while

Algorithm 4: CSMS-Pc: Pseudo code of CSMS algorithm.

Input: Dimension dim, population size ps, the total number of iteration I_{max}, matrix **M**, and $\overline{\mathbf{M}}$
Output: Optimal solution
Initialization: $\mathbf{X}^0=(\vec{X}_1^0,\vec{X}_2^0,\vec{X}_3^0,\cdots,\vec{X}_i^0,\cdots,\vec{X}_{ps}^0)^T$ and initial velocity v_0 by Eq. (4)
//**Global best selection**//
$\mathbf{X}opb=\mathbf{X}$, $f(\vec{X}_g)=min(f(\vec{X}opb_i))$, $\vec{X}_g=\vec{X}opb_i$
while $t<0.5*I_{max}$ **do**
  // **Enter the gas state**//
  Set specific parameters: $\alpha_{max}=1$, $\alpha_{min}=0.8$, $\mu=0.8$, $\beta=0.8$ and $T=0.9$
  Apply the general details as Algorithm 3
end while
while $t<0.9*I_{max}$ **do**
  //**Enter the liquid state**//
  Set specific parameters: $\alpha_{max}=0.6$, $\alpha_{min}=0.3$, $\mu=0.4$, $\beta=0.2$ and $T=0.2$
  Apply the general details as Algorithm 3
end while
while $t<I_{max}$ **do**
  //**Enter the solid state**//
  Set specific parameters: $\alpha_{max}=0.1$, $\alpha_{min}=0$, $\mu=0.1$, $\beta=0$ and $T=0$
  Apply the general details as Algorithm 3
end while
/Random positions by Eq. (9)/
//**Positions selection**//
if $f(\vec{X}_i^{t+1})<f(\vec{X}opb_i^t)$ **then**
  $f(\vec{X}opb_i^{t+1})=f(\vec{X}_i^{t+1})$
end if
$\mathbf{X}=\mathbf{X}opb$
//**Optimal position**//
$f(\vec{X}_g)=min(f(\vec{X}_i))$, $\vec{X}_g=\vec{X}_i$
return $f(\vec{X}_g)$, $\vec{X}_g$

The novel algorithm is combined with SMS and QUATRE. It introduces matrices and adopts a new update strategy. The candidates evolve in the form of the matrix that makes the movement mechanism have good coordination. Thereby, CSMS improves competitiveness.

4. The proposed optimization-based embedding and extraction for digital watermarking with improved SMS

In this section, the proposed optimization-based embedding with the optimal embedding factors on DWT-SVD and extraction is described. This way is performed on the QR code.

4.1. Combine algorithms with DWT-SVD

The subsection mainly enhances the robustness utilizing our proposed algorithms to select the more suitable embedding factor based on DWT-SVD. The proposed method is a digital watermarking, the robustness of the original image may not be strong when attacked. For the sake of protecting the security of the watermark, it is encrypted. The resulting graph is as Fig. 2.

We first perform DWT on the carrier image and then select the appropriate sub-bands. Therefore, four pictures are obtained. Then the selected four sub-images and the watermark are decomposed using SVD. The key is that the embedding factors should consider both the quality of the image and the perfect extraction of the watermark.

[1] $a=\begin{bmatrix}a_{11} & a_{12}\\ a_{21} & a22\end{bmatrix}, b=\begin{bmatrix}b_{11} & b_{12}\\ b_{21} & b_{22}\end{bmatrix}\Rightarrow a\otimes b=\begin{bmatrix}a_{11}\times b_{11} & a_{12}\times b_{12}\\ a_{21}\times b_{21} & a_{22}\times b_{22}\end{bmatrix}.$

Fig. 2. The original watermark and encrypted watermark.

Table 2
Single-modal function in experiments.

Number	Function	Space	Dimension	fmin
1	$f_1(z) = \sum_{l=1}^{p} z_l^2$	$[-100, 100]$	30	0
2	$f_2(z) = \sum_{l=1}^{p} \lvert z_l \rvert + \prod_{l=1}^{p} \lvert z_l \rvert$	$[-10, 10]$	30	0
3	$f_3(z) = \sum_{l=1}^{p} \left(\sum_{j=1}^{l} z_j \right)^2$	$[-100, 100]$	30	0
4	$f_4(z) = max_l \lvert z_l \rvert, l \in [1, p]$	$[-100, 100]$	30	0
5	$f_5(z) = \sum_{l=1}^{p-1} \left[100 \left(z_{l+1} - z_l^2 \right)^2 + \left(z_l - 1 \right)^2 \right]$	$[-30, 30]$	30	0
6	$f_6(z) = \sum_{l=1}^{p} \left(\left[z_l + 0.5 \right] \right)^2$	$[-100, 100]$	30	0
7	$f_7(z) = \sum_{l=1}^{p} l \times z_l^2 + rand[0, 1)$	$[-1.28, 1.28]$	30	0

Finally, the watermark is combined with these sub-graphs employing the gained embedding factors which are calculated by our proposed algorithm. This method uses normalized correlation functions as the criterion to measure. We take two random ways from the five attacks. The five attack modes are cutting the image, increasing contrast, reducing contrast, product noise, Gaussian noise, and 45-degree rotation. After these tests, it should have the ability to resist attacks or interference to some extent.

4.1.1. Embedding process

The details of embedding watermark as follows.

Step 1: Decompose the image $\mathbf{I_o}$ into four sub-bands by three-level DWT.

Step 2: Use SVD to every sub-image.

Step 3: Apply SVD to decompose the watermark $\mathbf{W}$ and generate singular values.

Step 4: Changes the singular values of four sub-bands with the sub-images and the watermark decomposed.

Step 5: Modify the new DWT coefficients.

Step 6: Perform the inverse DWT technique about the obtained DWT coefficients, so that generate the watermarked image $\mathbf{I_w^*}$.

The embedding process is complete. Here, the familiar function is used to embed watermark as $C_i^w = C_i + q \times C_{iw}$. Here, C_i denotes the singular value of the cover image information, and C_{iw} is the singular value of watermark information undergoing SVD. The q presents the embedding strength factors which selected by sSMS-Pc and CSMS-Pc.

4.1.2. Extracting process

Similarly, according to the embedding method, the digital watermark extraction process with DWT-SVD is depicted.

Step 1: Generate four different sun-bands using DWT for the watermarked image $\mathbf{I_w^*}$.

Step 2: For each sub-image, take advantage of SVD to decompose.

Step 3: Refer to the embedding method determine the selected strength factor q with $C'_{iw} = \frac{(C_i^w - C_i)}{q}$.

Step 4: Obtain the modified DWT coefficient.

Step 5: Perform inverse DWT, and extract watermark $\mathbf{W}^*$.

According to the above process, the embedding factors play a role in DWT-SVD. q decides the invisibility of the watermark and robustness that image resists to attack. In a word, how to select the value of q is the key to this technique. Whereupon our study is mainly aimed at solving it. Applying the optimal solution q in our proposed algorithms plays a significant role in the embedding and extraction process. This work takes advantage of sSMS and CSMS to select embedding factors as suitable as possible to acquire better performance under some attack tests.

4.1.3. Novel algorithms optimize embedding factors

Recently, some people use PSO, Cuckoo Search (CS) algorithm, and Artificial Bee Colony (ABC) algorithm to select better embedding factor (Saxena et al., 2018; Dey et al., 2013; Sharma et al., 2019). The watermark will be embedded in the color image, but we want it not to be affected by attacks or distortion. The binary image embedded into the color image should not affect the original image and the extracted watermark.

The watermark signal is added into the carrier image by using the predefined function $f(*)$ for embedding and extracting in the process. Then, we can get the image containing a watermark. With embedded information, it is hoped to achieve a better balance between invisibility and robustness of watermarking. The choice of q is crucial, so this paper introduces our method to select the appropriate embedding factors.

In the wavelet domain, the method uses SVD to modify the wavelet coefficients with q, to achieve the watermark embedding. The improved SMS is applied to optimize q so that it achieves the adaptive process of embedding the watermark. In consequence, the measure should meet Eq. (14).

$$\max\ f(q) = \mathrm{NC}(\mathbf{I_o}, \mathbf{I_w^*}) + \mathrm{NC}(\mathbf{W}, \mathbf{W}^*), \tag{14}$$

where NC means the normalized correlation between two images, and it is a common measure in digital watermarking (Ali et al., 2014). This function is defined by Eq. (15) and it measures the similarity of the original image and the recovered image. $\mathrm{NC}(\mathbf{I_o}, \mathbf{I_w^*})$ denotes the correlation value between the original image and image embedded watermark. $\mathrm{NC}(\mathbf{W}, \mathbf{W}^*)$ is the correlation value between the original watermark and watermark that extracted. The value of NC is between 0 and 1. When the value equals 1, the quality is the best one. The bigger the value of NC is the stronger its resistance to attack. After the embedding process, some attacks are used to test the robustness which would be evaluated by NC.

$$\mathrm{NC}(\mathbf{C}, \mathbf{C}^*) = \frac{\sum_i^M \sum_l^M C(i,l) \times C^*(i,l)}{\sqrt{\sum_i^M \sum_l^M C(i,l)^2}\sqrt{\sum_i^M \sum_l^M C^*(i,l)^2}}, \tag{15}$$

where $\mathbf{C}$ and $\mathbf{C}^*$ denote the original image and the recovered image, respectively. M is the size of the image.

The paper takes lots of attacks and distortions into consideration, so we change Eq. (14) into Eq. (16). Consequently, Eq. (16) is regarded as the fitness function that finds the optimal solution among the optimization process.

$$\min\ f(q) = \frac{1}{\mathrm{NC}(\mathbf{I_o}, \mathbf{I_w^*})} - \sum_{i=1}^{num} \mathrm{NC}(\mathbf{W}, \mathbf{W_i^*}), \tag{16}$$

where num is the total number of attacks suffered. Our algorithm contains many particles that have no quality. These candidates try to find the best solution. The main optimization is finding the embedding strength factors. The process is as follows.

Step 1: Initialize basic parameters, the position $\mathbf{X}$ and velocity $\mathbf{V}$ of molecules.

Step 2: Evaluate the value of fitness function utilizing Eq. (16). Firstly, make $q = \vec{X}_i$. And then embed the watermark into carrier image, attack the resulted image and extract the watermark successively. Finally, generate the $\mathrm{NC}(\mathbf{W}, \mathbf{W}^*)$ and $\mathrm{NC}(\mathbf{I_o}, \mathbf{I_w^*})$ and calculate the fitness function.

Step 3: Update the position of the particle making use of sSMS or CSMS. Compare their fitness values with the global best location and select the optimal solution $\vec{X}_g$.

Step 4: Satisfy the termination condition and assign the optimal solution of molecule to the embed factor $q = \vec{X}_g$, where $q = (q_1, q_2, q_3, q_4)$.

Table 3
Multimodal functions in experiments.

Number	Function	Space	Dimension	fmin
8	$f_8(z) = \sum_{l=1}^{p} -z_l \times sin\left(\sqrt{\lvert z_l \rvert}\right)$	$[-500, 500]$	30	−12 569
9	$f_9(z) = \sum_{l=1}^{p} \left[z_l^2 - 10 \times cos\left(2\pi z_l\right) + 10\right]$	$[-5.12, 5.12]$	30	0
10	$f_{10}(z) = -20 \times exp\left(-0.2\sqrt{\frac{1}{p}\sum_{l=1}^{p} z_l^2}\right)$ $-exp\left(\frac{1}{p}\sum_{l=1}^{p} cos\left(2\pi z_l\right) + 20 + 2.718\right)$	$[-32, 32]$	30	0
11	$f_{11}(z) = \frac{1}{4000} \times \sum_{l=1}^{p} z_l^2 - \prod_{l=1}^{p} cos\left(\frac{z_l}{\sqrt{l}}\right) + 1$	$[-600, 600]$	30	0
12	$f_{12}(z) = \frac{\pi}{p} \times \left\{ 10 \times sin\left(\pi y_1\right) + \sum_{l=1}^{p-1}\left(y_l - 1\right)^2\left[1 + 10 \times sin^2(\pi y_{l+1})\right] + (y_p - 1)^2 \right\}$ $+ \sum_{l=1}^{p} u(z_l, 10, 100, 4),$ $y_l = 1 + \frac{z_l+1}{4} \times u(z_l, a, k, p) = \begin{cases} k(z_l - a), z > a \\ 0, -a < z_l < a \\ k(-z_l - a), z > a \end{cases}$	$[-50, 50]$	30	0
13	$f_{13}(z) = 0.1 \times \left\{ sin^2\left(3\pi z_1\right) + \sum_{l=1}^{p}\left(z_l - 1\right)^2\left[1 + sin^2\left(3\pi z_l + 1\right)\right] + \left(z_p - 1\right)^2\left[1 + sin^2\left(2\pi z_p\right)\right] \right\}$ $+ \sum_{l=1}^{p} u\left(z_l, 10, 100, 4\right)$	$[-50, 50]$	30	0

Table 4
Fixed-dimension multimodal functions in experiments.

Number	Function	Space	Dimension	fmin
14	$f_{14}(z) = \left(\frac{1}{500} \times \sum_{j=1}^{25} \frac{1}{j+\sum_{i=1}^{2}(z_i - a_{ij})^6}\right)^{-1}$	$[-65, 65]$	2	1
15	$f_{15}(z) = \sum_{l=1}^{11}\left[a_l - \frac{z_1(b_l^2 + b_l z^2)}{b_l^2 + b_l z^3 + z^4}\right]^2$	$[-5, 5]$	4	0.00030
16	$f_{16}(z) = 4z_1^2 - 2.1z_1^4 + \frac{1}{3}z_1^6$ $+ z_1 z_2 - 4z_2^2 + 4z_2^4$	$[-5, 5]$	2	−1.0316
17	$f_{17}(z) = \left[1 + \left(z_1 + z_2 + 1\right)^2 \times \left(19 - 14z_1 + 3z_1^2 - 14z_2 + 6z_1 z_2 + 3z_2^2\right)\right]$ $\times \left[30 + \left(2z_1 - 3z_2\right)^2 \times \left(18 - 32z_1 + 12z_1^2 + 48z_2 - 36z_1 z_2 + 27z_2^2\right)\right]$	$[-2, 2]$	2	3
18	$f_{18}(z) = -\sum_{l=1}^{4} c_l \times exp\left(-\sum_{j=1}^{3} a_{lj}\left(z_j - p_{lj}\right)^2\right)$	$[1, 3]$	3	−3.86
19	$f_{19}(z) = -\sum_{l=1}^{4} c_l \times exp\left(-\sum_{j=1}^{6} a_{lj}\left(z_j - p_{lj}\right)^2\right)$	$[0, 1]$	6	−3.32
20	$f_{20}(z) = -\sum_{l=1}^{5}\left[\left(Z - a_l\right)\left(Z - a_l\right)^T + c_l\right]^{-1}$	$[0, 10]$	4	−10.1532
21	$f_{21}(z) = -\sum_{l=1}^{7}\left[\left(Z - a_l\right)\left(Z - a_l\right)^T + c_l\right]^{-1}$	$[0, 10]$	4	−10.4028
22	$f_{22}(z) = -\sum_{l=1}^{10}\left[\left(Z - a_l\right)\left(Z - a_l\right)^T + c_l\right]^{-1}$	$[0, 10]$	4	−10.5363
23	$f_{23}(z) = \left(\sum_{l=1}^{5} l \times cos\left((l+1) z_1 + l\right)\right)$ $\times \left(\sum_{l=1}^{5} l \times cos\left((l+1) z_2 + l\right)\right)$	$[-5.12, 5.12]$	2	−186.7309

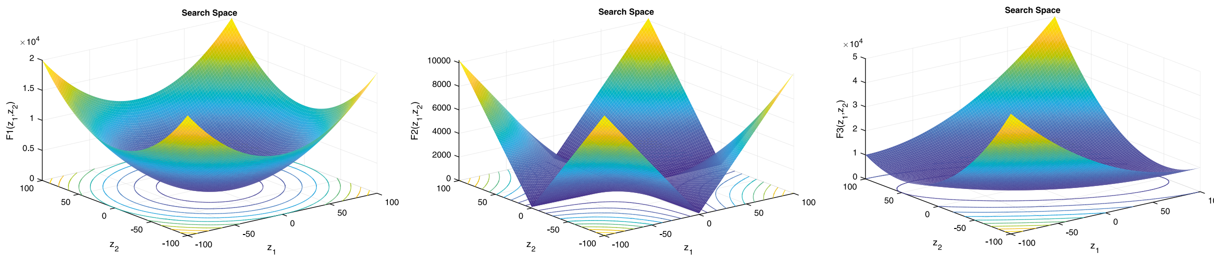
4.2. QR code watermarking with novel algorithms

The primary method of this QR code watermarking is to use two proposed sSMS and CSMS to optimize the embedding factors in DWT-SVD. The QR code is a two-dimensional symbol. Now, it can be seen and applied everywhere every day. It is also regarded as a picture in brief. It can carry a lot of data in a limited space and be read with high-speed reading. The QR code owns superiority in both performance and function. Recently, some people have beautified the appearance of QR codes. Cai et al. come up with the beautifying two-level QR code (Cai et al., 2019).

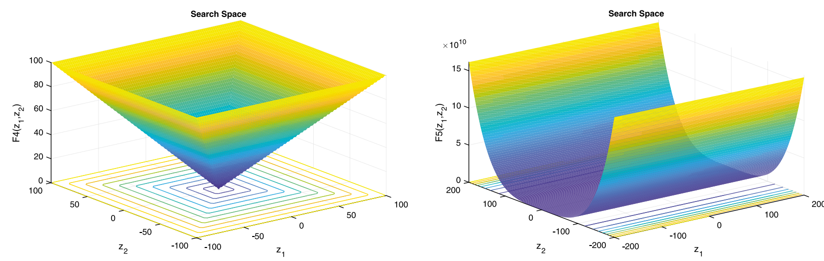
The QR code has other drawbacks, such as their fault tolerance is very low. In other words, it is difficult to embed the digital watermark into them. The embedded intensity factor affects whether the original QR code can be decoded as originally. Therefore, this paper uses improved DWT-SVD to realize two layers of encryption. The first level takes advantage of the characteristic of storing large amounts of data. This method possesses unique functional modules that could be resistant to interference and have the error correction functionality. The

Table 5
Parameter setting in this experiment.

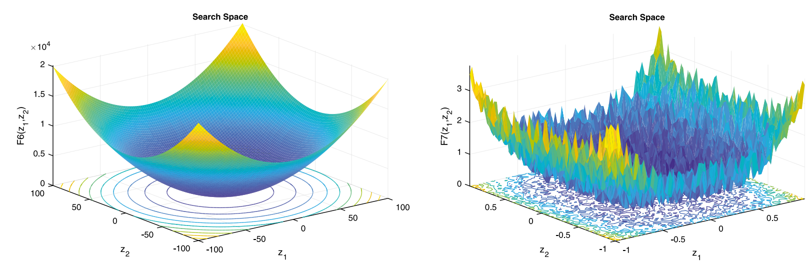
Algorithm	Parameter
PSO	$cp_1 = 2, cp_2 = 2, \omega_{max} = 0.9, \omega_{min} = 0.2.$
SMS	The gas phase: $\alpha = 0.9, \mu = 0.8, \beta = 0.8, T = 0.9;$ The liquid phase: $\alpha = 0.4, \mu = 0.4, \beta = 0.2, T = 0.2;$ The solid phase: $\alpha = 0.1, \mu = 0.1, \beta = 0, T = 0.$
QUATRE	$F = 0.8, \mathbf{M},$ "QUATRE/best/1": $\mathbf{B} = \mathbf{X}_{g,t} + F * (\mathbf{X}_{r2,t} - \mathbf{X}_{r3,t}).$
sSMS	The gas phase: $\alpha_{max} = 1, \alpha_{min} = 0.8, \mu = 0.8, \beta = 0.8, T = 0.9;$ The liquid phase: $\alpha_{max} = 0.6, \alpha_{min} = 0.3, \mu = 0.4, \beta = 0.2, T = 0.2;$ The solid phase: $\alpha_{max} = 0.1, \alpha_{min} = 0, \mu = 0.1, \beta = 0, T = 0.$
CSMS	$\mathbf{M};$ The gas phase: $\alpha_{max} = 1, \alpha_{min} = 0.8, \mu = 0.8, \beta = 0.8, T = 0.9;$ The liquid phase: $\alpha_{max} = 0.6, \alpha_{min} = 0.3, \mu = 0.4, \beta = 0.2, T = 0.2;$ The solid phase: $\alpha_{max} = 0.1, \alpha_{min} = 0, \mu = 0.1, \beta = 0, T = 0.$



(a) Single-modal function: 1-3

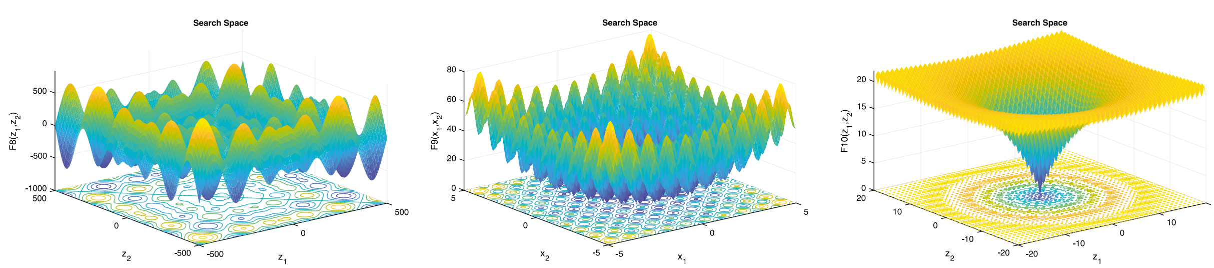


(b) Single-modal function: 4-5

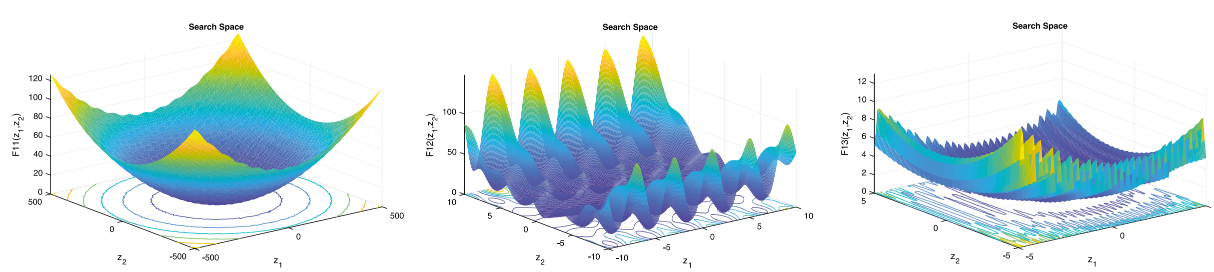


(c) Single-modal function: 6-7

Fig. 3. Search space of benchmark functions 1–7.



(a) Multimodal function: 8-10



(b) Multimodal function: 11-13

Fig. 4. Search space of benchmark functions 8–13.

Table 6
The comparison of results with five unimodal functions.

Function	Index	SMS	PSO	sSMS	QUATRE	CSMS
f1	BR	4.40E+04	4.36E+04	1.68E+03	1.20E−09	**2.07E−246**
	MEANB	4.16E+04	5.50E+04	1.87E+03	1.68E−09	**8.08E−151**
	SDB	3.61E+03	6.86E+03	1.90E+02	1.40E−09	**5.12E−150**
f2	BR	1.45E+05	3.35E+09	1.08E+02	6.99E−06	**1.81E−91**
	MEANB	4.16E+04	1.58E+09	93.26596957	6.80E−06	**6.45E−81**
	SDB	5.87E+06	9.46E+09	17.28492547	2.37E−06	**4.56E−80**
f3	BR	5.08E+04	5.75E+04	5.02E+03	5.58E+02	**2.68E−196**
	MEANB	4.68E+04	7.13E+04	5.01E+03	1.12E+03	**6.63E−168**
	SDB	6.52E+03	1.03E+04	5.70E+02	6.00E+02	**0**
f4	BR	69.222484166	87.921062993	20.278747693	1.938346536	**5.798784304e−232**
	MEANB	70.880428566	85.252044215	21.779277953	1.177985922	**4.395875729e−84**
	SDB	3.087496192	3.254571648	1.843893288	0.705758841	**3.106747110e−83**
f5	BR	9.90E+07	2.53E+08	1.74E+07	**22.16265412**	28.73455514
	MEANB	9.72E+07	1.98E+08	1.73E+07	39.83859082	**2.89E+01**
	SDB	1.68E+07	4.31E+07	1.31E+06	29.77145526	**0.079893651**
f6	BR	4.59E+04	5.29E+04	1.99E+03	**6.58E−10**	2.063800198
	MEANB	4.09E+04	5.56E+04	1.86E+03	**1.78E−09**	4.371738644
	SDB	3.47E+03	6.93E+03	2.60E+02	**1.63E−09**	2.128534056
f7	BR	53.70895897	67.27838225	0.1390219	0.022282218	**3.68E−05**
	MEANB	45.05498786	82.9858382	0.143531092	0.021768117	**3.70E−04**
	SDB	8.321939142	19.23080636	0.070933237	0.006548667	**2.78E−04**

Table 7
The comparison of results with multimodal functions.

Function	Index	SMS	PSO	sSMS	QUATRE	CSMS
f8	BR	−3.84E+03	−7.00E+03	**−3.61E+85**	−8.50E+71	−9.02E+26
	MEANB	−4.23E+03	−6.37E+03	**−1.25E+92**	−1.31E+77	−2.30E+25
	SDB	**3.07E+02**	4.80E+02	8.87E+92	6.60E+77	1.29E+26
f9	BR	3.20E+02	4.06E+02	1.89E+02	4.06E+02	**0**
	MEANB	3.35E+02	3.95E+02	2.03E+02	37.07619095	**0**
	SDB	14.00415159	24.9118145	21.44880967	8.914930587	**0**
f10	BR	19.88531637	19.96057431	19.00339328	3.93E−06	**8.88E−16**
	MEANB	19.78232656	20.0110004	18.86183954	1.20E−05	**8.88E−16**
	SDB	0.145084306	0.181917813	0.248285047	5.14E−06	**0**
f11	BR	3.10E+02	5.09E+02	35.96432698	3.62E−09	**0**
	MEANB	3.71E+02	5.10E+02	33.44238652	0.004285776	**0**
	SDB	39.39704112	58.69853133	6.457585824	0.006128183	**0**
f12	BR	1.04E+08	4.62E+08	4.65E+05	**1.46E−10**	0.11142029
	MEANB	1.75E+08	4.37E+08	5.61E+05	**0.057810883**	0.186620164
	SDB	4.18E+07	1.16E+08	1.61E+05	0.235470403	**0.102909958**
f13	BR	5.02E+08	8.09E+08	5.49E+06	**3.37E−09**	1.156880177
	MEANB	3.93E+08	7.80E+08	6.76E+06	**1.40E−09**	1.211088152
	SDB	7.55E+07	1.89E+08	1.03E+06	**1.41E−09**	7.55E+07

second layer uses digital watermarking to embed a watermark into the QR code with adaptive embedding factors q. Ultimately, the obtained QR code with a watermark can still be decoded correctly.

5. Experiment and performance analysis

This section mainly discusses the evaluation of the proposed optimization-based embedding factors for digital watermarking. It is mainly divided into three parts: benchmark test functions, simulation of novel DWT-SVD technique, the application in the QR code.

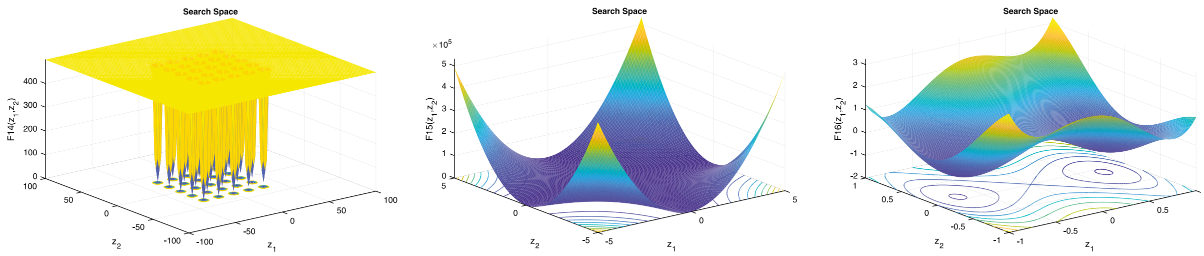
5.1. Experiment configuration

The simulation still takes "Lena" as the original carrier image which is shown in Fig. 6(a). Similarly, the original watermark is on the left of Fig. 2. The paper carries out the environment of the simulation in Matlab R2018a. The type of DWT transform is the Haar wavelet. Haar wavelet is a kind of wavelet, and it is one of the most simple orthogonal normalized wavelet. Comparing with other types of wavelet, it has the characteristic of relatively low calculation and maintains better edge information.

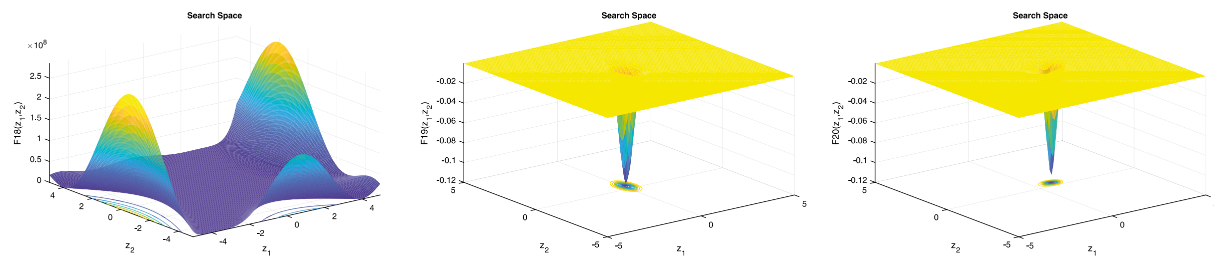
5.2. Benchmark function testing for two algorithms

The experiment test is discussed in this subsection. The 23 benchmark functions are used to test the performance of sSMS and CSMS (Pan et al., 2019). These selected functions are divided into three types. There are single-modal function, multi-modal function, and fixed-dimension multi-modal function. In such tables, Table 2 shows unimodal test functions and Table 3 contains multimodal test functions. Multimodal test functions with fixed dimensions are included at Table 4. Besides Figs. 3–5 portray the search space model of these benchmark functions, respectively.

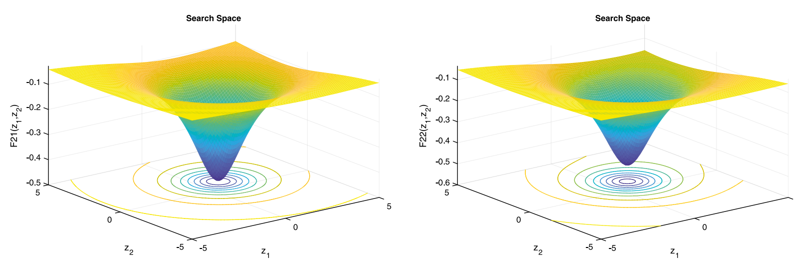
Under twenty-three test functions, two improved schemes are compared with PSO, original SMS, and QUATRE. This experiment sets the population size ps to 60. In the process of evaluation, the maximum number of iteration I_{max} is 600. The total number of iterations is set to this value in many experiments. Compared with $I_{max} = 600, I_{max} = 800$ and $I_{max} = 1000$, it can be seen that at the $t = 0.8 * I_{max}$ it gets convergence and the optimum is reached. This I_{max} is set to the minimum value 600. Some parameters in the algorithm are unique to them. All parameters set in this paper are described in Table 5. The cp_1 and cp_2 are the learning factors in PSO. The weight factor decreases linearly from w_{max} to w_{min}.



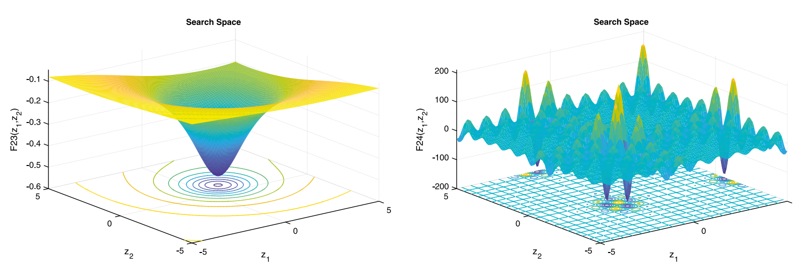
(a) Fixed-dimension multimodal function: 14-16



(b) Fixed-dimension multimodal function: 17-19



(c) Fixed-dimension multimodal function: 20-21



(d) Fixed-dimension multimodal function: 22-23

Fig. 5. Search space of benchmark functions 14–23.

After running 50 times, the results of each algorithm are gained. Multimodal functions can efficiently reflect the performance of avoiding poor local optima and find the global optimal solution. Table 6, Table 7, and Table 8 represent the results of algorithms including the best solution, the average value of optimal and the standard deviation best-so-far. In these tables, BR means the best value so far. The MEANB is the mean value of the best results under 50 tests. The SDB signifies the standard deviation. After comparing these results of five algorithms, the best one is highlighted with every benchmark function in the table. It can be known from the comparison results, sSMS wins five times on MEANB and wins nine times on SDB. QUATRE has ten best results and CSMS has eleven best solutions. In the meanwhile, CSMS behaves better in the mean and standard than the other four algorithms. This suggests that these two schemes acquire excellent effect. CSMS especially shows the overall stability and optimization ability.

5.3. Simulation of novel DWT-SVD with adaptive q

The subsection introduces simulation of the optimized DWT-SVD. The embedding factor q is vital in the DWT-SVD because it decides the quality of the watermarked image.

The experiment puts sSMS and CSMS into use in choosing the embedding factors. In addition, they are compared with PSO, CS, and ABC. To support the watermarking scheme in accomplishing these objectives, Nitin Saxena et al. find suitable watermark strength using Dynamic-PSO (DPSO) for colored images (Saxena et al., 2018). Nilanjan Dey et al. design a robust biomedical content authentication system by embedding the logo of the hospital utilizing both discrete wavelet transformation and CS (Dey et al., 2013). Sourabh Sharma et al. propose an adaptive color image watermarking using ABC (Sharma et al., 2019).

In this current work, this paper proposes a new robust and adaptive watermarking scheme using our sSMS and CSMS. When selecting the

Table 8
The comparison of results with fixed-dimension multimodal functions.

Function	Index	SMS	PSO	sSMS	QUATRE	CSMS
f14	BR	1.53025774	0.999931298	**0.998003838**	**0.998003838**	**0.998003838**
	MEANB	1.050096976	**0.998003838**	**0.998003838**	**0.998003838**	0.9988414
	SDB	0.133617298	0.00312512	1.79E−12	**0**	0.002296556
f15	BR	0.00213685	0.001263835	0.001026981	9.82E−04	**8.51E−04**
	MEANB	0.002969886	0.006037907	0.001635347	8.32E−04	**6.36E−04**
	SDB	0.001327377	0.008144535	8.53E−04	**1.92E−04**	3.66E−04
f16	BR	−1.028384367	−1.030796735	**−1.031628453**	**−1.031628453**	−1.031614126
	MEANB	−1.027046386	−1.030256928	−1.031628395	−1.031628453	**−1.031438453**
	SDB	0.004594045	0.001373587	4.12E−07	**2.99E−16**	2.54E−04
f17	BR	3.076398265	3.025486302	**3**	**3**	3.000224685
	MEANB	3.0983746	3.006495265	**3**	**3**	3.004129783
	SDB	0.106319512	0.005225648	**2.06E−08**	8.74E−07	0.008593892
f18	BR	−3.855782389	−3.862677967	**−3.86278**	**−3.86278**	−3.858342133
	MEANB	−3.856730093	−3.860803384	−3.862782083	**−3.862782148**	−3.859040168
	SDB	0.00385967	0.002412909	1.04E−07	3.14E−15	**0**
f19	BR	−3.009168899	−3.140337949	**−3.321737495**	−3.20310205	−3.123778543
	MEANB	−3.061997428	−3.126449023	−3.266395647	**−3.269682198**	−3.099702616
	SDB	0.068856324	0.065970191	0.060161376	0.059616166	**0.054886561**
f20	BR	−2.880086613	−7.225487027	**−9.794841576**	−5.10077214	−3.617900914
	MEANB	−3.249422707	−4.084886508	−7.94305722	**−8.014875633**	−4.406848276
	SDB	1.004695309	1.223635537	2.054229703	2.827854614	**1.197282852**
f21	BR	−2.672857955	−4.834055704	−9.472189498	**−10.40294057**	−3.020024534
	MEANB	−3.685258518	−4.707521632	−8.644216595	**−9.00415919**	−4.704667509
	SDB	**0.897971745**	1.688944158	1.788673716	2.708135353	1.323712989
f22	BR	−4.37752623	−3.600924777	**−10.53640982**	−9.765206383	−4.883691601
	MEANB	−3.838353206	−4.729523155	**−10.05358577**	−8.660317039	-4.614450413
	SDB	**1.102529388**	1.646121488	1.488592928	1.906481906	1.358816347
f23	BR	−1.85E+02	**−1.87E+02**	**−1.87E+02**	**−1.87E+02**	**−1.87E+02**
	MEANB	−1.86E+02	−1.86E+02	**−1.87E+02**	**−1.87E+02**	**−1.87E+02**
	SDB	0.981427741	0.362352198	0.205542485	**4.52E−06**	0.182068848

(a) (b)

Fig. 6. (a) The original image; (b) The watermarked image.

Fig. 7. The original QR code which has beautified.

Table 9
Results from different algorithms.

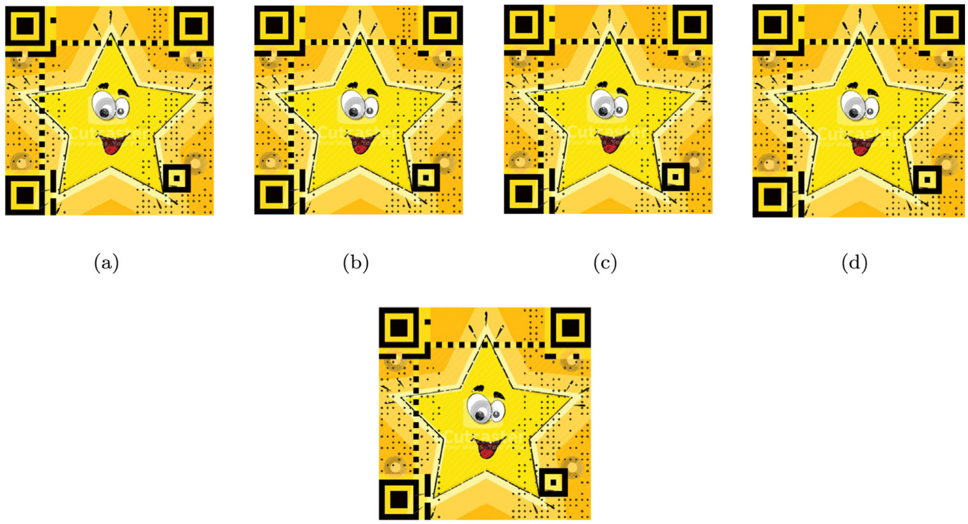
Algorithm	q_1	q_2	q_3	q_4	NC	PSNR
PSO (Saxena et al., 2018)	5	5	1.8640	2.0513	1	48.9
QUATRE	6.1787	9.7501	5.5509	5.5661	1	51.2318
CS (Dey et al., 2013)	9.4927	3.0290	9.6003	9.9376	1	48.5926
ABC (Sharma et al., 2019)	6.6859	3.2701	6.2582	1.8473	1	49.0384
sSMS	3.9757	4.6927	1.2515	3.2760	1	49.7056
CSMS	3.0267	0.8548	1.8269	2.2102	1	**53.0032**

Table 10
There are results of digital watermarking in a QR code.

Algorithm	NC	PSNR
PSO	1	50.25
QUATRE	1	48.50
CS	1	49.20
ABC	1	46.22
sSMS	1	49.46
CSMS	1	54.54

frequency bands of the original image, the tertiary decomposition of DWT is adopted. Finally, the four sub-graphs of different frequency bands are selected. It determines that there are 4 variables in the set of q, that is, the dimension of each particle in our algorithms is 4. Each variable represents the possible embedding strength factors for four images belong to different sub-bands. The population size is 10. The search range of particles is (0, 10] according to the actual embedding case. Fig. 6(b) is the final watermarked image.

Image quality assessment can be divided into subjective and objective evaluation methods. The subjective method is scored by the observer to the image quality. But it has a high workload, is time-consuming, and inconvenient to use. Therefore, the paper employs NC and PSNR as the criterion for quality evaluation. PSNR denotes the invisibility of the watermark. It compares the original carrier and the image that carried watermark. PSNR can described as Eq. (17). If its value is big, the difference between them is small so that the effect of



(a) (b) (c) (d) (e)

Fig. 8. QR code with the encrypted watermark. (a) QR code with watermark using PSO; (b) QR code with watermark using CS; (c) QR code with watermark using ABC; (d) QR code with watermark using sSMS; (e) QR code with watermark using CSMS.

(a) (b) (c) (d) (e)

Fig. 9. Extract watermark from QR code. (a) Extract using PSO; (b) Extract using CS; (c) Extract using ABC; (d) Extract using sSMS; (e) Extract using CSMS.

digital watermarking is excellent.

$$\mathrm{PSNR(dB)} = 10 \times \log_{10} \frac{I_{o\,max}^{2}}{\frac{1}{m \times m} \sum_i^m \sum_r^m [I_o(i,r) - I_w^*(i,r)]^2} \mathrm{(dB)}. \quad (17)$$

Table 9 contains the results of DWT-SVD with five algorithms. sSMS and CSMS behave better on PSNR in these algorithms. At the same experimental environment, the comparison of results shows intuitively the superiority of the adaptive embedding strength factors obtained by these optimized algorithms including PSO (Saxena et al., 2018), CS (Dey et al., 2013), ABC (Sharma et al., 2019). In addition, the optimal solution with q is listed. The invisibility and robustness are compared. It can be seen that the proposed methods have stronger resistance and can be well implemented.

5.4. Experiment on QR code

Using new DWT-SVD with optimal embedding factors realizes the QR code watermarking in this subsection. We apply three-level decomposition DWT to embed a watermark into low-frequency sub-band coefficients. The embedding factors are obtained taking advantage of sSMS and CSMS. Depending on two novel methods, it hopes that they could realize secure information hiding with the QR code. The embedding process is divided into two steps. Firstly, the "Shandong University of Science and Technology" is hidden in a QR code and then beautify this QR code. The result is shown in Fig. 7. Lastly, we perform our study on this beautified QR code. The generated QR code is shown in Fig. 8.

The extraction process is the same as the common picture. Fig. 9 is the watermark extracted from Fig. 8 according to different algorithms: PSO, CS, ABC, sSMS, and CSMS, respectively. The watermark is embedded into QR code which has beautified, the above experiment confirms that there is no effect on decoding. We scan the QR code with watermark and then "Shandong University of Science and Technology" is still obtained.

Table 10 displays the PSNR and NC after embedding the watermark. These data explain that embedding and extraction work well. It can be seen from Table 10 that the value of NC is 1, which means that the watermark can be completely extracted under different algorithms. All of them have low distortion. It can also be seen from Table 10 that PSNR has improved using sSMS or CSMS compared with CS and ABC, which shows that our algorithms have better imperceptibility. Therefore, we can say that the performance of this novel digital watermarking is remarkably effective.

6. Conclusions and future prospect

The study puts up with two novel algorithms based on basic SMS. CSMS and sSMS are combined with DWT-SVD in digital watermarking for low distortion and robustness like PSO (Saxena et al., 2018), CS (Dey et al., 2013), and ABC (Sharma et al., 2019). To enhance the robustness, the watermark is embedded into the low-frequency DWT coefficients. These coefficients are decomposed with SVD. Based on the better embedding scaling factors using our algorithms, the watermark information is embedded into the selected four different sub-bands of the host image. The algorithm considers the good balance of invisibility and anti-attack in the whole process.

It has no impact on decoding the QR code normally after embedding a watermark into QR code. According to the numerical value of the experimental results, it can be seen that our proposed algorithms have lower distortion and imperceptibility. They obtain excellent invisibility and robustness. As a consequence, we propose good optimization-based schemes with adaptive optimal factor in digital watermarking. In the future, we dedicate to not only improve the robustness of watermark and invisibility but also focus on the storage capacity of information hiding.

CRediT authorship contribution statement

Jeng-Shyang Pan: Conceptualization, Methodology, Supervision. **Xiao-Xue Sun:** Methodology, Software, Writing - original draft. **Shu-Chuan Chu:** Data curation, Methodology. **Ajith Abraham:** Writing - review & editing. **Bin Yan:** Software, Writing - original draft.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Abraham, A., Nedjah, N., de Macedo Mourelle, L., 2006. Evolutionary Computation: from Genetic Algorithms to Genetic Programming. Genetic Systems Programming. Springer, pp. 1–20.

Ali, M., Ahn, C.W., Siarry, P., 2014. Differential evolution algorithm for the selection of optimal scaling factors in image watermarking. Eng. Appl. Artif. Intell. 31, 15–26.

Bhatnagar, G., Raman, B., 2009. A new robust reference watermarking scheme based on DWT-SVD. Comput. Stand. Interfaces 31 (5), 1002–1013.

Cai, H.-L., Yan, B., Chen, N., Pan, J.-S., Yang, H.-M., 2019. Beautified QR code with high storage capacity using sequential module modulation. Multimedia Tools Appl. 78 (16), 22575–22599.

Chu, S.-C., Tsai, P.-W., Pan, J.-S., 2006. Cat swarm optimization. In: Pacific Rim International Conference on Artificial Intelligence. Springer, pp. 854–858.

Corchado, E., Abraham, A., de Carvalho, A., 2010. Hybrid intelligent algorithms and applications. Inf. Sci.—Inform. Comput. Sci. Intell. Syst. Appl.: Int. J. 180 (14), 2633–2634.

Cuevas, E., Echavarría, A., Ramírez-Ortegón, M.A., 2014. An optimization algorithm inspired by the states of matter that improves the balance between exploration and exploitation. Appl. Intell. 40 (2), 256–272.

Das, S., Abraham, A., Konar, A., 2008. Particle swarm optimization and differential evolution algorithms: Technical analysis, applications and hybridization perspectives. In: Advances of Computational Intelligence in Industrial Systems. Springer, pp. 1–38.

Dey, N., Samanta, S., Yang, X.-S., Das, A., Chaudhuri, S.S., 2013. Optimisation of scaling factors in electrocardiogram signal watermarking using cuckoo search. Int. J. Bio-Inspired Comput. 5 (5), 315–326.

Du, Z.-G., Pan, J.-S., Chu, S.-C., Luo, H.-J., Hu, P., 2020. QUasi-Affine TRansformation evolutionary algorithm with communication schemes for application of RSSI in wireless sensor networks. IEEE Access 8, 8583–8594.

Duan, H., Qiao, P., 2014. Pigeon-inspired optimization: a new swarm intelligence optimizer for air robot path planning. Int. J. Intell. Comput. Cybern. 7 (1), 24–37.

ElShafie, D.R., Kharma, N., Ward, R., 2008. Parameter optimization of an embedded watermark using a genetic algorithm. In: 2008 3rd International Symposium on Communications, Control and Signal Processing. IEEE, pp. 1263–1267.

Emary, E., Yamany, W., Hassanien, A.E., Snasel, V., 2015. Multi-Objective Gray-Wolf Optimization for Attribute Reduction. 65, Elsevier, pp. 623–632,

Ganic, E., Eskicioglu, A.M., 2004. A DFT-Based Semi-blind Multiple Watermarking Scheme for Images. CUNY Brooklyn College 2900.

Hu, P., Pan, J.-S., Chu, S.-C., Chai, Q.-W., Liu, T., Li, Z.-C., 2019. New hybrid algorithms for prediction of daily load of power network. Appl. Sci. 9 (21), 4514.

Hu, X.-M., Zhang, J., Chen, H., 2014. Optimal vaccine distribution strategy for Different Age Groups of Population: A Differential Evolution Algorithm Approach. Math. Probl. Eng. 2014.

Kennedy, J., Eberhart, R., 1995. Particle swarm optimization. In: Proceedings of ICNN'95-International Conference on Neural Networks, Vol. 4. IEEE, pp. 1942–1948.

Kim, D.H., Abraham, A., Cho, J.H., 2007. A hybrid genetic algorithm and bacterial foraging approach for global optimization. Inform. Sci. 177 (18), 3918–3937.

Kundur, D., Hatzinakos, D., 1998. Digital watermarking using multiresolution wavelet decomposition. In: Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP'98 (Cat. No. 98CH36181). 5, IEEE, pp. 2969–2972.

Liu, R., Tan, T., 2002. An SVD-based watermarking scheme for protecting rightful ownership. IEEE Trans. Multimed. 4 (1), 121–128.

Liu, T., Yan, B., Pan, J.-S., 2019. Color visual secret sharing for QR code with perfect module reconstruction. Appl. Sci. 9 (21), 4670.

Lu, L., Sun, X., Cai, L., 2010. A robust image watermarking based on DCT by Arnold transform and spread spectrum. In: 2010 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE), Vol. 1. IEEE, pp. V1–198.

Meng, Z., Pan, J.-S., 2018. QUasi-Affine transformation evolution with external archive (QUATRE-EAR): an enhanced structure for differential evolution. Knowl.-Based Syst. 155, 35–53.

Meng, Z., Pan, J.-S., Kong, L., 2018. Parameters with adaptive learning mechanism (PALM) for the enhancement of differential evolution. Knowl.-Based Syst. 141, 92–112.

Meng, Z., Pan, J.-S., Tseng, K.-K., 2019. PaDE: An enhanced differential evolution algorithm with novel control parameter adaptation schemes for numerical optimization. Knowl.-Based Syst. 168, 80–99.

Meng, Z., Pan, J.-S., Xu, H., 2016. QUasi-Affine TRansformation Evolutionary (QUATRE) algorithm: a cooperative swarm based algorithm for global optimization. Knowl.-Based Syst. 109, 104–121.

Mirjalili, S., Mirjalili, S.M., Hatamlou, A., 2016. Multi-verse optimizer: a nature-inspired algorithm for global optimization. Neural Comput. Appl. 27 (2), 495–513.

Mohammad, A.A., Alhaj, A., Shaltaf, S., 2008. An improved SVD-based watermarking scheme for protecting rightful ownership. Signal Process. 88 (9), 2158–2180.

Nabil, E., 2016. A modified flower pollination algorithm for global optimization. Expert Syst. Appl. 57, 192–203.

Nguyen, T.-T., Pan, J.-S., Dao, T.-K., 2019. An improved flower pollination algorithm for optimizing layouts of nodes in wireless sensor network. IEEE Access 7, 75985–75998.

O'Ruanaidh, J., Dowling, W., Boland, F., 1996. Watermarking digital images for copyright protection. IEE Proc.-Vis. Image Signal Process. 143 (4), 250–256.

Pan, J.-S., Hu, P., Chu, S.-C., 2019. Novel parallel heterogeneous Meta-Heuristic and its communication strategies for the prediction of wind power. Processes 7 (11), 845.

Pan, J.-S., Meng, Z., Xu, H., Li, X., 2016. QUasi-Affine TRansformation Evolution (QUATRE) algorithm: A new simple and accurate structure for global optimization. In: International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems. Springer, pp. 657–667.

Pan, J.-S., Song, P.-C., Chu, S.-C., Peng, Y.-J., 2020. Improved compact cuckoo search algorithm applied to location of drone logistics hub. Mathematics 8 (3), 333.

Patvardhan, C., Kumar, P., Lakshmi, C.V., 2018. Effective color image watermarking scheme using YCbCr color space and QR code. Multimedia Tools Appl. 77 (10), 12655–12677.

Samanta, S., Choudhury, A., Dey, N., Ashour, A., Balas, V., 2017. Quantum-inspired evolutionary algorithm for scaling factor optimization during manifold medical information embedding. In: Quantum Inspired Computational Intelligence. Elsevier, pp. 285–326.

Saxena, N., Mishra, K., Tripathi, A., 2018. DWT-SVD-based color image watermarking using dynamic-PSO. In: Advances in Computer and Computational Sciences. Springer, pp. 343–351.

Sharma, S., Sharma, H., Sharma, J.B., 2019. An adaptive color image watermarking using RDWT-SVD and artificial bee colony based quality metric strength factor optimization. Appl. Soft Comput. 84, 105696.

Sun, X.-X., Pan, J.-S., Chu, S.-C., Hu, P., Tian, A.-Q., 2020. A novel pigeon-inspired optimization with QUasi-Affine TRansformation evolutionary algorithm for DV-Hop in wireless sensor networks. Int. J. Distrib. Sens. Netw. 16 (6), 1550147720932749.

Tian, A.-Q., Chu, S.-C., Pan, J.-S., Cui, H., Zheng, W.-M., 2020. A compact pigeon-inspired optimization for maximum short-term generation mode in cascade hydroelectric power station. Sustainability 12 (3), 767.

Tsai, P.-W., Pan, J.-S., Chen, S.-M., Liao, B.-Y., Hao, S.-P., 2008. Parallel cat swarm optimization. In: 2008 International Conference on Machine Learning and Cybernetics. 6, IEEE, pp. 3328–3333.

Van Schyndel, R.G., Tirkel, A.Z., Osborne, C.F., 1994. A digital watermark. In: Proceedings of 1st International Conference on Image Processing. IEEE, pp. 86–90.

Yang, Y., Sun, X., Yang, H., Li, C.-T., 2008. Removable visible image watermarking algorithm in the discrete cosine transform domain. J. Electron. Imaging 17 (3), 033008.