IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 21 May 2024, accepted 7 July 2024, date of publication 12 July 2024, date of current version 24 July 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3427357

RESEARCH ARTICLE

Novel Integer Reversible Charlier Transform for Image Reversible Data Hiding Application

MOHAMED YAMNI[1], ACHRAF DAOUI[2], PAWEŁPŁAWIAK[3,4], OSAMA ALFARRAJ[5], AND AHMED A. ABD EL-LATIF[6,7], (Senior Member, IEEE)

[1]Dhar El Mahrez Faculty of Science, Sidi Mohamed Ben Abdellah-Fez University, Fes 30003, Morocco
[2]National School of Applied Sciences, Sidi Mohamed Ben Abdellah-Fez University, Fes 30003, Morocco
[3]Department of Computer Science, Faculty of Computer Science and Telecommunications, Cracow University of Technology, 31-155 Krakow, Poland
[4]Institute of Theoretical and Applied Informatics, Polish Academy of Sciences, 44-100 Gliwice, Poland
[5]Computer Science Department, Community College, King Saud University, Riyadh 11437, Saudi Arabia
[6]Faculty of Computing, Information Countermeauser Technique Institute, School of Cyberspace Science, Harbin Institute of Technology, Harbin 150001, China
[7]Department of Mathematics and Computer Science, Faculty of Science, Menoufia University, Shebeen El-Kom 32511, Egypt

Corresponding author: Ahmed A. Abd El-Latif (ahmedabdellatif@ieee.org)

This work was supported by the Researchers Supporting Project, King Saud University, Riyadh, Saudi Arabia, under Grant RSP2024R102.

ABSTRACT The discrete Charlier moment transform, while extensively utilized in image processing, is inherently lossy and non-integer reversible, making it unsuitable for lossless image applications. To address this, we propose the Integer Reversible Charlier Transform (IRCT), which operates on integer values and produces integer coefficients, enabling perfect and unique recovery of the original input data. The IRCT maintains the orthogonality and invertibility ensuring exact similarity between original and reconstructed images. We leverage the capabilities of the IRCT to develop a novel reversible data hiding (RDH) scheme. This scheme embeds additional data into images by modifying the histogram in the IRCT domain. By capitalizing on the concentrated nature of the IRCT histogram, characterized by high peaks, our method achieves a significantly high embedding capacity while preserving image quality and robustness to statistical attacks. Comparative performance evaluations underscore the effectiveness of the IRCT-based RDH scheme over existing techniques across various domains, positioning it as a promising solution for secure data transmission.

INDEX TERMS Discrete orthogonal moments, integer reversible Charlier transform, secure data transmission, reversible data hiding, histogram modification.

I. INTRODUCTION

In recent years, digital data hiding has emerged as a practical solution for ensuring secure transmission of digital data [1], [2], [3]. This technique involves concealing a secret message within a cover medium, such as digital images, in a manner that remains imperceptible and can only be detected and decoded by authorized individuals [4], [5].

For sensitive images, particularly those used in fields like medicine or the military, it is crucial to fully restore the original cover image after extracting the hidden message [6]. This requirement renders traditional data hiding algorithms unsuitable for such images [3]. Consequently, researchers have turned their attention to reversible data hiding (RDH) methods, dedicating their efforts to advancing this area of study [5], [7], [8].

The associate editor coordinating the review of this manuscript and approving it for publication was Jiachen Yang.

RDH methods can be categorized into three distinct types: spatial domain methods, compression domain methods, and transform domain methods [9]. The first category directly embeds the secret message into the cover image, while the second category utilizes compression techniques for embedding. The third category involves transforming the cover image into the transform domain before embedding the secret message.

Spatial and compression domain RDH methods are relatively less robust to statistical analysis because they directly manipulate the pixel intensity in the cover image [10], [11]. In contrast, transform domain RDH methods offer a higher level of robustness by manipulating the coefficients of the transformed cover image. Consequently, even if a malicious individual attempts to analyze the image to uncover hidden information, it would be more challenging for them to detect the secret message. This has led to a growing interest in employing transforms in RDH approaches [12], [13].

Among the various transforms used in image processing, moment transforms stand out as a popular class extensively employed for characterizing image features in different information security domains. These transforms serve as powerful tools and find applications in image watermarking [14], [15], image encryption [16], image zero-watermarking [17], [18], and image steganography [2], [19].

In recent years, significant progress has been made in reversible data hiding (RDH) techniques. One notable method proposed by Ni et al. [20] is histogram modification, which enhances image quality by modifying the histogram of the cover image to embed a secret message. However, spatial domain approaches [21], [22] based on Ni et al.'s technique [20] have limited embedding capacity due to their reliance on histogram peak points. As a result, researchers have shifted their focus towards transform domain methods [12], [23], [24] that leverage histogram modification to overcome these limitations.

Despite these advancements, moment transforms have not been extensively explored in the context of RDH. This is primarily because moment transforms are inherently lossy and lack integer reversibility. When applied to a digital image, which is represented by integer values, these transforms produce moment coefficients as floating-point real numbers. Consequently, during the inverse moment transform, these coefficients are rounded based on finite precision used in computations, resulting in potential information loss. Consequently, the reconstructed images never attain perfect similarity to the original images.

This paper introduces the Integer Reversible Charlier Transform (IRCT), which addresses the critical limitations of the discrete Charlier moment transform in lossless image applications [25]. The IRCT operates on integer-valued signals and produces integer-valued output coefficients, ensuring precise and unique recovery of the original input signal. This property is particularly valuable in applications where numerical accuracy is crucial and floating-point operations are undesirable, such as RDH application. The IRCT is based on discrete orthogonal Charlier polynomials [26], known for their native discreteness. Additionally, a Charlier polynomial matrix of size N $\times$ N possesses desirable properties, including orthogonality, invertibility, the equality of its inverse and transpose, a determinant of 1, and all leading principal submatrix minors being 1s. Exploiting these properties, we factorize an N $\times$ N Charlier polynomial matrix into a product of single-row elementary reversible matrices (SERMs) [27], each of which is directly integer reversible. When applied to a digital image, the IRCT achieves perfect reconstruction, resulting in infinite PSNR (peak signal-to-noise ratio), zero MSE (Mean Square Error), and an SSIM 1 (Structural Similarity Index) of 1, indicating exact similarity between the original and reconstructed images.

Furthermore, this paper presents a novel image RDH scheme for secure data transmission using the IRCT and histogram modification. The proposed method modifies the histogram in the IRCT domain, leveraging the concentrated nature of the IRCT histogram, which has high peaks, thereby achieving a very high embedding capacity while preserving image quality and robustness to statistical attacks.

The effectiveness of the proposed IRCT-based RDH method is extensively evaluated and compared with state-of-the-art RDH methods across multiple domains, including spatial, discrete cosine transform (DCT), discrete wavelet transform (DWT), and integer transform domains. Simulation results demonstrate that the IRCT-based RDH method offers a high embedding capacity, superior image quality, and robust resistance to statistical attacks.

The main contributions of this paper are as follows:

- Introduction of a novel transform, the Integer Reversible Charlier Transform (IRCT), which overcomes the limitations of the discrete Charlier moment transform in lossless image applications.
- The IRCT achieves perfect reconstruction of digital images, characterized by an infinite PSNR, zero MSE, and an SSIM of 1, ensuring exact similarity between the original and reconstructed images.
- Proposition of a novel RDH scheme based on the IRCT for secure data transmission.
- The proposed RDH scheme leverages the highly concentrated nature of the IRCT histogram, characterized by high peaks, to achieve significantly high embedding capacity while preserving image quality and robustness against statistical attacks.

The remainder of this work is organized as follows: Section II provides background information on the Charlier transform, Section III outlines the procedure for deriving the IRCT, Section IV explains the IRCT-based RDH method, Section V evaluates the effectiveness of the proposed method, and Section VI concludes the paper by summarizing its main contributions.

II. PRELIMINARIES

The Charlier transform, also referred to as the Charlier moment transform, is a mathematical tool widely used for extracting features from discrete data like signals, images, and 3D objects. It's crucial in many areas of image and signal processing, like speech enhancement [28], signal compression and encryption [29], signal and image reconstruction [26], 3D object classification [30], as well as texture classification, image segmentation, and denoising [31]. It also finds relevance in the field of control systems engineering [32].

The kernel function of the Charlier transform is based on discrete Charlier polynomials, which are a set of orthogonal polynomials defined for discrete variables. The Charlier polynomial of order n is defined as follows [25], [26]:

$$C_n^{a_1}(x) = \sqrt{\frac{w(x)}{\rho(n)}}\,{}_2F_0(-n, -x; -1/a_1); n,$$
$$x = 0, 1, 2 \ldots, \infty;\ a_1 > 0. \quad (1)$$

where $w(x)$ is the weight function (Eq. 2), $\rho(n)$ is the norm function (Eq. 3), and ${}_2F_0()$ is the hypergeometric function (Eq. 4).

$$w(x) = \frac{e^{-a_1} a_1^x}{x!}. \tag{2}$$

$$\rho(n) = \frac{n!}{a_1^n}. \tag{3}$$

$${}_2F_0(z, y; ; x) = \sum_{k=0}^{n} \frac{(z)_k (y)_k (x)^k}{k!}. \tag{4}$$

here $(z)_k$ is the Pochhammer symbol where $(z)_0 = 1$ and $(z)_k = z.(z+1)\ldots(z+k-1), k > 1$.

The Charlier polynomial $C_n^{a_1}(x)$ exhibits the orthogonal property described as follows:

$$\sum_{x=0}^{\infty} C_n^{a_1}(x) C_m^{a_1}(x) = \delta_{n,m}; n, m \geq 0. \tag{5}$$

Additionally, it exhibits the following recurrence relation [29]:

$$C_n^{a_1}(x) = AC_{n-1}^{a_1}(x) + BC_{n-2}^{a_1}(x);\ n \geq 2. \tag{6}$$

where

$$\begin{aligned} C_0^{a_1}(x) &= \sqrt{\frac{a_1}{x}} \times C_0^{a_1}(x-1) \text{ with } C_0^{a_1}(0) = \sqrt{e^{-a_1}} \\ C_1^{a_1}(x) &= \sqrt{a_1}\left(1 - \frac{x}{a_1}\right) \times C_0^{a_1}(x) \\ A &= \frac{a_1 + n - x - 1}{a_1}\sqrt{\frac{a_1}{n}} \\ B &= -\sqrt{1 - \frac{1}{n}} \end{aligned} \tag{7}$$

It should be noted that the orthogonality property of Charlier polynomials $C_n^{a_1}(x)$ is maintained when the variable x tends to infinity according to the Eq. (5). We recommend computing Charlier polynomials using the Gram-Schmidt process (GSP) [33] to maintain orthogonality in a finite range.

For an $N \times N$ image $g(x,y)$, the 2-D Charlier transform in terms of Charlier polynomial is defined as [25]:

$$M_{n,m} = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} C_n^{a_1}(x) C_m^{a_1}(y) g(x, y), n, \quad m = 0, 1, \ldots, N-1 \tag{8}$$

The inverse transform resulting from the orthogonality property of Charlier polynomials (Eq.5) is expressed as follows:

$$g(x, y) = \sum_{n=0}^{N-1}\sum_{m=0}^{N-1} M_{n,m} C_n^{a_1}(x) C_m^{a_1}(y) \tag{9}$$

The Charlier transform and its inverse can also be implemented in the matrix form as follows [25]:

$$\boldsymbol{M} = \boldsymbol{C}.\boldsymbol{g}.\boldsymbol{C}^T \text{ and } \boldsymbol{g} = \boldsymbol{C}^T.\boldsymbol{M}.\boldsymbol{C}. \tag{10}$$

where $\boldsymbol{C}^T = \boldsymbol{C}^{-1}$ and $\boldsymbol{C}(n, x) = C_n^{a_1}(x)$.

Despite being widely used, the Charlier transform is not suitable for lossless image applications for two primary reasons. Firstly, it is a lossy, which means some information is lost during the transform process. Secondly, it lacks the integer reversibility property because it produces real floating-point (non-integer) values in the transform domain, and due to finite precision, it is impossible to perfectly recover the original integer values from these real floating-point values during the inverse transform during the inverse transform. As a result, the Charlier transform is not desirable in lossless applications and situations where the use of floating-point operations is not acceptable or suitable, such as in reversible data hiding (RDH) application.

III. PROPOSED INTEGER REVERSIBLE CHARLIER TRANSFORM

The proposed Integer Reversible Charlier Transform (IRCT) is implemented by factorizing an N × N Charlier polynomial matrix (the kernel function of the Charlier transform) into a product of single-row elementary reversible matrices (SERMs), with each SERM being directly integer reversible. This construction ensures that the IRCT is an integer-to-integer mapping, preserving the integer nature of the input signal and guaranteeing lossless transformation. This inherent property of the IRCT makes it highly desirable in numerous applications where the exact recovery of the original input signal from its transform domain coefficients is paramount. The detailed development of IRCT is described as follows.

An $N \times N$ kernel matrix of Charlier transform $\boldsymbol{C}$ has the following properties:

- $\boldsymbol{C}$ is orthogonal, that is $\boldsymbol{C}^T.\boldsymbol{C} = \boldsymbol{C}.\boldsymbol{C}^T = \boldsymbol{I}$ where $\boldsymbol{I}$ is $N \times N$ identity matrix.
- $\boldsymbol{C}$ is invertible and its inverse is equal to its transpose: $\boldsymbol{C}^{-1} = \boldsymbol{C}^T$
- The determinant of $\boldsymbol{C}$ is 1: $\det(\boldsymbol{C}) = 1$.
- All the minors of the leading principal submatrices are 1s.

With these properties of the matrix $\boldsymbol{C}$, it can be factorized into $N + 1$ SERMs as follows [27]:

$$\boldsymbol{C} = \boldsymbol{P}\boldsymbol{S}_N \ldots .\boldsymbol{S}_2\boldsymbol{S}_1\boldsymbol{S}_0. \tag{11}$$

where $\boldsymbol{P}$ is the permutation matrix of $\boldsymbol{C}$, and $\boldsymbol{S}_i (i = 0, 1, 2, \ldots, N)$ are the SERMs defined as

$$\begin{aligned} \boldsymbol{S}_0 &= \boldsymbol{I} + \boldsymbol{e}_N \boldsymbol{s}_0^T \\ \boldsymbol{S}_i &= \boldsymbol{I} + \boldsymbol{e}_i \boldsymbol{s}_i^T,\ i = 1, 2, \ldots, N. \end{aligned} \tag{12}$$

here $\boldsymbol{e}_i$ is the i-th column of identity matrix $\boldsymbol{I}$, and $\boldsymbol{s}_i$ are the vectors constituting the necessary components of the SERMs where the i-th $(i = 1, 2, \ldots, N)$ element must be equal to zero.

The previous SERMs have the following property:

$$\begin{aligned} \boldsymbol{S}_0^{-1} &= \boldsymbol{S}_0 = \boldsymbol{I} + \boldsymbol{e}_N \boldsymbol{s}_0^T \\ \boldsymbol{S}_i^{-1} &= \boldsymbol{I} - \boldsymbol{e}_i \boldsymbol{s}_i^T,\ i = 1, 2, \ldots, N. \end{aligned} \tag{13}$$

The computational process in this study adopts a block computing approach to compute the Charlier transform of the image efficiently. This approach involves dividing the image into non-overlapping blocks, each with a size of 8×8 pixels. By employing this block-based computation, the complexity of image processing operations is significantly reduced. Instead of processing the entire image at once, breaking it down into smaller blocks allows for faster computation. The 8×8 block computing approach is widely used in image processing due to its effectiveness in reducing computational burden. It has been applied in various areas such as image RDH [34],, image watermarking [33], image restoration [35], and image compression [36].

At the bottom of the next page, a standard 8×8 Charlier polynomial matrix $\boldsymbol{C}$ is presented. This matrix can reliably map integers to integers. Notably, it possesses key properties: invertibility, with its inverse equating to its transpose, a determinant value of 1, and all minors of leading principal submatrices being 1s. Consequently, the matrix $\boldsymbol{C}$ conforms to a SERM factorization $\boldsymbol{C} = \boldsymbol{P}\boldsymbol{S}_8 \ldots \boldsymbol{S}_2\boldsymbol{S}_1\boldsymbol{S}_0$, where the permutation matrix is:

$$\boldsymbol{P} = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

and $\boldsymbol{S}_0 = \boldsymbol{I} + \boldsymbol{e}_8\boldsymbol{s}_0^T$, and $\boldsymbol{S}_i = \boldsymbol{I} + \boldsymbol{e}_i\boldsymbol{s}_i^T$ $(i = 1, 2, \ldots, 8)$. The vectors $\boldsymbol{s}_i$ $(i = 0, 1, 2, \ldots, 8)$ to generate SERMs are computed and listed in the second matrix located at the bottom of the next page.

For an integer signal $\boldsymbol{f} = (f_0, f_1, \ldots, f_7)^T$, the 1-D IRCT can be defined as:

$$\boldsymbol{M} = IRCT(\boldsymbol{f}) = \boldsymbol{P}\lfloor \boldsymbol{S}_8 \ldots \lfloor \boldsymbol{S}_1 \lfloor \boldsymbol{S}_0\boldsymbol{f} \rfloor \rfloor \ldots \rfloor \tag{14}$$

where $\lfloor . \rfloor$ is the rounding arithmetic.

The corresponding inverse IRCT (iIRCT) can be expressed as

$$\boldsymbol{f} = iIRCT(\boldsymbol{M}) = \left\lfloor \boldsymbol{S}_0^{-1} \ldots \left\lfloor \boldsymbol{S}_7^{-1} \left\lfloor \boldsymbol{S}_8^{-1}\boldsymbol{P}^T\boldsymbol{M} \right\rfloor \right\rfloor \ldots \right\rfloor. \tag{15}$$

The 1-D IRCT can be extended to the 2-D IRCT. To compute the 2-D IRCT of an 8×8 block $g(x, y)$, the IRCT is first applied to each column of the block and then to each row of the resulting transform. Mathematically, it can be defined as follows:

$$\boldsymbol{M} = IRCT\left((IRCT(\boldsymbol{g}))^T\right). \tag{16}$$

and the 2-D inverse IRCT is

$$\boldsymbol{g} = iIRCT\left((iIRCT(\boldsymbol{M}))^T\right). \tag{17}$$

The present test aims to evaluate the reconstruction capabilities of both the proposed IRCT and the standard Charlier transform. For this purpose, nine grayscale images of size 512×512, from CVG database [37], are used in this test. Evaluation of the reconstructed image quality is conducted using three metrics: PSNR (Eq. 18), MSE (Eq. 19), and SSIM (Eq. 20). In this test, each input image is divided into 8×8 blocks of pixels. Then, each block is transformed by the Charlier transform. Next, the image is reconstructed by applying the inverse Charlier transform on each block. Finally, the reconstruction errors, based on PSNR, MSE, and SSIM are calculated and presented in Table 1. Here, we used the *long* format defined by the IEEE® standard on floating-point arithmetic where floating point numbers have a finite precision of approximately 16 significant decimal points.

From Table 1, it is evident that the proposed IRCT achieves perfect reconstruction, with PSNR, MSE, and SSIM values of $+\text{inf}$, 0, and 1, respectively. These results signify that the reconstructed images are exactly identical to the original images. This is expected because our IRCT is lossless integer transform. Therefore, IRCT is very desirable for RDH application.

On the other hand, the reconstructed images using the standard Charlier transform is not exactly the same as the original images. The loss of information is mainly because the standard Charlier transform is not an integer-reversible transform. Indeed, standard Charlier transform produces real valued coefficients (floating point numbers) in the transform domain, when the inverse transform is applied, these coefficients are rounded according to the precision (finite precision) used in the computation process, which can lead to some loss of information. Figure 1 displays the reconstruction error versus the number of digits rounded after the decimal point. Here, only the PSNR is shown for better visibility. This figure shows that even the reconstructed image quality improves with an increase in the number of digits rounded after the decimal point, there is always a loss of information and, therefore, data hiding using Charlier transform will always result in some loss of data. On the other hand, if we use the proposed IRCT instead of the standard floating-point Charlier transform, lossless data hiding can be achieved. This is a significant advantage. The proposed IRCT will be used as cover transform coefficients for the new image RDH.

IV. REVERSIBLE DATA HIDING USING PROPOSED IRCT

In this section, we propose a novel IRCT-based Reversible Data Hiding (RDH) method designed for secure data transmission in images. Since our IRCT is both lossless and integer reversible, the proposed method ensures that the original cover image can be recovered losslessly from the stego image after extracting the hidden data.

The proposed method adopts the histogram modification technique presented in [20], but with a crucial difference. Instead of modifying the histogram of the cover image directly, we modify the histogram in the IRCT domain. This approach achieves a very high embedding capacity because the IRCT histogram is more concentrated and has higher peaks, as illustrated in Figure 2.

TABLE 1. Reconstruction errors by using the proposed IRCT and classical Charlier transform.

Transform	Metric	Images								
		Mandrill	Lake	House	Fruits	Peppers	Cablecar	Boats	Tiffany	Yacht
IRCT	PSNR(dB)	+Inf	+Inf	+Inf	Inf	+Inf	+Inf	+Inf	+Inf	+Inf
	MSE	0	0	0	0	0	0	0	0	0
	SSIM	1	1	1	1	1	1	1	1	1
Classical Charlier	PSNR(dB)	314.8	314.7	313.0	317.7	315.2	314.5	315.1	311.2	316.0
	MSE ($\times 10^{-27}$)	2.141	2.205	3.191	1.093	1.929	2.265	1.985	4.850	1.623
	SSIM	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.999

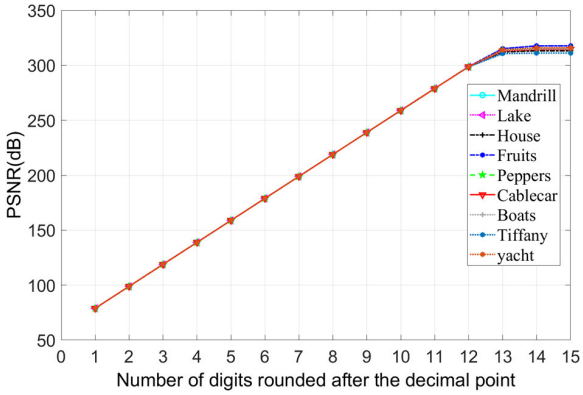

FIGURE 1. Reconstruction error using standard floating-point Charlier transform according to the number of digits rounded after the decimal point.

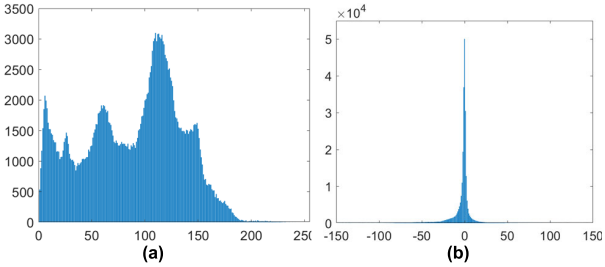

FIGURE 2. (a) Histogram of Fruits image, (b) Histogram of the IRCT matrix of Fruits image.

In the secret message embedding process, the IRCT is first applied to the cover image. Next, an embedding space is created in the resulting IRCT matrix by modifying the IRCT coefficients according to a threshold value T. Specifically, the method uses an integer threshold T (where (T > 0): coefficients lower than -T are decreased by T, and those higher than or equal to T are increased by T. This creates empty spaces in the histogram of the IRCT matrix. The threshold T is chosen based on the required hiding capacity, which increases with higher T values. Subsequently, the secret message is embedded into the IRCT coefficients. To embed a "1" bit, strictly negative IRCT coefficients in the range [−T, T) are reduced by T, and positive coefficients in the same range are increased by T. To embed a "0" bit, the IRCT coefficients within the range [−T, T) remain unaltered. Finally, the stego image is reconstructed by applying the inverse IRCT to the modified IRCT matrix.

For the extraction process on the receiver side, the IRCT is applied to the stego image, allowing the secret message to be extracted and the original cover image to be recovered

$$\boldsymbol{C} = \begin{pmatrix} 0.1389 & 0.2779 & 0.3930 & 0.4538 & 0.4538 & 0.4059 & 0.3314 & 0.2505 \\ 0.3041 & 0.4460 & 0.4013 & 0.1984 & -0.0665 & -0.2965 & -0.4356 & -0.4755 \\ 0.4693 & 0.3969 & 0.0286 & -0.3126 & -0.3889 & -0.1751 & 0.1948 & 0.5513 \\ 0.5442 & 0.0713 & -0.3770 & -0.2871 & 0.1518 & 0.4224 & 0.1937 & -0.4850 \\ 0.4839 & -0.3228 & -0.3061 & 0.2656 & 0.3557 & -0.1753 & -0.4683 & 0.3477 \\ 0.3287 & -0.5066 & 0.1777 & 0.3613 & -0.3022 & -0.2910 & 0.5057 & -0.2037 \\ 0.1640 & -0.4078 & 0.5125 & -0.2308 & -0.2781 & 0.5280 & -0.3549 & 0.0934 \\ 0.0528 & -0.1848 & 0.3920 & -0.5659 & 0.5659 & -0.3796 & 0.1550 & -0.0293 \end{pmatrix}$$

$$\begin{bmatrix} \boldsymbol{s}_0^T \\ \boldsymbol{s}_1^T \\ \boldsymbol{s}_2^T \\ \boldsymbol{s}_3^T \\ \boldsymbol{s}_4^T \\ \boldsymbol{s}_5^T \\ \boldsymbol{s}_6^T \\ \boldsymbol{s}_7^T \\ \boldsymbol{s}_8^T \end{bmatrix} = \begin{pmatrix} 0.93979 & -1.13652 & 1.26507 & -1.71547 & -1.35061 & -1.16296 & -0.06403 & 0 \\ 0 & -0.47989 & 0.23660 & -1.11915 & -0.50328 & -0.14163 & 0.16263 & -0.48500 \\ -0.04883 & 0 & -0.65721 & 0.57844 & 0.33110 & 0.45908 & 0.23801 & 0.52758 \\ 0.76401 & 0.26599 & 0 & 0.06936 & -0.41883 & -0.86521 & -0.65145 & -0.25160 \\ 0.68839 & 0.33029 & -0.68598 & 0 & 0.56818 & -0.41004 & -0.95361 & 0.17332 \\ 0.15309 & -0.11104 & -0.01153 & -0.44260 & 0 & -0.24861 & -0.04903 & 0.25460 \\ -0.19278 & -0.37290 & 0.30744 & 0.14134 & -0.02874 & 0 & -0.02109 & 0.30169 \\ -0.57178 & 0.73076 & 0.40173 & 0.50862 & 0.53502 & 1.13361 & 0 & -0.50573 \\ 0.74761 & -1.83009 & 0.06685 & -0.07982 & -1.36432 & -1.53969 & 1.32207 & 0 \end{pmatrix}$$

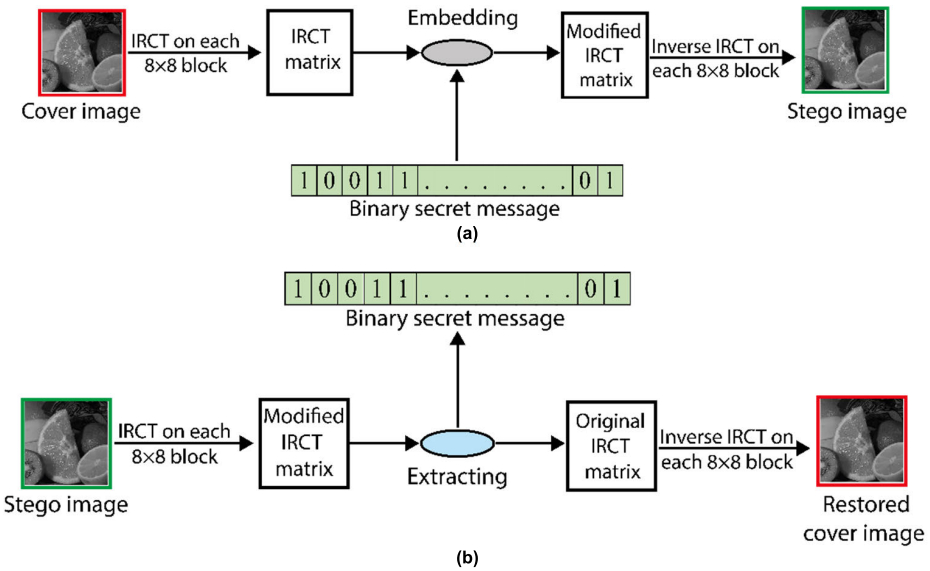

FIGURE 3. Flowchart of the proposed RDH method, (a) secret message embedding process, (b) secret message extraction and cover image recovery process.

by reversing the embedding steps. The following sections provide a detailed explanation of the proposed method.

A. SECRET MESSAGE EMBEDDING PROCESS

This section presents a detailed description of the secret message embedding process. The visual illustration of this process is shown in Figure 3(a).

Let g be a grayscale image of size n × m as cover image and S a binary sequence of length L as secret message. And let T be the threshold that controls the embedding process. The S is embedded into g by applying the following steps:

Step 1: The cover image g is divided into non-overlapping blocks of 8 × 8 pixels after subtracting 128 from each pixel.

This adjustment results in an IRCT histogram more centered on zero in the next step (step 3).

Step 2: Apply the proposed IRCT on each block of the cover image. We refer to the resulting n × m matrix of IRCT coefficients as **M**.

Step 3: Generate the histogram of the IRCT matrix **M**.

Step 4: Set T = 1, and then calculate L_{eff} the effective number of bits that can be embedded which is equal to the number of IRCT coefficients in the range [−T, T).

Step 5: If L_{eff} is equal to or greater than L the required number of bits to be embedded, go to the step 6, otherwise increase the threshold T until the condition is true.

Step 6: Before embedding the secret message into the matrix **M**, an empty embedding space is first created. This space is created by modifying the IRCT coefficients of the matrix **M**. The coefficients less than -T are reduced by T, while the coefficients greater than or equal to T are increased by T. The resulting matrix is denoted **M'**. This step makes the ranges [−2T, -T) and [T, 2T) empty in the matrix **M'**.

Step 7: Embed the secret message S = {s(k), k = 1,2,…,L} by modifying the matrix **M'**. The IRCT coefficients in the interval [−T, T) are modified according to the bit value of the secret message: If s(k) the bit to embedded is a "1", the selected IRCT coefficient is reduced or increased by T if it is strictly negative or positive, respectively. If the bit to be embedded is a "0", the value of the selected IRCT coefficient remains unchanged. The resulting matrix is denoted **M"**.

Step 8: Apply the inverse IRCT on each 8 × 8 block of the matrix **M"** and then add 128 to each pixel to reconstruct the stego image g_s.

The detail description of the proposed embedding process is given in Algorithm 1.

TABLE 2. Embedding capacity, PSNR, and SSIM for threshold T = 1.

Image	*C* (bpp)	PSNR (dB)	SSIM
Mandrill	0.0787	45.8558	0.9961
Lake	0.1139	45.8431	0.9913
House	0.2021	45.8159	0.9897
Fruits	0.3319	45.7895	0.9865
Peppers	0.1473	45.8165	0.9885
Athens	0.1879	45.8240	0.9895
Barnfall	0.1042	45.8493	0.9938
Cablecar	0.2470	45.8038	0.9880
Boats	0.2851	45.7759	0.9880
Tiffany	0.1673	45.8137	0.9862
Yacht	0.2997	45.7978	0.9891
Anhinga	0.1859	45.8781	0.9907
Airplane	0.2064	45.8111	0.9861
Elaine	0.1211	45.8332	0.9906
France	0.3582	45.9005	0.9892
Average	0.2024	45.8272	0.9896

Algorithm 1 Secret Message Embedding Process

Input: Cover image g of size n × m, Binary secret message S of length L.
Output: Stego image g_s

Generate the matrix **M**
Set T = 1;
Calculate L_{eff} the number of IRCT coefficients in the range [−T, T).
while (L_{eff} < L) **do**
  Set T = T + 1;
  Calculate L_{eff} with the new value of T.
end
// Generate the IRCT matrix **M'**
Set **M'**= **M**;
for i = 1 to n **do**
  for j = 1 to m **do**
    if (M(i,j) < -T) **then**
      M'(i,j) = **M**(i,j)-T;
    end
    if (M(i,j) >= T) **then**
      M'(i,j) = **M**(i,j)+T;
    end
  end
end
// Embedding process: Generate the matrix **M"**.
Set **M"** = **M'**;
Set k = 1;
for i = 1 to n **do**
  for j = 1 to m **do**
    if (k < L) **then**
      if (s(k) == 1 && **M'**(i,j) >= -T && **M'**(i,j) < 0) **then**
        M"(i,j) = **M'**(i,j)-T; k ++;
      end
      if (s(k) == 1 && **M'**(i,j) >= 0 && **M'**(i,j) < T) **then**
        M"(i,j) = **M'**(i,j)+T; k++;
      end
      if (s(k) == 0 && **M'**(i,j) >= -T && **M'**(i,j) < T) **then**
      M" (i,j) = **M'**(i,j); k++;
    end
  end
end
end
Reconstruct the stego image g_s from **M"**.
Return g_S.

B. SECRET MESSAGE EXTRACTION AND COVER IMAGE RECOVERY

Figure 3(b) highlights the key steps involved in the process of extracting the secret message and recovering the original cover image from the stego image g_s. Here are the steps to be carried out:

Step 1: Subtract 128 from each pixel in the stego image g_s and divide it into 8 × 8 non-overlapping blocks.

Step 2: Apply IRCT to each block, then the resulting IRCT blocks are combined to obtain the modified matrix **M"**.

Step 3: Extract the secret message S = {s(k), k = 1,2,…,L} from the matrix **M"** as follows: **M"** is scanned from top to bottom and from left to right. Then, if an IRCT coefficient where its value in the range [−T, T) is encountered, a bit "0" is extracted. If an IRCT coefficient where its value in the range [−2T, −T) or [T, 2T) is encountered, a bit "1" is extracted.

Algorithm 2 Secret Message Extraction and Original Cover Image Recovery Process

Input: Stego image g_s of size n× m.
Output: Extracted secret message S of length L, Original cover image g.

Generate the modified matrix **M"**.
/∗ Secret message extraction process: Extract the secret message S = {s(k),k = 1,2,..,L} from **M"**. ∗/
k = 1;
for i = 1 to n **do**
  for j = 1 to m **do**
    if (k <= L) **then**
      if (**M"**(i,j) >= -T && **M"**(i,j) < T) **then**
        s(k) = 0; k ++;
      end
      if (**M"**(i,j) >= -2T && **M"**(i,j) < -T) **then**
        s(k) = 1; k++;
      end
      if (**M"**(i,j) >= T && **M"**(i,j) < 2T) **then**
        s(k) = 1; k++;
      end
    end
  end
end
// Original cover image recovery process: Recover the matrix **M'** from **M"**.
Set **M'** = **M"**;
for i = 1 to n **do**
  for j = 1 to m **do**
    if (**M"**(i,j) < -T) **then**
      M'(i,j) = **M"**(i,j)+ T;
    end
    if (**M"**(i,j) >= T) **then**
      M'(i,j) = **M"**(i,j)-T;
    end
  end
end
Recover the initial cover image g from the matrix **M'**.
Return S and g.

Step 4: After extracting the secret message, we remove the embedding space from the matrix **M"** and recover the matrix **M'** by applying step 6 of the embedding process but in reverse order.

Step 5: Recover the original cover image by applying the inverse IRCT on each 8 × 8 block of the matrix **M'** and then add 128 to each pixel.

The detailed description of the process for extracting the secret message and recovering the original cover image is provided in Algorithm 2.

V. EXPERIMENTAL RESULTS AND COMPARISON

The simulations were performed in Matlab R2022b on a personal computer equipped with an AMD Ryzen 5 5600U CPU, which has 6 cores running at 2.30 GHz, and 8 GB of RAM. In this study, we use different grayscale images from CVG database [37]. It is a very popular and interesting database including different image categories (texture, satellite, medical images, etc.) and often used in the literature for tasks image RDH. A set of fifteen commonly used 512 × 512 images, including Mandrill, Lake, House, Fruits, Peppers, Athens, Barnfall, Cablecar, Boats,

TABLE 3. Embedding capacity (C) and PSNR for thresholds from 1 to 6.

Image	T=1		T=2		T=3		T=4		T=5		T=6	
	C(bpp)	PSNR	*C*(bpp)	PSNR	*C*(bpp)	PSNR	*C*(bpp)	PSNR	*C*(bpp)	PSNR	*C*(bpp)	PSNR
Mandrill	0.0787	45.8492	0.1528	41.6212	0.2225	38.6668	0.2853	36.4884	0.3421	34.7841	0.3925	33.3731
Lake	0.1139	45.8254	0.2186	41.7259	0.3125	38.8571	0.3931	36.7354	0.4616	35.0817	0.5189	33.7102
House	0.2021	45.8153	0.3537	41.9013	0.4515	39.1402	0.5185	37.0304	0.5706	35.3574	0.6106	33.9440
Fruits	0.3319	45.7716	0.5222	42.1672	0.6131	39.4782	0.6649	37.3870	0.7013	35.6845	0.7288	34.2641
Peppers	0.1473	45.8283	0.2846	41.8117	0.4023	39.0369	0.4992	36.9956	0.5749	35.3752	0.6338	34.0424
Athens	0.1879	45.8237	0.3408	41.9080	0.4557	39.1586	0.5407	37.1084	0.6046	35.4619	0.6523	34.0881
Barnfall	0.1042	45.8381	0.2021	41.6899	0.2910	38.8082	0.3691	36.6886	0.4365	35.0148	0.4955	33.6492
Cablecar	0.2470	45.8037	0.4185	42.0214	0.5154	39.2733	0.5713	37.1614	0.6104	35.4522	0.6436	34.0381
Boats	0.2851	45.7833	0.4671	42.0750	0.5631	39.3785	0.6207	37.2756	0.6620	35.6024	0.6939	34.1709
Tiffany	0.1673	45.8163	0.3118	41.8599	0.4236	39.0840	0.5034	37.0120	0.5615	35.3451	0.6064	33.9547
Yacht	0.2997	45.7964	0.4874	42.1242	0.5813	39.4221	0.6352	37.3132	0.6716	35.6142	0.6991	34.1993
Anhinga	0.1859	45.8857	0.2899	41.8481	0.3701	38.9767	0.4298	36.8248	0.4780	35.1139	0.5213	33.7110
Airplane	0.2064	45.8159	0.3693	41.9487	0.4771	39.1983	0.5504	37.1279	0.6039	35.4528	0.6453	34.0562
Elaine	0.1211	45.8363	0.2344	41.7372	0.3355	38.8885	0.4251	36.8090	0.5024	35.1796	0.5675	33.8366
France	0.3582	45.9005	0.5347	42.2284	0.6279	39.5487	0.6704	37.3560	0.7145	35.8213	0.7291	34.3269

Tiffany, Yacht, Anhinga, Airplane, Elaine and France, were selected as cover images. These images were chosen for their widespread use in the field of image processing and RDH research. To assess the effectiveness of our proposed RDH method, a sequence of random bits was employed as the secret message. This choice allows for a rigorous evaluation of the method's performance under controlled conditions. By utilizing a random bit sequence, we can objectively measure the level of embedding and extraction achieved by our method. On each cover image, the secret message is embedded by modifying the IRCT histogram of image using various threshold values ranging from 1 to 10.

Our method is evaluated in terms of two key performance metrics: imperceptibility (stego image quality) and embedding capacity. To assess its effectiveness, the proposed method is compared to existing state-of-the-art RDH methods in multiple domains, including the spatial domain, DCT domain, DWT domain, and integer transform domain (including DCT and integer wavelet transforms).

The imperceptibility is assessed by employing the Peak Signal-to-Noise Ratio (PSNR) (Eq. 18) and the Structural Similarity Index (SSIM) (Eq. 20). These metrics offer quantitative assessments of the distortion introduced during the embedding process between the original cover image and the stego image generated after embedding the secret data. Higher PSNR and SSIM values indicate a better preservation of image quality.

$$\text{PSNR(dB)} = 10\log_{10}\left(\frac{255^2}{MSE}\right). \tag{18}$$

$$\text{MSE} = \frac{1}{M \times N}\sum_{x=1}^{M-1}\sum_{y=1}^{N-1}[g(x,y) - g_s(x,y)]^2. \tag{19}$$

$$\text{SSIM} = \frac{(2\mu_g\mu_{g_s} + c_1)(2\sigma_{g,g_s} + c_2)}{(\mu_g^2 + \mu_{g_s}^2 + c_1)(\sigma_g^2 + \sigma_{g_s}^2 + c_2)}. \tag{20}$$

where g and g_s are the cover image and stego one of size $N \times M$, respectively. σ_{g,g_s} is the covariance between the cover image and the stego image, (μ_g, σ_g^2) are mean and variance of the cover image and $(\mu_{g_s}, \sigma_{g_s}^2)$ are those of the stego image, (c_1, c_2) are constants added to prevent division by zero situations.

The embedding capacity C which represents the information quantity embedded into the cover image is measured in bits per pixel (bpp) and defined as follows:

$$C(bpp) = \frac{\text{Secret message size (bits)}}{\text{Total number of pixels in cover image}}. \tag{21}$$

The embedding capacity depends on the threshold T that controls the embedding process. It increases with the increase of the threshold value T.

Table 2 presents the PSNR, SSIM, and embedding capacity for fifteen stego images using our proposed method. The average values of these metrics are shown in the last row of the table. For this test, we set the threshold T = 1 in the embedding procedure, embedding a number of bits of the secret message equal to the number of IRCT coefficients in the range $[-1, +1)$ for each image. The results indicate that our method achieves high PSNR and SSIM values, greater than 45.7 dB and close to 0.99, respectively, for all images. This demonstrates that our method produces high-quality stego images, closely resembling their corresponding original images. On the other hand, the proposed method provides an average embedding capacity of 0.2024, which is high considering the good quality of the stego images obtained. The embedding capacity varies depending on the cover image

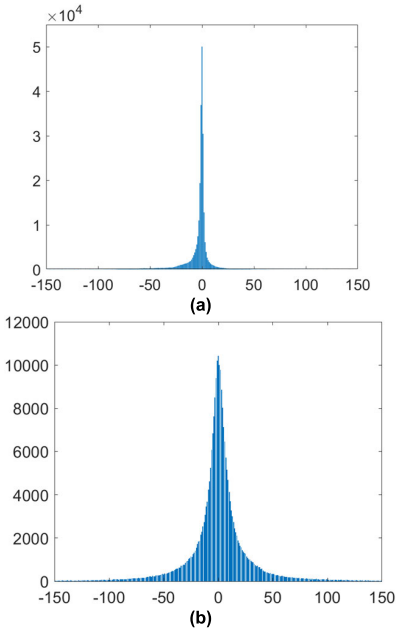

FIGURE 4. (a) IRCT Histogram of Fruits image (b) IRCT Histogram of Mandrill image.

TABLE 4. Average embedding capacity, PSNR and SSIM for thresholds from 1 to 10.

Image	C (bpp)	PSNR (dB)	SSIM
Mandrill	0.3445	35.3853	0.9556
Lake	0.4512	35.6514	0.9109
House	0.5414	35.8363	0.8995
Fruits	0.6666	36.1085	0.8764
Peppers	0.5447	35.8951	0.8886
Athens	0.5698	35.9420	0.8997
Barnfall	0.4320	35.6038	0.9312
Cablecar	0.5802	35.9173	0.8870
Boats	0.6288	36.0302	0.8928
Tiffany	0.5293	35.8321	0.8733
Yacht	0.6357	36.0427	0.8962
Anhinga	0.4720	35.6744	0.9079
Airplane	0.5677	35.9146	0.8683
Elaine	0.4932	35.7672	0.9026
France	0.6756	36.1970	0.8746
Average	0.5422	35.8532	0.8976

used. For example, the Fruits and France images recorded the highest capacity, followed by Yacht, Boats, Cablecar, House, and Airplane images. Other images recorded slightly lower capacities, with the Mandrill image having the lowest embedding capacity due to its reduced number of IRCT coefficients in the range $[-1, +1)$. Figure 4 illustrates the significant difference in embedding capacity between the Fruits and Mandrill images. Overall, the average PSNR, SSIM, and embedding capacity values indicate satisfactory results.

TABLE 5. Difference in the histogram distribution for thresholds from 1 to 10.

Image	Euclidean	Chebychev	Correlation
Mandrill	0.0080	0.0017	0.0096
Lake	0.0129	0.0036	0.0320
House	0.0219	0.0110	0.0503
Fruits	0.0160	0.0127	0.0587
Peppers	0.0087	0.0033	0.0229
Athens	0.1085	0.0634	0.4483
Barnfall	0.0652	0.0216	0.2474
Cablecar	0.0275	0.0150	0.0713
Boats	0.0261	0.0192	0.0766
Tiffany	0.0245	0.0192	0.0497
Yacht	0.0093	0.0029	0.0243
Anhinga	0.1209	0.0911	0.3577
Airplane	0.0196	0.0075	0.0194
Elaine	0.0077	0.0017	0.0122
France	0.0107	0.0045	0.0225
Average	0.0325	0.0186	0.1002

The variation in stego image quality and embedding capacity is more pronounced at lower thresholds and decreases at higher thresholds. For example, in Table 3, increasing the threshold T from 1 to 2 for the Fruits image decreases the PSNR by 3.6044 dB and increases the embedding capacity by 0.1903 bpp. Conversely, increasing the threshold T from 5 to 6 results in a PSNR decrease of 1.4204 dB and an embedding capacity increase of 0.0275 bpp. Overall, lower thresholds significantly impact embedding capacity and stego image quality, while higher thresholds have a lesser effect. Table 3 shows the embedding capacities and PSNRs of the stego images as the threshold T increases progressively from 1 to 6. Table 4 presents the average values of embedding capacity, PSNR, and SSIM for thresholds ranging from 1 to 10. These results demonstrate that both embedding capacity and stego image distortion increase with higher thresholds.

To evaluate the robustness of our IRCT-based RDH method against statistical attacks, we compared the histograms of cover images with those of stego images generated using our method. Figure 5 displays the cover images, their corresponding stego images, and their histograms with the threshold $T = 1$. The histograms of the stego images closely resemble those of the cover images, indicating the effectiveness of our proposed RDH method in terms of robustness against statistical attacks.

To quantify the deviation of the stego image histograms from the cover image histograms, we calculated the distances between the two normalized histograms using three different metrics: Euclidean distance, Chebyshev distance, and Correlation distance. Smaller values of these distances indicate

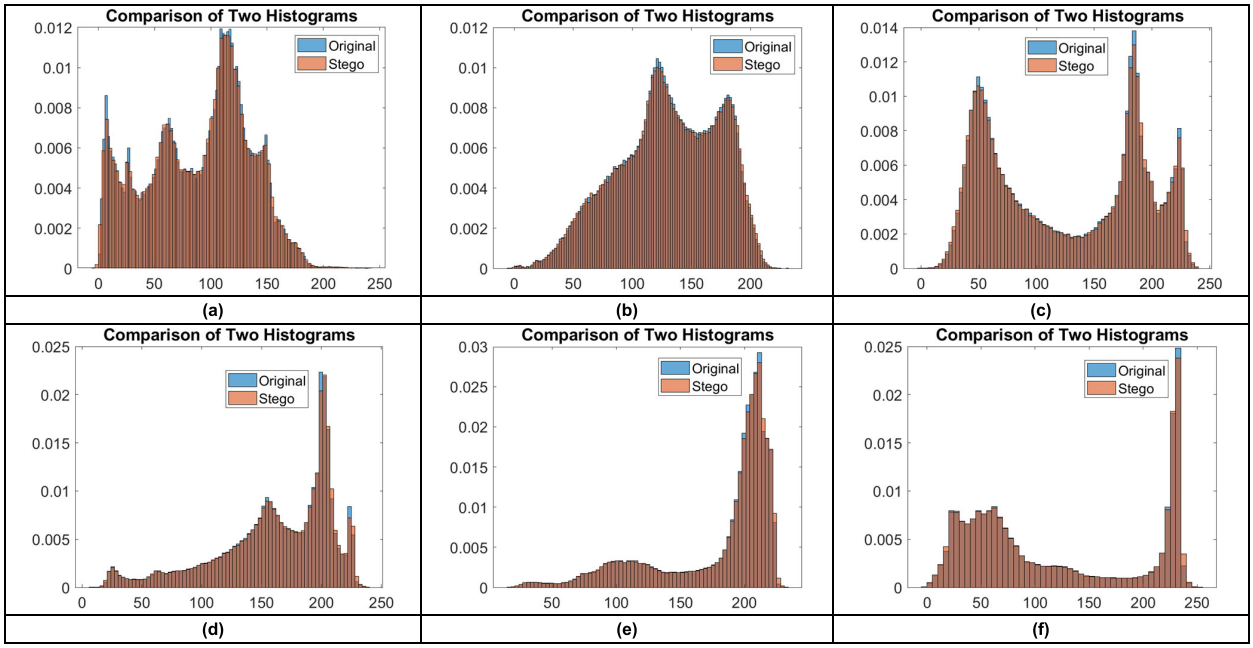

FIGURE 5. Comparison of cover image and stego image histograms, (a) Fruits, (b) Mandrill, (c) Lake, (d) House, (e) Airplane, and (f) Cablecar.

TABLE 6. Comparison of RDH methods based on embedding capacity (C) and PSNR.

Method	France		Mandrill		Airplane		Boat		Average	
	C(bpp)	PSNR	C(bpp)	PSNR	C(bpp)	PSNR	C(bpp)	PSNR	C(bpp)	PSNR
Proposed	0.3582	45.9005	0.0787	45.8551	0.2064	45.8154	0.2851	45.7829	0.2321	45.8385
[20]	0.0821	48.20	0.020	48.20	0.0617	48.3	0.0279	48.20	0.0479	48.2250
[21]	0.0849	48.2	0.0207	48.2	0.0885	48.3	0.0390	48.3	0.0583	48.2500
Haar [12]	0.127	48.44	0.050	48.21	0.218	48.53	0.137	49.75	0.133	48.732
9/7-F [12]	0.165	45.34	0.067	45.46	0.258	45.35	0.137	45.26	0.157	45.35
9/7 [12]	0.165	45.33	0.067	45.45	0.258	45.34	0.136	45.25	0.157	45.35
[24]	0.1940	44.106	0.0950	43.007	0.3605	44.017	0.1893	43.335	0.2097	43.6163

TABLE 7. Performance comparison between the standard DWT [34] and DCT [38] transforms and the proposed IRCT.

Image	[34]		[38]		Proposed	
	C(bpp)	PSNR	C(bpp)	PSNR	C(bpp)	PSNR
France	-----	-----	-----	-----	0.3582	45.9005
Mandrill	0.1350	26.46	-----	-----	0.0787	45.8522
Airplane	0.1404	29.98	0.1505	27.84	0.2064	45.8066
Boat	0.1400	29.75	0.1493	31.41	0.2851	45.7825
Average	0.1385	28.7300	0.1499	29.6250	0.2321	45.8355

minimal deviations between the histograms. Table 5 presents the average distances for fifteen test images using thresholds ranging from T = 1 to T = 10. The results show that the distances (Euclidean, Chebyshev, Correlation) between the histograms of the stego images and the cover images are very small and approach zero. Specifically, the average distances are 0.0325, 0.0186, and 0.1002 for Euclidean, Chebyshev, and Correlation distances, respectively. These results indicate a high similarity between the histograms, demonstrating the robustness of our method against statistical attacks.

To evaluate the performance of our IRCT-based RDH method in terms of embedding capacity, we generated stego images with embedding capacities ranging from 0.1 to 0.8, in increments of 0.1. For each embedding capacity, we adjusted the threshold T until the effective embedding capacity matched the desired embedding capacity. We then calculated the PSNR and SSIM for each stego image. Figure 6 presents the PSNR and SSIM values for fifteen stego images at these different embedding capacities. The figure shows that as the requested embedding capacity increases, the PSNR and SSIM values of the stego images decrease. This is expected because higher embedding capacities require higher threshold values, which introduce more distortion into the stego images. Nevertheless, all stego images exhibit SSIM values greater than 0.9 and PSNR values greater than 35 dB for embedding capacities up to 0.5, demonstrating the effectiveness of our proposed method.

Our IRCT-based RDH method is compared with two spatial-domain image RDH methods reported in [20] and [21], as well as with image RDH methods utilizing

TABLE 8. Comparison of RDH methods based on embedding capacity (C) and SSIM.

Method	France		Mandrill		Airplane		Boat		Average	
	C(bpp)	SSIM	*C*(bpp)	SSIM	*C*(bpp)	SSIM	*C*(bpp)	SSIM	*C*(bpp)	SSIM
Proposed	0.3689	0.9889	0.220	0.9832	0.45	0.9478	0.6200	0.9358	0.4147	0.9639
Haar [12]	0.3450	0.9854	0.2000	0.9765	0.3975	0.8995	0.60	0.90368	0.3856	0.9413
[24]	0.2604	0.9645	0.2204	0.9764	0.3783	0.9664	0.5834	0.9353	0.3606	0.9607
[23]	0.305	0.9864	0.2	0.9981	----	----	----	----	0.25	0.9923

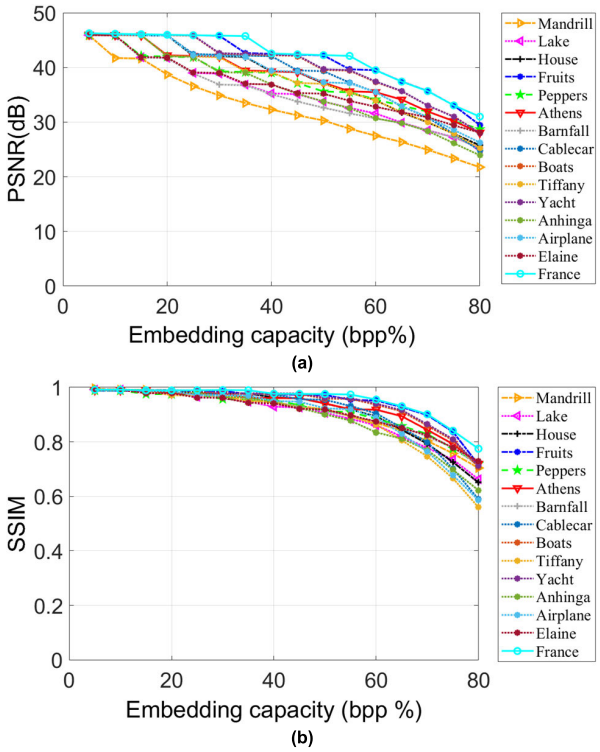

FIGURE 6. Performance evaluation when embedding capacity increases.

integer transforms, including the Haar transform [12], 9/7 transform [12], 9/7-F transform [12], DCT [24], and standard transform-based image RDH methods such as DWT [38] and DCT [34]. For this comparison, we selected standard images, including Mandrill, Airplane, Boat and France, to analyze the performance across different methods.

The results of the comparison between our IRCT-based method and other state-of-the-art methods are presented in Table 6 in terms of PSNR and embedding capacity. The average results are displayed in the last column. The results indicate that our method provides a PSNR that is approximately 2.38 dB lower than the methods operating in the spatial domain [20], [21], but it offers a significantly higher embedding capacity, nearly five times greater. On the other hand, compared to transform domain methods, our method outperforms those based on the 9/7 transform [12], 9/7-F transform [12], and DCT [24] in both PSNR and capacity. Only the method based on the integer Haar transform [12] has a slightly higher PSNR but offers an embedding capacity almost two times lower than our method.

Table 7 presents a comparison in terms of PSNR and embedding capacity between our proposed IRCT-based method and other methods based on standard transforms such as DWT [38] and DCT [34]. The average results are shown in the last row of Table 7. It is evident that our IRCT-based method provides a significantly higher PSNR and high embedding capacity compared to standard DWT and DCT transforms. The limitations of methods [38], and [34] are due to the fact that standard DWT and DCT produce real coefficients, requiring more auxiliary data to support the fractional part, thus reducing the quality of the stego images and the embedding capacity.

The evaluation of our proposed method in terms of SSIM and embedding capacity is presented in Table 8, with average results in the last column. For comparison, RDH methods based on DCT [24], integer Haar wavelet transform [12], and a genetic method [23] are considered. Our IRCT transform offers relatively high embedding capacity and SSIM for all test images compared to the integer Haar and DCT transforms. Additionally, the proposed method provides almost double the embedding capacity of the method in [23], with only slightly lower SSIM.

VI. CONCLUSION

This paper introduces the Integer Reversible Charlier Transform (IRCT) and its innovative application in a reversible data hiding (RDH) scheme for digital images. The IRCT overcomes the inherent limitations of the discrete Charlier moment transform by operating on integer values and producing integer coefficients, allowing for precise and unique recovery of original input values. This results in perfect reconstruction, with exact similarity to the original image.

The proposed IRCT-based RDH method integrates histogram modification in the IRCT domain, leveraging the highly concentrated nature of the IRCT histogram to achieve a very high embedding capacity. Performance evaluations demonstrate that the method offers superior embedding capacity, image quality, and robust resistance to statistical attacks compared to existing techniques in various domains, including spatial and transform domains. These enhancements make it a promising choice for practical implementations in other lossless applications, such as lossless data compression, medical imaging, and secure data transmission.

Despite the success of the proposed method, it also has a drawback. The proposed RDH method and all RDH methods mentioned in this paper are exclusively designed for high-resource computing devices like personal computers (PCs), which typically encompass CPUs, GPUs, and ample memory capacities. This reliance on high-resource environments limits the applicability of these methods in scenarios with resource constraints, such as embedded systems or IoT devices, including microcontrollers. Future work could focus on addressing these limitations by developing a lightweight RDH system that requires low memory demand and reduced execution time, specifically designed for extremely resource-constrained embedded devices.

REFERENCES

[1] R. Kumar, S. Chand, and S. Singh, "A reversible high capacity data hiding scheme using combinatorial strategy," *Int. J. Multimedia Intell. Secur.*, vol. 3, no. 2, p. 146, 2018, doi: 10.1504/ijmis.2018.096356.

[2] Q. Li, X. Wang, B. Ma, X. Wang, C. Wang, Z. Xia, and Y. Shi, "Image steganography based on style transfer and quaternion exponent moments," *Appl. Soft Comput.*, vol. 110, Oct. 2021, Art. no. 107618.

[3] A. Malik, S. Singh, and R. Kumar, "Recovery based high capacity reversible data hiding scheme using even-odd embedding," *Multimedia Tools Appl.*, vol. 77, no. 12, pp. 15803–15827, Jun. 2018, doi: 10.1007/s11042-017-5156-1.

[4] R. Kumar, S. Chand, and S. Singh, "An improved histogram-shifting-imitated reversible data hiding based on HVS characteristics," *Multimedia Tools Appl.*, vol. 77, no. 11, pp. 13445–13457, Jun. 2018, doi: 10.1007/s11042-017-4960-y.

[5] X. Zeng, L. Ping, and Z. Li, "A reversible data hiding scheme using center pixel difference," *J. Multimedia*, vol. 5, no. 4, pp. 763–768, Aug. 2010.

[6] N. Kumar, R. Kumar, and R. Caldelli, "Local moment driven PVO based reversible data hiding," *IEEE Signal Process. Lett.*, vol. 28, pp. 1335–1339, 2021.

[7] X. Wang, X. Wang, B. Ma, Q. Li, C. Wang, and Y. Shi, "High-performance reversible data hiding based on ridge regression prediction algorithm," *Signal Process.*, vol. 204, Mar. 2023, Art. no. 108818.

[8] S. Agrawal and M. Kumar, "Reversible data hiding for medical images using integer-to-integer wavelet transform," in *Proc. IEEE Students' Conf. Electr., Electron. Comput. Sci. (SCEECS)*, Mar. 2016, pp. 1–5.

[9] R. Kumar, S. Chand, and S. Singh, "An optimal high capacity reversible data hiding scheme using move to front coding for LZW codes," *Multimedia Tools Appl.*, vol. 78, no. 16, pp. 22977–23001, Aug. 2019, doi: 10.1007/s11042-019-7640-2.

[10] C. C. Lin and X. L. Liu, "A reversible data hiding scheme for block truncation compressions based on histogram modification," in *Proc. 6th Int. Conf. Genetic Evol. Comput.*, 2012, pp. 157–160.

[11] P. Tsai, "Histogram-based reversible data hiding for vector quantisation-compressed images," *IET Image Process.*, vol. 3, no. 2, pp. 100–114, Apr. 2009.

[12] A. Shaik and V. Thanikaiselvan, "Comparative analysis of integer wavelet transforms in reversible data hiding using threshold based histogram modification," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 33, no. 7, pp. 878–889, Sep. 2021.

[13] N. Muhammad, N. Bibi, Z. Mahmood, and D.-G. Kim, "Blind data hiding technique using the fresnelet transform," *SpringerPlus*, vol. 4, no. 1, pp. 1–15, Dec. 2015.

[14] M. Yamni, H. Karmouni, M. Sayyouri, and H. Qjidaa, "Image watermarking using separable fractional moments of Charlier–Meixner," *J. Franklin Inst.*, vol. 358, no. 4, pp. 2535–2560, Mar. 2021.

[15] C. Deng, X. Gao, X. Li, and D. Tao, "A local tchebichef moments-based robust image watermarking," *Signal Process.*, vol. 89, no. 8, pp. 1531–1539, Aug. 2009.

[16] A. Daoui, M. Yamni, H. Karmouni, M. Sayyouri, H. Qjidaa, M. Ahmad, and A. A. Abd El-Latif, "Color stereo image encryption and local zero-watermarking schemes using octonion Hahn moments and modified Henon map," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 34, no. 10, pp. 8927–8954, Nov. 2022.

[17] M. Yamni, A. Daoui, H. Karmouni, S. Elmalih, A. Ben-fares, M. Sayyouri, H. Qjidaa, M. Maaroufi, B. Alami, and M. O. Jamil, "Copyright protection of multiple CT images using octonion Krawtchouk moments and grey wolf optimizer," *J. Franklin Inst.*, vol. 360, no. 7, pp. 4719–4752, May 2023.

[18] M. Yamni, H. Karmouni, A. Daoui, O. El ogri, M. Sayyouri, and H. Qjidaa, "Blind image zero-watermarking algorithm based on radial Krawtchouk moments and chaotic system," in *Proc. Int. Conf. Intell. Syst. Comput. Vis. (ISCV)*, Jun. 2020, pp. 1–7.

[19] M. Yamni, A. Daoui, and A. A. Abd El-Latif, "Efficient color image steganography based on new adapted chaotic dynamical system with discrete orthogonal moment transforms," *Math. Comput. Simul.*, Feb. 2024, doi: 10.1016/j.matcom.2024.01.023.

[20] Z. Ni, Y.-Q. Shi, N. Ansari, and W. Su, "Reversible data hiding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 3, pp. 354–362, Mar. 2006.

[21] W.-C. Kuo, D.-J. Jiang, and Y.-C. Huang, "A reversible data hiding scheme based on block division," in *Proc. Congr. Image Signal Process.*, vol. 1, May 2008, pp. 365–369.

[22] L. Liu, C.-C. Chang, and A. Wang, "Reversible data hiding scheme based on histogram shifting of n-bit planes," *Multimedia Tools Appl.*, vol. 75, no. 18, pp. 11311–11326, Sep. 2016.

[23] M. Arsalan, S. A. Malik, and A. Khan, "Intelligent reversible watermarking in integer wavelet domain for medical images," *J. Syst. Softw.*, vol. 85, no. 4, pp. 883–894, Apr. 2012.

[24] Y.-K. Lin, "High capacity reversible data hiding scheme based upon discrete cosine transformation," *J. Syst. Softw.*, vol. 85, no. 10, pp. 2395–2404, Oct. 2012.

[25] H. Zhu, M. Liu, H. Shu, H. Zhang, and L. Luo, "General form for obtaining discrete orthogonal moments," *IET Image Process.*, vol. 4, no. 5, p. 335, 2010.

[26] A. Daoui, M. Yamni, O. El Ogri, H. Karmouni, M. Sayyouri, and H. Qjidaa, "Stable computation of higher order charlier moments for signal and image reconstruction," *Inf. Sci.*, vol. 521, pp. 251–276, Jun. 2020.

[27] P. Hao and Q. Shi, "Matrix factorizations for reversible integer mapping," *IEEE Trans. Signal Process.*, vol. 49, no. 10, pp. 2314–2324, Oct. 2001.

[28] S. A. Jerjees, H. J. Mohammed, H. S. Radeaf, B. M. Mahmmod, and S. H. Abdulhussain, "Deep learning-based speech enhancement algorithm using charlier transform," in *Proc. 15th Int. Conf. Develop. ESystems Eng. (DeSE)*, Jan. 2023, pp. 100–105, doi: 10.1109/DeSE58274.2023.10099854.

[29] A. Daoui, H. Karmouni, M. Sayyouri, and H. Qjidaa, "Efficient methods for signal processing using charlier moments and artificial bee colony algorithm," *Circuits, Syst., Signal Process.*, vol. 41, no. 1, pp. 166–195, Jan. 2022, doi: 10.1007/s00034-021-01764-z.

[30] M. Yamni, A. Daoui, O. El Ogri, H. Karmouni, M. Sayyouri, H. Qjidaa, M. Maaroufi, and B. Alami, "Fast and accurate computation of 3D charlier moment invariants for 3D image classification," *Circuits, Syst., Signal Process.*, vol. 40, no. 12, pp. 6193–6223, Dec. 2021.

[31] P. A. Raj, "Charlier and meixner moments and their application for texture and image de-noising problems," in *Information, Photonics and Communication*. Cham, Switzerland: Springer, 2020, pp. 123–134.

[32] K.-L. Xu, Y.-L. Jiang, Z. Li, and L. Li, "Model reduction of discrete time-delay systems based on charlier polynomials and high-order Krylov subspaces," *Linear Algebr. Appl.*, vol. 661, pp. 222–246, Mar. 2023, doi: 10.1016/j.laa.2022.12.020.

[33] M. Yamni, A. Daoui, O. El Ogri, H. Karmouni, M. Sayyouri, H. Qjidaa, and J. Flusser, "Fractional charlier moments for image reconstruction and image watermarking," *Signal Process.*, vol. 171, Jun. 2020, Art. no. 107509, doi: 10.1016/j.sigpro.2020.107509.

[34] C.-C. Chang, C.-C. Lin, C.-S. Tseng, and W.-L. Tai, "Reversible hiding in DCT-based compressed images," *Inf. Sci.*, vol. 177, no. 13, pp. 2768–2786, Jul. 2007.

[35] S. D. Rane, G. Sapiro, and M. Bertalmio, "Structure and texture filling-in of missing image blocks in wireless transmission and compression applications," *IEEE Trans. Image Process.*, vol. 12, no. 3, pp. 296–303, Mar. 2003.

[36] Y. Chen and P. Hao, "Integer reversible transformation to make JPEG lossless," in *Proc. 7th Int. Conf. Signal Process*, 2004, pp. 835–838.

[37] (2023). *Computer Vision Group (UGR)*. [Online]. Available: https://ccia.ugr.es/cvg/index2.php

[38] H.-Y. Huang and S.-H. Chang, "A 9/7 wavelet-based lossless data hiding," in *Proc. IEEE Symp. Comput. Intell. For Multimedia, Signal Vis. Process.*, Apr. 2011, pp. 1–6.