Received August 28, 2020, accepted September 19, 2020, date of publication October 2, 2020, date of current version October 15, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3028315

A Dynamic Three-Bit Image Steganography Algorithm for Medical and e-Healthcare Systems

GHAZANFAR FAROOQ SIDDIQUI1, MUHAMMAD ZAFAR IQBAL1,2, KHALID SALEEM1, (Senior Member, IEEE), ZAFAR SAEED3, ADEEL AHMED1, (Member, IEEE), IBRAHIM A. HAMEED4, (Senior Member, IEEE), AND MUHAMMAD FAHAD KHAN5

1Department of Computer Science, Quaid-i-Azam University, Islamabad 45320, Pakistan
2Department of Computer Science, The Islamia University of Bahawalpur, Bahawalpur 63100, Pakistan
3Faculty of Information Technology, University of Central Punjab, Lahore 54000, Pakistan
4Department of ICT and Natural Sciences, Norwegian University of Science and Technology, 6009 Ålesund, Norway
5Department of Software Engineering, Foundation University Islamabad, Islamabad 44000, Pakistan

Corresponding author: Muhammad Zafar Iqbal (mziqbal@cs.qau.edu.pk)

ABSTRACT Massive advances in internet infrastructure are impacting e-healthcare services compared to conventional means. Therefore, extra care and protection is needed for extremely confidential patient medical records. With this intention, we have proposed an enhanced image steganography method, to improve imperceptibility and data hiding capacity of stego images. The proposed Image Region Decomposition (IRD) method, embeds more secret information with better imperceptibility, in patient’s medical images. The algorithm decomposes the grayscale magnetic resonance imaging (MRI) images into three unique regions: low-intensity, medium-intensity, and high-intensity. Each region is made up of \( k \) number of pixels, and in each pixel we operate the block of \( n \) least significant bits (LSBs), where \( 1 \leq n \leq 3 \). Four classes of MRI images of different dimensions are used for embedding. Data with different volumes are used to test the images for imperceptibility and verified with quality factors. The proposed IRD algorithm is tested for performance, on the set of brain MRI images using peak signal-to-noise ratio (PSNR), mean square error (MSE) and structural similarity (SSIM) index. The results elucidated that the MRI stego image is imperceptible, like the original cover image by adjusting 2\textsuperscript{nd} and 1\textsuperscript{st} LSBs in the low-intensity region. Our proposed steganography technique provides a better average PSNR (49.27), than other similar methods. The empirical results show that the proposed IRD algorithm, significantly improves the imperceptibility and data embedding capacity, compared to the existing state-of-the-art methods.

INDEX TERMS Data payload, image region decomposition, LSB, MRI, spatial domain, steganography.

I. INTRODUCTION

Over the years, the infrastructure of the Internet has expanded significantly from urban to rural areas. Nowadays, images are the main component of multimedia content [1], [2]. With the massive and rapid development of the Internet and network infrastructure, is a common way of using image steganography methods to hide confidential data in different image modalities [3]. A large number of changes in the computing world, including hardware, software, and networks, have created threats to copyright protection and content integrity. Steganography systems are used for invisible communication to embed secret data bits in any communication medium [4]. Information concealment techniques are used to exchange confidential data, withstanding intruders attacks (passive or active). Passive steganalysis exposes the absence / presence of secret data in the stego medium. In contrast, active steganalysis focuses on finding important attributes like original confidential data, data length, location, secret key, and so on [5]. The primitive types of steganography schemes are: spatial domain and transform or frequency domain [6]. In spatial domain schemes, the bits
of the pixel values are directly exploited. The most popular spatial domain steganography schemes are based on the least significant bit substitutions [7]–[15]. In the frequency domain, transformation-based schemes are implemented [4], [16]–[20]. Over the years, the steganography of images has been studied significantly and categorized as reversible and irreversible. The recovery of the hidden data and the restoration of the original image is a focal point of the reversible techniques. At the same time, the irreversible methods mainly focus on the recovery of the hidden data [21]. The telemedicine framework allows healthcare facilities to be available in geographically isolated areas to monitor a patient’s condition remotely [22]. A patient’s medical reports are highly confidential and require special attention when sharing over networks. In e-healthcare systems, the protection of sensitive data requires special attention from a security perspective [23]. Usually, description of images is provided as text. If there is no text report accompanying the image, based on the opinion of the radiologist, the image appears incomprehensible unless some specialists see it [24]. Images can be altered with false information and redistributed to defame a person or organization. Therefore a significant need for content protection. Steganography has become a sufficient solution for such scenarios [25]. In a simple steganography technique, images are more likely to steal confidential information [26]. We have developed an efficient IRD image steganography scheme with better built-in secret data protection. In our steganography technique, sensitive patient diagnostic reports and other secret information are integrated into MRI images with good imperceptibility and high payload capacity. Our embedding procedure (Algorithm 1) embeds data up to 3rd LSB of host images without any clue for the third party on secret information. The embedding procedure is used at the sender side, to hide confidential patient reports, and the reverse procedure is used for extracting secret data from the receiver side. Our goal is to hide a patient’s medical information in MRI images with improved imperceptibility and data payload capacity so that the patient’s medical history is easily accessible to the consultant from MRI images. Mathematically, steganography is defined as:

\[
\text{Stego} = \text{Embed}(c, m, k).
\]

\[
\text{Message} = \text{Ext}(s).
\]

Here, Embed and Ext are the mapping functions for embedding and extracting data in (1) and (2), respectively. Where c is the cover medium, the secret key is denoted by k, and the stego medium is denoted by s with secret data message m.

Our key contributions are as follows:

- Introduced a novel method with significant performance in the context of imperceptibility and payload capacity.
- Carrying out a detailed evaluation and comparison of performance with other similar state-of-the-art procedures.

II. RELATED WORK

Liao et al. [27] used the interblock technique for embedding purposes. JPEG (Joint Photographic Experts Group) images are considered a host or stego image. This technique is specific to medical JPEG images to hide patient records. Adjacent discrete cosine transform (DCT) blocks of similar positions are used to calculate the difference between the coefficients. The work of Sajjad et al. [28] is based on the detection of the region of interest (ROI) and then embedding this ROI to the host image. Some cloud resources are used for encryption of stego images and then transmitted to the receiver over any medium. The receiving side performs the decryption procedure to separate the ROI from the host image and can be used by the concerned consultant. Alsaidi et al. [29] analyzes the use of steganography in computer forensics and explains how criminals can use it to hide evidence. In addition, their research offers study directions for forensic experts. According to Elhoseny et al. [30], nowadays Internet of Things (IoT) devices play an important role in healthcare systems. Level 1 and 2 2D discrete wavelet transform techniques are used to embed patient data in any cover medium. Grayscale and color images are used for cover images. Standard encryption is applied to text data before embedding into the cover media. Various statistical measures are applied to verify the imperceptibility of the cover medium. Statistical scoring works best for secret textual information compared to similar existing techniques [31]. Biometric systems face many security and data integrity challenges. Steganography can play an important role in biometric security. LSB and PVD based steganography methods are widely used to protect biometric data and resist various statistical attacks. Shehab et al. [32] present a delicate watermark technique for self-retrieval and authentication of images in medical applications. A singular value decomposition (SVD) scheme is used on the blocks of the broken image. The SVD block-wise tracks are substituted to the host image LSBs. The technique worked well to recover the original data in case of tampering with the host image. Lee [33] uses the reversible watermark technique on the segmented image, the background region, and the object region. If tampering or forgery has been done to image modalities such as X-rays, computed tomography (CT) or MRI images, the proposed techniques work well to detect the tampering using the hash code. The reversible watermark techniques are particularly effective where medical systems are more vulnerable to forgery or tampering. Kaw et al. [34] offer a method of incorporating data based on optical pixel repetition to integrate patient records into their clinical images. The proposed technique divides the cover image into two by two blocks. Each block contains 16 possible arrangements with four pixel positions.
The electronic patient record is integrated into each block by substituting secret data bits to each block pixel bits. The work of Parah et al. [35] is based on dividing the host image into non-overlapping blocks of $n^{th}$ size. These blocks are based on both non-seed pixels and seed pixels. Only non-seed pixels are used for data embedding to achieve better imperceptibility and payload capacity. The selection of image pixels from the non-sequential least significant bits is based on pixel similarity and fuzzy logic. Pixels with similar intensity values are used to embed secret patient data. The patient’s electroencephalogram (EEG) signal data is used for integration into the MRI host images of the patient [36]. With the increase in popularity of the Internet, people want to share images, videos, documents on the transmission medium. There has been a need to prevent the data from being lost using digital steganography. In addition, information security has a high demand due to the growing concern of the digital market [15]. The imperceptibility and the payload capacity are somehow inversely proportional to each other. If one factor decreases, the other will be increased [37]. The persistence capability is high when the stego media is secured for data elimination and warp attacks. Robustness is the main concern of watermarking algorithms while imperceptibility and storage capacity are major concerns of steganography [2]. The Authors Sahu and Swain have implemented very useful data embedding techniques to improve PSNR and data embedding capability; double layer reversible data embedding method to embed the data in four images [21]. Reversible data embedding method for embedding data in pixels of similar images using LSB match [6]. The right-most n-bit replacement technique uses a pair of similar pixels [3]. The technique of pixel value differencing and modulus function with minimization of the fall of the boundary problem [38]. The rightmost n bits are used for embedding where n is between one and four [39]. The pixel overlap block is based on five pixels from the right, this block is divided into four sub-blocks, 1$^{st}$ and 5$^{th}$, 2$^{nd}$ and 5$^{th}$, 3$^{rd}$ and 5$^{th}$, 4$^{th}$ and 5$^{th}$ [40]. The bit flipping method works on 7$^{th}$ and 8$^{th}$ to hide secret data in cover images [41]. The work of Wazirali and Chachzo [42] divides the regions of the image into non-edge and edge regions. The secret data can only be integrated into the image of the edge region. Zero crossings and log mask with grouping are used to divide the image into edge and not edge regions. Wang et al. [43] in 2018 came up with improvements to the existing distortion feature for jpeg images. The minimization of image distortion is caused by the embedding procedure. A reference image was built before compression which is close to the original host image. Li et al. [44] introduced a technique for embedding data into multiple images known as batch steganography, unlike traditional steganography where only one image is used at a time for embedding purposes. Secret data bits can be retrieved from more than one share, in case of unusual condition in the communication medium during data transmission [44]. Communication channels are widely suitable for compressed jpeg images. Before sending it to the channel, an intermediate image is created, which is close enough to stego image. Tao applied the coefficient adjustment compression scheme in this way so that the original stego image and the compressed image remain similar [45]. Li and Zhang [46] proposed a significant technique for hiding secret data in a fingerprint image, constructed directly from a hidden message. There is no need for a cover signal for embedding purposes, like conventional steganography schemes. The secret message is used as a piece of the hologram to construct the fingerprint image and mapped to the polynomial and encoded at different points of polarities [46].

### III. PROPOSED METHOD

Our proposed algorithm dynamically segments the image into three regions based on intensity and exploits pixel bytes up to 3$^{rd}$ LSB. The three unique regions are low intensity, medium intensity, and high intensity, denoted by $L$, $M$, and $H$ respectively. The threshold value $t_1$ and $t_2$ divides the image into three unique regions. The size of each region can vary dynamically from image to image. In the low intensity (first) region, we exploit pixel bits up to three least significant bits. Secret patient data integrated into third LSB with adjustment of 1$^{st}$ and 2$^{nd}$ LSB, while medium intensity region (second) works with 2$^{nd}$ LSB with the adjustment of 1$^{st}$ LSB. In the high intensity (third) region, only 1$^{st}$ LSB is used for data embedding. These three gray level ranges are used for the incorporation of secret data. Our proposed embedding Algorithm 1 first reads random grayscale values using the pixel index, if the value is in the first region, modify the 3$^{rd}$ LSB and maintain image quality with improvement of 2$^{nd}$ and 1$^{st}$ LSB. In the case of the second region, only 2$^{nd}$ LSB is used with the adjustment of 1$^{st}$ LSB and if the pixel intensity value is in the region 3$^{rd}$, only the first LSB is operated. A secret key is calculated to randomly select the pixel index value before the embedding and extracting procedures. The real number range of 2 to 9 is used to calculate the value of the secret key. If someone intercepts the stego image media LSBs, they will not be able to completely destroy the secret data. At the most, they could attempt to change all (three) LSBs, which will drastically decrease the visual quality and make the stego image noticeable to human eyes. The Figure 1 illustrates the proposed IRD methodology.

### A. EMBEDDING PROCEDURE

The embedding Algorithm 1 first calculates the bytes available in the host image for modification. If the host image capacity bytes available for embedding is less than or equal to the size of SDB, the embedding process will start otherwise an error has occurred. The secret data bits are integrated one by one with the LSBs in the host image. If the pixel intensity range is in the first region, modify 3$^{rd}$ LSB and adjust the 2$^{nd}$ and 1$^{st}$ LSB. If the pixel intensity range is in the second region, then modify 2$^{nd}$ LSB and adjust the 1$^{st}$ LSB, and if the pixel intensity range is in the third region, adjust 1$^{st}$ LSB only. If the gray level range is 0 to 85 then change the third bit with the adjustment of 2$^{nd}$ and a 1$^{st}$ bit.
Algorithm 1: Secret Data Embedding

Input: Cover / host Image, Secret data bits (SDB)
Output: Stego Image

Compute secret data bits size
Check host image pixel intensity value, for example its gVal

repeat
    Acquire next gVal and SDB
    if gVal ∈ FIRST region then
        HOLD ← gVal ∧ 7
        if HOLD ≤ 3 and SDB = 0 then
            gVal ← gVal ∨ HOLD
        else if HOLD ≤ 3 and SDB = 1 then
            gVal ← gVal ∨ 7
        else if HOLD > 3 and SDB = 1 then
            gVal ← gVal ∨ HOLD
        else if HOLD > 3 and SDB = 0 then
            gVal ← gVal − 3
        else
            Continue
    else if gVal ∈ SECOND region then
        HOLD ← gVal ∧ 3
        if HOLD ≤ 1 and SDB = 0 then
            gVal ← gVal ∨ HOLD
        else if HOLD ≤ 1 and SDB = 1 then
            gVal ← gVal ∨ 3
        else if HOLD > 1 and SDB = 1 then
            gVal ← gVal ∨ HOLD
        else if HOLD > 1 and SDB = 0 then
            gVal ← gVal − 2
        else
            Continue
    else
        gVal ← gVal ∧ 254
        gVal ← gVal ∨ SDB

until the embedding of last SDB;

C. EXAMPLE OF PROPOSED IRD ALGORITHM

1) EMBEDDING EXAMPLE

Step 1: Suppose, the randomly selected pixel value of the host image in decimal is 84 (gVal), equal to (01010100)₂ and the embedding bit stream is 01010100. We used threshold t₁ = 86 and t₂ = 171. The selected pixel value (84) belongs to low-intensity (L) region because the selected value is less than threshold t₁.

Step 2: As L intensity region is considered among the three low (L), medium (M), and High (H). now the logical AND operation performed with constant value 7, equal to (111)₂. (01010100)₂ ∧ (111)₂ = (00001000)₂ (Hold) and the first bit of embedding bit stream is 0 (SDB=zero).

Step 3: As per Algorithm 1 (01010100)₂ subtraction (100)₂ = (01010000)₂ perform subtraction with 4. (01010000)₂ Addition (011)₂ = (01010011)₂ perform addition with 3. After SDB substitution the new pixel value (gVal) is (01010011)₂ = 83 in decimal.

Step 4: Suppose, the randomly selected pixel value of host image in decimal is 154 (gVal), equal to (10011010)₂ for the same embedding bit stream. The selected pixel value (154) belongs to Medium-intensity (M) region because the selected value lies between threshold t₁ and t₂. Now the logical AND operation performed with constant value 3, equal to (011)₂. (10011010)₂ ∧ (011)₂ = (10000010)₂ (Hold) and the first bit of embedding bit stream is 0 (SDB=zero).

Step 5: As per Algorithm 1 (100011010)₂ subtraction (010)₂ = (10011000)₂ perform subtraction with 2. (10011000)₂ Addition (001)₂ = (10011001)₂ perform addition with 1. After SDB substitution the new pixel value (gVal) is (10011001)₂ = 153 in decimal.

Step 6: Suppose, the randomly selected pixel value of host image in decimal is 237 (gVal), equal to (11101101)₂ for the same embedding bit stream. The selected pixel value (237) belongs to High-intensity (H) region because the selected value is greater than threshold t₂. As per Algorithm 1 (11101101)₂ ∧ (11111111)₂ = (11101101)₂. next (1110110100)₂ ∨ (SDB) = (111011010)₂. After SDB substitution the new pixel value (gVal) is (111011010)₂ = 236 in decimal.

Step 7: Embedding is done for three cases. The embedding bits are (000)₂.

2) EXTRACTION EXAMPLE

Step 1: Suppose the randomly selected pixel value is (01010011)₂ = 83 (gVal is less than threshold t₁). The bit value of third LSB is extracted, which is 0 now.

Step 2: Suppose the pixel value is (10011001)₂ = 153 in decimal (gVal lies between threshold t₁ and t₂). The bit value of second LSB is extracted, which is 0 now.

Step 3: Suppose the pixel value is (11101100)₂ = 236 in decimal (gVal is greater than threshold t₂). The bit value of first LSB is extracted, which is 0 now.

Step 4: The extracted bits are (000)₂. Extraction is done.

B. EXTRACTION PROCEDURE

The extraction process will begin by reading the grayscale pixel index values from the stego image using the secret key. The Algorithm 2 describes the extraction procedure in detail.
FIGURE 1. Block diagram of the IRD steganographic system.

Algorithm 2 Data Extraction Procedure

Input: Stego Image
Output: Secrete Data

repeat
  if $gVal \in \text{Low-intensity region}$ then
    Read the bit value of third LSB $i.e., t_1$
  else if $gVal \in \text{Medium-intensity region}$ then
    Read the bit value of second LSB $i.e., t_2$
  else
    Read the bit value of first LSB $i.e., t_3$
until all secret data bits are extracted.

D. ERROR METRICS

Two common error metrics, MSE, PSNR [47] and an SSIM quality metric are used to compare the image degradation between the original and stego images. Suppose we have two $m \times n$ image dimension, $x$ and $y$, then MSE, PSNR and SSIM are displayed in (3), (4) and (5), where MAXI is 255 for gray images.

$$MSE = \frac{1}{n \times m} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [x(i, j) - y(i, j)].$$  (3)

$$PSNR = 10 \times \log_{10} \left( \frac{MAXI^2}{MSE} \right).$$  (4)

$$SSIM(x, y) = \left[ l(x, y) \right] \cdot \left[ c(x, y) \right] \cdot \left[ s(x, y) \right].$$  (5)

The SSIM function is based on the following three components: the luminance similarity in (6), the contrast similarity in (7), and the structural similarity (8). These are calculated as follows for the two images $x$ and $y$ [47].

$$l(x, y) = \left( \frac{2\mu_x \mu_y + c_1}{\mu_x^2 + \mu_y^2 + c_1} \right).$$  (6)

$$c(x, y) = \left( \frac{2\sigma_x \sigma_y + c_2}{\sigma_x^2 + \sigma_y^2 + c_2} \right).$$  (7)

$$s(x, y) = \left( \frac{\sigma_{xy} + c_3}{\sigma_x \sigma_y + c_3} \right).$$  (8)

The mean values of the original and processed image are denoted by $\mu_x, \mu_y$, and the standard deviation of the original and processed image is defined by $\sigma_x$ and $\sigma_y$. The co-variance of $x$ and $y$ images is denoted by $\sigma_{xy}$. $c_1, c_2,$ and $c_3$ represent constant values [47].

IV. EXPERIMENTAL SETUP

We have used sample images from a well know data repository\(^1\) to test our method on various images with a variety of dimensions as given in Table 1.

| Image Dimension | No of images | Variable size data for embedding |
|-----------------|--------------|----------------------------------|
| 128 × 128       | 20           | 1KB, 2KB                         |
| 256 × 256       | 20           | 2KB, 4KB, 8KB                    |
| 512 × 512       | 20           | 8KB, 16KB, 32KB                  |
| 1024 × 1024     | 20           | 50KB, 100KB, 130KB               |
| 1024 × 1024     | 20           | 62KB (patient report size)       |

We considered twenty cases for each set of variable dimension for our experiments. The images in Figure 2 from $n1$ to $n10$ are negative or normal images without cancer, and the images from $p1$ to $p10$ are positive or abnormal cases with cancer. The purpose of using variable size embedding data is to test the performance and the real strength of the proposed algorithm.

We tested our proposed algorithm with four image dimensions with ten different payload configurations, as shown in Table 1. We used MSE, PSNR, and SSIM as evaluation matrices for images of various dimensions and the embedding data of different sizes for the performance.

\(^1\)(https://www.kaggle.com/navoneel/brain-mri-images-for-brain-tumor-detection) (accessed on February 24, 2020)
V. RESULTS AND ANALYSIS

We compare our algorithm at 100% payload capacity of host images. Analysis of MSE, PSNR, and SSIM shows that the stego images are highly imperceptible and cannot be discriminated by human eyes. The average PSNR 45.61 and SSIM 0.974 respectively for 1KB payload, while for 2KB payload it is 41.29 and 0.953 as shown in Table 2.

Secret data of various sizes, up to 100% capacity of MRI host images are embedded to test imperceptibility and payload capacity. Images of dimension 256 × 256 are tested on three different payloads i.e., 2KB, 4KB, and 8KB. Table 3 shows the results on MSE, PSNR, and SSIM. The average PSNR value for 2KB, 4KB and 8KB is 45.09 and SSIM is 0.966.

We evaluate the IRD method on a higher dimension, 512 × 512, and embedding data up to 32KB. The results for 8KB, 16KB and 32KB are shown in Table 4. The average PSNR for the given payload is 46.21, and the SSIM is 0.963, respectively.

The payload size increases to 130KB for 1024 × 1024 dimension images. The average PSNR for 50KB, 100KB and 130KB is 45.03 and SSIM 0.974, shown in the Table 5.

---

TABLE 2. Results of stego images of 128 × 128 dimension, having 1KB and 2KB embedding data size.

| No  | MSE 1KB | PSNR 1KB | SSIM 1KB | MSE 2KB | PSNR 2KB | SSIM 2KB |
|-----|---------|----------|----------|---------|----------|----------|
| n1  | 1.88    | 45.38    | 0.991    | 4.92    | 41.20    | 0.990    |
| n2  | 1.64    | 45.96    | 0.977    | 4.72    | 41.39    | 0.955    |
| n3  | 1.31    | 46.93    | 0.995    | 4.02    | 42.08    | 0.994    |
| n4  | 1.63    | 45.98    | 0.99    | 4.72    | 41.38    | 0.985    |
| n5  | 1.47    | 46.41    | 0.992    | 4.55    | 41.73    | 0.990    |
| n6  | 1.62    | 46.00    | 0.987    | 4.73    | 41.37    | 0.987    |
| n7  | 1.92    | 45.28    | 0.965    | 5.31    | 40.87    | 0.916    |
| n8  | 2.20    | 44.69    | 0.957    | 5.75    | 40.52    | 0.912    |

VOLUME 8, 2020
TABLE 5. Results of stego images of 1024 × 1024 dimension, having 50KB, 100KB, and 130KB embedding data size.

|      | MSE  | PSNR | SSIM | MSE  | PSNR | SSIM | MSE  | PSNR | SSIM |
|------|------|------|------|------|------|------|------|------|------|
| No   |      |      |      |      |      |      |      |      |      |
| n1   | 1.40 | 46.66| 0.990| 2.40 | 44.32| 0.983| 3.160| 43.12| 0.987|
| n2   | 0.57 | 47.37| 0.986| 2.07 | 44.95| 0.995| 2.750| 43.72| 0.986|
| n3   | 0.87 | 48.73| 0.996| 1.73 | 45.72| 0.992| 2.230| 44.62| 0.996|
| n4   | 1.10 | 47.70| 0.989| 2.40 | 44.62| 0.986| 3.000| 43.35| 0.987|
| n5   | 0.93 | 48.41| 0.990| 1.99 | 45.13| 0.989| 2.590| 43.98| 0.988|
| n6   | 1.04 | 47.93| 0.990| 2.28 | 44.54| 0.984| 2.800| 43.64| 0.986|
| n7   | 1.47 | 46.43| 0.985| 2.62 | 43.94| 0.949| 3.540| 42.63| 0.968|
| n8   | 1.67 | 45.89| 0.967| 2.79 | 43.66| 0.953| 3.910| 42.20| 0.960|
| n9   | 1.59 | 46.10| 0.959| 3.21 | 43.05| 0.915| 4.060| 42.04| 0.947|
| n10  | 1.56 | 46.19| 0.961| 2.55 | 44.06| 0.949| 3.400| 42.80| 0.963|
| p1   | 1.60 | 46.07| 0.965| 2.82 | 43.62| 0.945| 3.740| 42.39| 0.953|
| p2   | 1.67 | 45.88| 0.963| 2.91 | 43.48| 0.939| 3.700| 42.44| 0.962|
| p3   | 1.51 | 46.33| 0.964| 2.30 | 44.49| 0.951| 3.210| 43.06| 0.966|
| p4   | 1.39 | 46.67| 0.976| 2.47 | 44.19| 0.964| 3.370| 42.85| 0.974|
| p5   | 1.60 | 46.08| 0.969| 2.79 | 43.66| 0.963| 3.820| 42.30| 0.968|
| p6   | 0.92 | 48.46| 0.992| 1.67 | 45.90| 0.985| 2.270| 44.55| 0.988|
| p7   | 0.99 | 48.15| 0.991| 1.60 | 46.08| 0.992| 2.210| 44.67| 0.990|
| p8   | 0.84 | 48.85| 0.991| 1.57 | 46.14| 0.979| 2.070| 44.96| 0.991|
| p9   | 1.36 | 46.76| 0.968| 2.20 | 44.69| 0.962| 2.950| 43.43| 0.971|
| p10  | 1.08 | 47.79| 0.995| 1.75 | 45.69| 0.996| 2.200| 44.69| 0.995|
| Avg  | 1.29 | 47.13| 0.979| 2.31 | 44.60| 0.967| 3.050| 43.37| 0.976|

Figures 3 and 4 respectively visualize the average PSNR and SSIM at maximum payload for four dimensions 128 × 128, 256 × 256, 512 × 512, 1024 × 1024 at maximum payload.

Figures 3 and 4 respectively visualize the average PSNR and SSIM at maximum payload for four dimensions 128 × 128, 256 × 256, 512 × 512, 1024 × 1024 at maximum payload. Each dimension contains twenty images. The maximum average PSNR and SSIM at maximum payload for all four dimension images is 43.40 and 0.955 respectively.

Furthermore, the proposed method is tested on real patient’s data\(^2\) as shown in Figure 5. The results elucidated that the stego images are visually imperceptible to human eyes and almost similar to the original host images, as shown in Figure 6 and maintain a better PSNR value as shown in the Table 6.

\(^2\)Courtesy: Akbar Niazi Teaching Hospital Islamabad https://www.anth.pk (accessed on October 15, 2019)
TABLE 6. Results of stego images of 1024 × 1024 dimension, having 62KB patient report.

| No  | MSE  | PSNR  | SSIM  |
|-----|------|-------|-------|
| n1  | 1.63 | 45.99 | 0.982 |
| n2  | 1.36 | 46.79 | 0.980 |
| n3  | 1.06 | 47.85 | 0.997 |
| n4  | 1.30 | 46.96 | 0.993 |
| n5  | 1.10 | 47.69 | 0.994 |
| n6  | 1.30 | 46.97 | 0.987 |
| n7  | 1.77 | 45.62 | 0.968 |
| n8  | 1.96 | 45.20 | 0.982 |
| n9  | 1.99 | 45.13 | 0.972 |
| n10 | 1.83 | 45.48 | 0.978 |
| p1  | 1.92 | 45.28 | 0.972 |
| p2  | 2.03 | 45.04 | 0.976 |
| p3  | 1.75 | 45.68 | 0.979 |
| p4  | 1.66 | 45.90 | 0.982 |
| p5  | 1.90 | 45.32 | 0.983 |
| p6  | 1.10 | 47.69 | 0.988 |
| p7  | 1.12 | 47.62 | 0.991 |
| p8  | 0.99 | 48.13 | 0.989 |
| p9  | 1.63 | 46.00 | 0.982 |
| p10 | 1.24 | 47.19 | 0.997 |
| Avg | 1.53 | 46.38 | 0.984 |

The Table 6 presents the result of the stego images embedded with patient report. The average PSNR value is 45.94, and the average SSIM is 0.98, respectively.

Figures 7 and 8 show the visual trend of PSNR and SSIM values, respectively. The maximum PSNR and SSIM values are calculated for stego image N3. The average PSNR value is over 45db and SSIM is very close to 1.

![Figure 7](image1.png)

**FIGURE 7.** Visual comparison of stego images PSNR for patient report.

![Figure 8](image2.png)

**FIGURE 8.** Visual comparison of stego images SSIM for patient report.

VI. EVALUATION AND DISCUSSION

The proposed method maintains the visual degradation of stego images to make them imperceptible and better payload capacity in terms of MSE, PSNR, SSIM, and bpp.

![Figure 9](image3.png)

**FIGURE 9.** The set of randomly selected medical images of size 512 × 512 for comparison purpose.

Loan et al. [48] performed experiments on the set of randomly selected medical images as shown in Figure 9, as well as on commonly available standard image processing images. We performed our experiments on a similar set of images for comparison purpose. The set of medical images consist of a standard size 512 × 512.

TABLE 7. Performance comparison of PSNR and SSIM on the set of randomly chosen medical images with the state-of-the-art technique.

| Sr. No | Loan et al. [48] | Proposed method |
|--------|-----------------|-----------------|
|        | PSNR    | SSIM  | PSNR    | SSIM  |
| Image 1 | 43.86   | 0.986 | 46.61   | 0.993 |
| Image 2 | 42.24   | 0.991 | 45.54   | 0.987 |
| Image 3 | 45.14   | 0.990 | 44.76   | 0.989 |
| Image 4 | 44.62   | 0.990 | 44.70   | 0.993 |
| Image 5 | 46.97   | 0.990 | 43.61   | 0.978 |
| Image 6 | 43.58   | 0.981 | 41.33   | 0.844 |
| Image 7 | 44.35   | 0.981 | 41.08   | 0.822 |
| Image 8 | 37.66   | 0.984 | 43.35   | 0.986 |
| Image 9 | 28.03   | 0.959 | 41.02   | 0.797 |
| Image 10| 33.50   | 0.968 | 40.29   | 0.757 |
| Avg    | 41.60   | 0.982 | 43.23   | 0.915 |

The performance comparison of the proposed IRD method with the state-of-the-art technique [48] on the set of randomly selected medical images from UCID dataset is shown in Table 7. The comparison of results with standard set of images that are commonly used in image processing are presented in Table 10 and Table 9. The average PSNR, SSIM, and bpp by [48] is 41.60, 0.982, and 0.816 respectively. Our proposed technique obtained better average PSNR and bpp, which are 43.20 and 1.03 respectively.

Table 8 shows the PSNR results at different threshold values. The threshold value $t_1$ and $t_2$ have a direct impact on the size of the image region. Threshold $t_1$ is significant, because as its value increases, the PSNR decreases because $t_1$ resizes the first region based on low intensity, which exploits the pixel value up to 3rd LSB. The results of our experiment

[^3]: (http://homepages.lboro.ac.uk/cogs/datasets/ucid/data/ucid.v2.tar.gz)
TABLE 8. Comparison of PSNR results at various image region divisions.

| Region Division (%) | Threshold | Len | Baboon | Pepper | Cameraman | Barbara |
|---------------------|-----------|-----|--------|--------|-----------|---------|
| L                   | M         | H   | t₁    | t₂    | PSNR      | PSNR    |
| 45                  | 35        | 20 | 116   | 206   | 44.511    | 44.299  |
| 33                  | 33        | 34 | 86    | 171   | 45.564    | 45.729  |
| 30                  | 30        | 40 | 76    | 152   | 46.149    | 46.340  |
| 25                  | 35        | 40 | 65    | 157   | 45.967    | 46.195  |
| 20                  | 35        | 45 | 51    | 141   | 46.520    | 46.639  |
| Only 3rd LSB        | -         | -  | -     | -     | 42.840    | 42.836  |

TABLE 9. Comparison of results on PSNR with the standard set of images when image dimension is 512 × 512 and embedding bits are 104,857.

| Stego image       | Muhammad et al. [49] | Rehman et al. [50] | Bailey and Curran [51] | Karim et al. [52] | Jassim [53] | Proposed (IRD) |
|-------------------|----------------------|--------------------|------------------------|-------------------|--------------|----------------|
| Lena              | 50.011               | 51.045             | 44.117                 | 42.954            | 44.931       | 49.827         |
| Baboon            | 49.099               | 51.997             | 44.669                 | 44.656            | 44.745       | 50.075         |
| Peppers           | 39.381               | 49.442             | 35.039                 | 31.225            | 34.022       | 50.149         |
| Cameraman         | 48.023               | 50.981             | 44.585                 | 41.559            | 45.213       | 47.884         |
| Barbara           | 47.335               | 50.452             | 46.112                 | 40.993            | 43.595       | 48.421         |
| Average           | 46.769               | 50.783             | 42.904                 | 40.277            | 42.501       | 49.271         |

FIGURE 10. Visual presentation for various values of threshold t₁.

This section is based on the comparison of the performance of the proposed IRD method with five state-of-the-art steganography techniques. We used a standard set of widely used images for comparing the performance of steganography techniques.

The PSNR results of [49]–[53] are based on 104, 857 bits. We used a similar number of bits for the data embedding. The results of the PSNR show that the proposed IRD method significantly outperforms other four baseline methods and remain comparable with Rehman et al. [50]. Table 9 shows the comparison of PNSR based on the size of 104, 857 bits.

Since the embedding of a larger data size with a higher PSNR shows the efficiency of the stego approach, therefore, we further developed the performance of the proposed IRD method with Rehman et al. [50] by increasing the data size to 235, 929 bits. The average PSNRs are 45.870 for the proposed approach and 38.857 for Rehman et al. [50], respectively. The proposed method retains its PSNR and significantly outperforms the baseline approach, as shown in Table 10.

Furthermore, we evaluated the proposed method at its maximum payload capacity (i.e., 263, 016 bits). Table 10 shows the average PSNR value 45.600 at maximum payload which is significant improvement. The stego images are imperceptible because of better PSNR. The average embedding rate in terms of bits per pixel (bpp) is 1.03. The standard size of 512 × 512 dimension image is 262, 144 pixels.

A. COMPUTATIONAL COMPLEXITY

Let N be the number of pixels in the cover image, the embedding algorithm first performs an intensity-based image division into three regions. Therefor N number of passes required for the division mechanism. Thus, the intensity-based image division has a time complexity of O(N). Iterations for each unique region take a constant time O(1). The embedding loop iterates M time where M is the length of the secret data. The total asymptotic time complexity for the data embedding into a source image is O(NM). In the same manner, it can be determined that the time complexity of the extraction algorithm is O(N). To determine the space complexity of the proposed steganography method, the data structures whose size varies with the change of input are taken into consideration. Arrays are used to store the cover image, stego image, and secret data. The gray image with N number of pixel takes N bytes in the memory. Therefore, the space complexity of the proposed algorithm is O(N).

TABLE 10. PSNR comparison with maximum data embedding capacity.

| Image       | Embedding bits 235,929 | Proposed | Embedding bits 263,016 | Proposed |
|-------------|------------------------|----------|------------------------|----------|
| Lena        | 41.035                 | 46.100   | 45.825                 |          |
| Baboon      | 39.728                 | 46.276   | 45.959                 | 46.091   |
| Peppers     | 38.443                 | 46.290   | 45.933                 | 44.923   |
| Cameraman   | 38.059                 | 45.093   | 45.600                 |          |
| Barbara     | 37.022                 | 45.592   | 45.202                 |          |
| Average     | 38.857                 | 45.870   | 45.600                 |          |
VII. CONCLUSION AND FUTURE WORK
In this research work, we proposed a novel IRD algorithm in the image spatial domain to embed variable-sized patient secret data into MRI host images. The algorithm first segments the image into three intensity-based regions. Three least significant bits are operated in low, medium, and high-intensity regions. In the low-intensity area, the substitution of secret data bits is done on 3rd LSB with the enhancement of 2nd and 1st LSB. In the medium intensity region two LSBs are operated, the substitution of secret data bits is done on 2nd LSB, with the adjustment of 1st LSB. In the high-intensity region, only 1st LSB is operated and substituted with secret data bits. The algorithm is tested over a set of MRI images for both positive and negative cases. The results of the proposed IRD methods are significant in terms of imperceptibility and payload capacity. The proposed IRD method is also evaluated over a standard set of images (lena, baboon, peppers, cameraman, barbara) of 512 × 512 dimension. The quality and structural similarity parameters MSE, PSNR, and SSIM verify the image degradation. The MSE and PSNR values always lie within the standard range. In the future, the proposed IRD method could also be extended to other high dimensional image modalities of various parts of the body as well as to color images.

REFERENCES

[1] I. J. Kadhim, P. Premaratne, P. J. Vial, and B. Halloran, “Comprehensive survey of image steganography: Techniques, evaluations, and trends in future research,” Neurocomputing, vol. 335, pp. 299–326, Mar. 2019.
[2] D. Artz, “Digital steganography: Hiding data within data,” IEEE Internet Comput., vol. 5, no. 3, pp. 75–80, May 2001.
[3] A. K. Sahu and G. Swain, “High fidelity based reversible data hiding using modified LSB matching and pixel difference,” J. King Saud Univ.-Comput. Inf. Sci., Jul. 2019, doi: 10.1016/j.jksuci.2019.07.004.
[4] H. Noda, M. Niimi, and E. Kawaguchi, “High-performance JPEG steganography using quantization index modulation in DCT domain,” Pattern Recognit. Lett., vol. 27, no. 5, pp. 455–461, Apr. 2006.
[5] R. Chandramouli, “A mathematical framework for active steganalysis,” Multimedia Syst., vol. 9, no. 3, pp. 303–311, Sep. 2003.
[6] A. Sahu and G. Swain, “Dual stego-imaging based reversible data hiding using improved LSB matching and pixel difference,” Expert Syst. Appl., vol. 37, no. 12, pp. 7703–7710, Dec. 2010.
[7] W.-J. Chen, C.-C. Chang, and T. H. N. Le, “High payload steganography mechanism using hybrid edge detector,” Expert Syst. Appl., vol. 37, no. 4, pp. 3292–3301, Apr. 2010.
[8] A. Ioannidou, S. T. Halkidis, and G. Stephanides, “A novel technique for image steganography based on a high payload method and edge detection,” Expert Syst. Appl., vol. 39, no. 14, pp. 15117–15124, Oct. 2012.
[9] D.-C. Wu and W.-H. Tsai, “A steganographic method for images by pixel-value differencing,” Pattern Recognit. Lett., vol. 24, nos. 9–10, pp. 1613–1626, Jun. 2003.
[10] C.-H. Yang, C.-Y. Weng, S.-J. Wang, and H.-M. Sun, “Adaptive data hiding in edge areas of images with spatial LSB domain systems,” IEEE Trans. Inf. Forensics Security, vol. 3, no. 3, pp. 488–497, Sep. 2008.
[11] M. Naor and A. Shamir, “Visual cryptography II: Improving the contrast via the cover base,” in Proc. Int. Workshop Secur. Protocols, Berlin, Germany: Springer, 1996, pp. 197–202. [Online]. Available: https://link.springer.com/chapter/10.1007/3-540-62494-5_18#citeas
[12] A. Shamir, “How to share a secret,” Commun. ACM, vol. 22, no. 11, pp. 612–613, Nov. 1979.
[13] C.-K. Chan and M. Cheng, “Hiding data in images by simple LSB substitution,” Pattern Recognit., vol. 37, no. 3, pp. 469–474, Mar. 2004.
GHAZANFAR FAROOQ SIDDQUI received the Ph.D. degree from Vrije Universiteit Amsterdam, The Netherlands, in 2010.

He was a Research Scholar with the Department of Computer Science, Vrije Universiteit Amsterdam. He is currently an Assistant Professor with the Department of Computer Science, Quaid-i-Azam University, Islamabad. His Ph.D. Scholarship was funded by the Higher Education Commission, Pakistan. He is also serving as an Evaluator for scientific projects with the Directorate of Science and Technology (DoST). He is also a reviewer for a number of peer-reviewed conferences and journals. He has also published numerous research papers in reputed conferences and journals. He is a member of the Federal Public Service Commission and the Khyber Pakhtunkhwa Public Service Commission. Furthermore, he is also a member of Board of Studies of Quaid-i-Azam University and National Textile University, Faisalabad.

MUHAMMAD ZAFAR IQBAL is currently pursuing the Ph.D. degree with the Department of Computer Science, Quaid-i-Azam University, Islamabad. He is also a Faculty Member of Computer Science with the Islamia University of Bahawalpur. His research interests include image analysis and processing, computer vision, artificial intelligence, and machine learning.

KHALID SALEEM (Senior Member, IEEE) received the M.Sc. degree from Quaid-i-Azam University, Pakistan, in 1994, and the M.Phil. and Ph.D. degrees from Montpellier 2 University, France, in 2005 and 2008, respectively, all in computer science. He was with the software industry from the last 20 years. He is currently an Assistant Professor with Quaid-i-Azam University. He has authored more than 30 research articles and various scientific reports and book chapters.

His research interests include data mining, schema matching and integration, database systems, bioinformatics, distributed systems, and data warehousing.

ZAFAR SAEED received the Ph.D. degree from the Department of Computer Science, Quaid-i-Azam University, Islamabad, Pakistan, in 2020. He was a Research Fellow with the University of Technology Sydney, Australia, from 2017 to 2018. He is currently an Assistant Professor with the University of Central Punjab, Lahore, Pakistan. He has been working on social media network and content analysis and published several articles in high-impact factor journals. His research interests include temporal data streams, social media analytics, and machine learning.

ADEEL AHMED (Member, IEEE) received the M.Phil. degree in computer science from Quaid-i-Azam University, Islamabad, Pakistan, in 2011, where he is currently pursuing the Ph.D. degree with the Department of Computer Science. He is also a Faculty Member of Computer Science with the National University of Modern Languages, Islamabad. He has authored more than ten research articles. His research interests include large scale complex schema matching and integration using machine learning, recommendation systems, social network analysis, and bioinformatics with a focus on simulation techniques. He was granted a Travel Award at the HUGO 15th Human Genome Meeting, Dubai, United Arab Emirates, in 2011.

IBRAHIM A. HAMEED (Senior Member, IEEE) received the dual Ph.D. degrees in industrial systems and information engineering and in mechanical engineering from Korea University, Seoul, South Korea, and Aarhus University, Aarhus, Denmark, respectively. He is currently a Professor with the Department of ICT and Natural Sciences, Faculty of Information Technology and Electrical Engineering, Norwegian University of Science and Technology (NTNU), Norway. He is also the Deputy Head of Research and Innovation with the Department of ICT and Natural Sciences, NTNU. He is also a Program Coordinator with the Department’s International Master Program in Simulation and Visualization. He is the author of more than 120 journal and conference papers. His current research interests include artificial intelligence, machine learning, optimization, and robotics. He is the Elected Chair of the IEEE Computational Intelligence Society (CIS) Norway Section.

MUHAMMAD FAHAD KHAN is currently pursuing the Ph.D. degree with the Department of Computer Science, Quaid-i-Azam University, Islamabad. He is also an Assistant Professor with Foundation University Islamabad. He has authored more than 30 research articles. His research interests include steganography, cryptography, and multimedia communication.

* * *