Ali M. Ahmad, Ghazali Sulong, Amjad Rehman*, Mohammed Hazim Alkawaz and Tanzila Saba

Data Hiding Based on Improved Exploiting Modification Direction Method and Huffman Coding

Abstract: The rapid growth of covert activities via communications network brought about an increasing need to provide an efficient method for data hiding to protect secret information from malicious attacks. One of the options is to combine two approaches, namely steganography and compression. However, its performance heavily relies on three major factors, payload, imperceptibility, and robustness, which are always in trade-offs. Thus, this study aims to hide a large amount of secret message inside a grayscale host image without sacrificing its quality and robustness. To realize the goal, a new two-tier data hiding technique is proposed that integrates an improved exploiting modification direction (EMD) method and Huffman coding. First, a secret message of an arbitrary plain text of characters is compressed and transformed into streams of bits; each character is compressed into a maximum of 5 bits per stream. The stream is then divided into two parts of different sizes of 3 and 2 bits. Subsequently, each part is transformed into its decimal value, which serves as a secret code. Second, a cover image is partitioned into groups of 5 pixels based on the original EMD method. Then, an enhancement is introduced by dividing the group into two parts, namely k_1 and k_2, which consist of 3 and 2 pixels, respectively. Furthermore, several groups are randomly selected for embedding purposes to increase the security. Then, for each selected group, each part is embedded with its corresponding secret code by modifying one grayscale value at most to hide the code in a $(2ki + 1)$-ary notational system. The process is repeated until a stego-image is eventually produced. Finally, the χ^2 test, which is considered one of the most severe attacks, is applied against the stego-image to evaluate the performance of the proposed method in terms of its robustness. The test revealed that the proposed method is more robust than both least significant bit embedding and the original EMD. Additionally, in terms of imperceptibility and capacity, the experimental results have also shown that the proposed method outperformed both the well-known methods, namely original EMD and optimized EMD, with a peak signal-to-noise ratio of 55.92 dB and payload of 52,428 bytes.

Keywords: Information security, steganography, Huffman coding, data hiding, direction evaluation.

DOI 10.1515/jisys-2014-0007
Received January 28, 2014.

1 Introduction

The need for methods that provide efficient work on the protection of data and private property of individuals has become very vital because of the huge growth of multimedia applications on networks. It is therefore

***Corresponding author: Amjad Rehman,** Management Information Systems Department, College of Business Administration, Salman bin Abdul Aziz University, Alkharj, KSA, e-mail: ar.khan@sau.edu.sa
Ali M. Ahmad and Mohammed Hazim Alkawaz: Faculty of Computing, Universiti Teknologi Malaysia, Johor Bahru, Malaysia; and Faculty of Computer Sciences and Mathematics, University of Mosul, Mosul, Iraq
Ghazali Sulong: Faculty of Computing, Universiti Teknologi Malaysia, Johor Bahru, Malaysia
Tanzila Saba: College of Computer and Information Sciences, Prince Sultan University, Riyadh, KSA

important to create methods that provide security for the media from thieves and hackers to prevent them from tampering and misrepresenting the data [8, 9].

Data protection consists of two techniques: cryptography and data hiding. Cryptography means the provision of protection for data storage and data transfer while using a secret key. Encryption is still a successful way to protect stored and transmitted data over a network. However, with the growing use of networks to send and receive data on the global information network, it has become very difficult to maintain these data [12, 14].

Data hiding has two main approaches: steganography and digital watermarking. These two approaches have many techniques [3, 15, 16], one of them is least significant bit (LSB) embedding steganography that alters a document to embed a secret message to reveal the actual ownership [13, 18].

The steganography system consists of two components, the embedder and a detector, and is used to protect secret information by embedding secret messages in the host media (cover image), for instance, text, images, audio, or video.

2 Related Work

The LSB is considered one of the most common methods [1, 2]. This approach deals directly with a cover image, after hiding a secret image within it. In general, bit-mapped images are commonly used. Every image consists of a set of pixels, and every pixel represents one color. The values of a grayscale image range from 0 to 255. If the value of the pixel is equal to 0, it signifies darkness, and when is equal to 255, it indicates lightness. Therefore, a gray-level image can be adjusted by adjusting these values. At least 8 bits are required to represent these values, and the binary system stores them from bits $a_1, a_2, \ldots, a_8$. The LSB substitution changes the last bit (a_1) to make imperceptible modification that cannot be detected by the human vision system. For instance, if the value of a pixel is 100, and we want to embed a 1, the pixel value becomes 101. Human vision cannot recognize this difference. However, the LSB can easily embed secret data into an image with an impalpable effect on the image. The pixel values after embedding can be computed using Eq. (1):

$$x'i = xi - xi \bmod 2k + mi, \tag{1}$$

where xi, and $x'i$ refer to the pixel before and after embedding, respectively; k refers to the number of bits that are going to be embedded; and mi refers to the value of the secret message.

Other researchers used intermediate significant bit planes [4, 5, 7, 10, 17, 19] to overcome the LSB drawbacks. Here, a higher bit plane is selected to embed the secret messages. Thus, it reduces stego-image quality but increases robustness. The higher the bit plane is chosen, the better robustness would be, but the poorer image quality would become and vice versa [11].

Reference [6] proposed the exploiting modification direction (EMD) to reduce distortion. EMD made use of n pixels as a group to embed secret digits in a $(2n + 1)$-ary notational system. During the embedding stage, it requires to increase or decrease one from the value of a particular pixel within the group using Eqs. (2) and (3):

$$f(g_1, g_2, \ldots, g_n) = \sum_{i=1}^{n} (g_i * i) \bmod (2*n + 1), \tag{2}$$

$$s = (d - f) \bmod (2n + 1), \tag{3}$$

where n indicates the number of pixels; $g_1, g_2, \ldots, g_n$ represent values of pixels within each group; s refers to the index of the image; and d indicates the value of the secret digit.

If $f = d$, no change is needed; if $f \neq d$ and $s \leq n$, $g(_s)$ is increased by 1. If $f \neq d$ and $s > n$, $g(_{2n+1-s})$ is decreased by 1. For example, let $n = 4$, $d = 3$, and the original pixel group be [99, 110, 120, 130]. Then, $f = (99*1 + 110*2 + 120*3 + 130*4) \bmod 9 = 2$, $s = (d - f) \bmod 9 = 1$, which is less than n. Thus, $g(_s) = g(_1) + 1 = 100$. Second example: let $d = 9$ and the same pixel group is employed. Then, $s = (d - f) \bmod 9 = 7$, which is bigger than n; thus, $g(_{2n+1-s}) = g(_2) - 1 = 109$. The embedded data can be extracted using the following extraction function of the stego-pixel group (Eq. (4)):

$$d = f(g_1, g_2, \ldots, g_n) = \sum_{i=1}^{n} (g_i * i) \bmod (2*n+1). \tag{4}$$

Ref. [6] proposes the relationship between the value of n and amount of payload that minimizes cover-image distortion. Here, the value of n is computed before embedding using Eq. (5):

$$\left\lfloor \frac{I_s}{n} \right\rfloor \times [\log_2(2n+1)] \geq p, \tag{5}$$

where I_s means the total number of pixels of the cover image, n indicates the amount of pixels for each group, and p represents the payload to be embedded.

3 Proposed Method

The basic proposed idea is to compress the secret message using Huffman coding. Then, the cover image is partitioned into groups, with each group divided into two parts, k_1 and k_2. Also, each secret code is divided into two digits, d_1 and d_2; the first digit (d_1) contains 3 bits and is embedded in the first part (k_1) of the group, whereas the second digit (d_2) contains 2 bits and is embedded in the second part (k_2) of the group.

3.1 Huffman Coding

Initially, secret data are compressed and transformed to bits stream and these bits are transferred into secret digits using the Huffman dictionary table. The Huffman coding can compress each character to a maximum of 5 bits per stream. The stream is divided into two parts, d_1 and d_2, which are 3 and 2 bits, respectively.

Huffman coding algorithm

Step 1: Construct a table containing individual letters and their frequency numbers.
Step 2: Sort the letters in an ascending order according to their frequency numbers.
Step 3: Add the first two frequency numbers and then rearrange the table again.
Step 4: Repeat step 3 until a single frequency number is achieved.
Step 5: Construct a Huffman tree by assigning each pair of branches with (0,1) for all branches of the tree.
Step 6: Rewrite the letters according to the Huffman tree.
Step 7: Construct the final table containing the entire secret letters with their codes.

3.2 An Improved EMD Method

The cover image is partitioned into groups of 5 pixels such that each group is divided into two parts, k_1 and k_2. The first secret digit (d_1) is embedded in the first part (k_1) and the second secret digit (d_2) is embedded in the second part (k_2) of the selected group by modifying one gray-scale value at most to hide the secret digit in $(2k_j + 1)$-ary notational system. In this case, each character can be embedded in one group.

The secret data will be transformed into $(2k_j + 1)$-ary notational system. Eq. (6) is used to transform each k_j-pixel part in the group into a reference value f:

$$f(p_1, p_2, \ldots, p_{kj}) = \left[\sum_{i=1}^{n} (p_i * i) \right] \bmod (2*k_j + 1), \tag{6}$$

where k_j indicates the value of pixels in each part, and $p_1, p_2, \ldots, p_{kj}$ represent the values of the pixels within each part, and $j = 1, 2$; $k_1 = 3$ or $k_2 = 2$.

$$s = (d_j - f) \bmod (2 * k_j + 1), \tag{7}$$

where s refers to the index of the image; d_j indicates the value of the secret digit. Eq. (7) is used to determine which pixel will be changed.

Here, the value of f is collected and compared with the secret digit d_j. If $f = d_j$, no change is needed; if $f \neq d_j$ and $s \leq 5$, g_s is increased by 1. If $f \neq d_j$ and $s > 5$, $g_{(2kj+1-s)}$ is decreased by 1.

The selection of the groups is done at random depending on the secret key known by the sender and receiver. Through the secret key and the random function, a set of random numbers can be generated that indicates the sequence of the groups in the cover image and the same set of the random numbers refer to the same sequence of the groups in the stego-image.

An improved EMD algorithm

Input: cover image of size $M*N$, array of codes from Huffman encoding $d[i]$
Output: stego-image
Step 1: Set $n = 5$, n is the number of pixels for each group
Step 2: Set row=size of the cover image divide by n
Step 3: Set k=length of secret message and i=1
Step 4: Set p_1=3 and p_2=2
Step 5: Construct array g[row][n]
Step 6: while $i \leq k$
Divide d[i] to d_1 with 3 bits and d_2 with 2 bits
r=random number between 1 and row
For j=1: p_1
f=(sum ($g[r][j]*j$)) mod ($2*p_1+1$)
End of loop j
If $f \neq d_1$ then $s=d_2-f$ mod ($2*p_1+1$)
If $s \leq p_1$, then increase $g[r][s]$ by 1
If $s>p_1$, then decrease $g[r][2*p_1+1-s]$ by 1
Set index=1 and z=4
For j=1: p_2
f=(sum ($g[r_z]$*index)) mod ($2*p_2+1$)
index=index+1, $z=z+1$
End of loop j
If $f \neq d_2$ then $s=(d_2-f)$ mod ($2*p_2+1$)
If $s>p_2$ then decrease g[r][$2*p_2+1-s$] by 1
Increase i by 1
End of loop i

3.3 Inversion of an Improved EMD

Once the stego-image is received by the recipient, the inversion of an improved EMD is applied to extract the codes. At first, the stego-image is partitioned into groups of 5 pixels and each group is divided into two parts, k_1 and k_2, that consist of 3 and 2 pixels, respectively. Second, random numbers are generated to select the groups. Subsequently, the first secret digit (d_1) is extracted from the first part and the second secret digit (d_2) is extracted from the second part of the selected group using Eq. (8). Next, the secret digits are converted into binary numbers and they are combined to create a new stream of the binary, then converted again to the decimal number to obtain the secret code. Finally, the secret codes are decompressed using the Huffman decoding to get the original message of plain text.

$$d_i = f(p'_1, p'_2, \ldots, p'_{ki}) = \left[\sum_{i=1}^{ki}(p'_i * i)\right] \bmod(2 * k_i + 1), \tag{8}$$

where $p'_1, p'_2, \ldots, p'_{ki}$ mean the pixels of the stego-image; k_i indicates the amount of pixels for each part in the group; and d_i is a secret code.

Inversion of an improved EMD algorithm

Input: Stego-image
Output: array of the secret digits
Step 1: set row=size of the stego-image divided by n
Step 2: segment the stego-image into row*n groups, by constructing g'[row][n]
Step 3: set L=1, length=size of the secret message
Step 4: set k_1=3 and k_2=2
Step 5: while $i \leq$length
  r=random number between 1 and row
For j=1: k_1
d_1=(sum ($g'[_r][_j]$*j)) mod (2*k_1+1)
End of loop j
    Convert d_1 to binary
    Index=1
    Start=4
    Last=5
    For j=start: last
d_2=(sum ($g'[_r][_j]$*index)) mod (2*k_2+1)
      Index=index+1
End of loop j
Convert d_2 to binary
d=combine between (d_1 and d_2)
    Increase L by 1
    If L>length
      Then go to step 6
      Else
      $Z[L]$=convert d to the decimal
      Increase i by 1
    End if
  End of loop i
Step 6: print the secret codes Z[]

4 Experimental Results

An experiment is conducted using four standard gray-scale images of 512 × 512 pixels as cover images shown in Figure 1, namely, Baboon, Airplane, Lena, and Tiffany. Furthermore, an arbitrary plain text that contains all letters of the alphabet is used as the secret message. To evaluate the quality of the stego-image, the following peak signal-to-noise ratio (PSNR) is employed:

$$\text{PSNR} = 10 \log_{10} \frac{255^2}{\text{MSE}}, \tag{9}$$

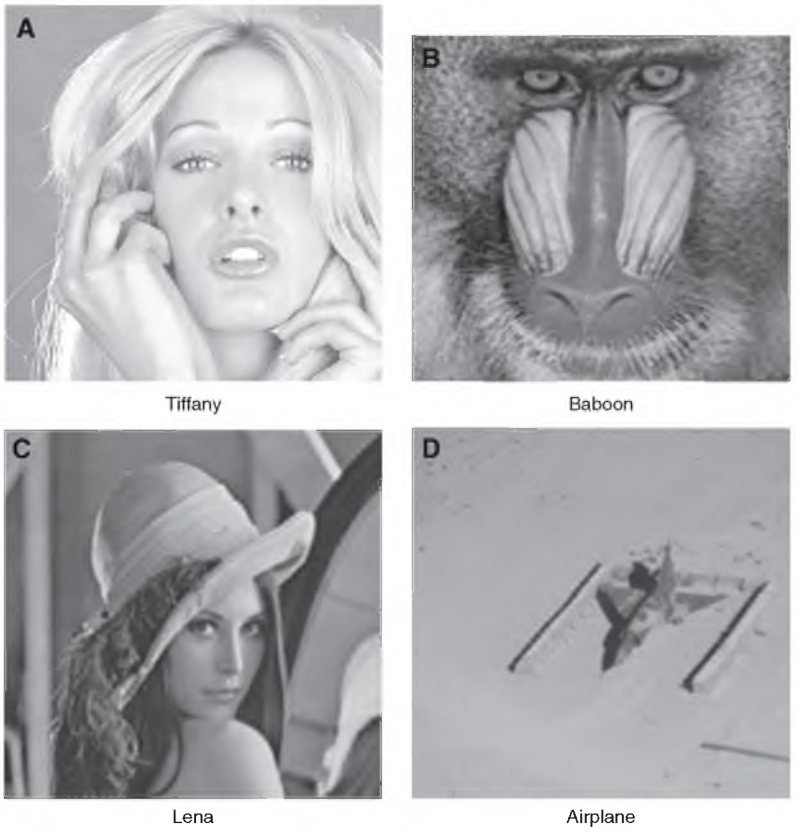

Figure 1. Test Images (A)–(D).

where MSE refers to the mean square error between two pixels.

$$\mathrm{MSE} = \frac{\sum_{i=1}^{r}\sum_{j=1}^{c}([\mathrm{cover}(i,j) - \mathrm{stego}(i,j)])^2}{r \times c}, \tag{10}$$

where r and c refer to the number of rows and columns, respectively, for the cover image and stego-image.

4.1 Imperceptibility

For the purpose of a validation test between the proposed technique and various well-known methods, namely EMD and optimized EMD, same sizes of payload are used.

As shown in Table 1, the experimental results are divided into four parts in different sizes of payload that are computed by bit per pixel. The embedding rate starts from the first part that utilizes full payload where the embedding rate equals 1.599 bits per pixel (bpp) (52,400 bytes) and ends with the fourth part with 0.5 bpp (16,384 bytes). The results of the first part showed that the proposed method is superior to the rest of the techniques in terms of both PSNR and payload viz. 55.93 dB and 1.599 bpp, respectively. However, this ratio (1.599 bpp) is overflow for both the original EMD and optimized EMD because the maximum payload that can be embedded for these two methods is 1.5 bpp. As shown for the second, third, and fourth part of the results, it is obvious that the performance of the proposed method is maintained better in terms of PSNR with the same embedding rate, where the difference of PSNR is quite significant between the proposed method and the second best method, which is around 4 dB when the embedding rate equals 1.5 bpp (49,152 bytes), and the difference of PSNR is 3 and 5 dB when the embedding rate equals 1.0 bpp (32,768 bytes) and 0.5 bpp (16,384 bytes), respectively. As a result, the proposed method produces high performance capable of embedding large amounts of secret messages without sacrificing image quality.

Table 1. Performance of the Proposed Method against both EMD and Optimized EMD.

Bpp	Method	Lena	Airplane	Tiffany	Baboon	AVG
1.599	Proposed method n = 5	55.92	55.95	55.94	55.92	55.92
	Opt EMD n = 2	Overflow				
	EMD n = 2	Overflow				
1.5	Proposed method n = 5	56.21	56.25	56.19	56.22	56.21
	Opt EMD n = 2	52.11	52.10	52.11	52.11	52.11
	EMD n = 2	52.11	52.10	52.11	52.11	52.11
1.0	Proposed method n = 5	57.99	57.02	57.98	57.99	57.99
	Opt EMD n = 4	54.67	54.67	54.66	54.66	54.66
	EMD n = 2	53.86	53.87	53.86	53.87	53.87
0.5	Proposed method n = 5	60.97	60.98	61.06	61.05	61.01
	Opt EMD n = 9	58.37	58.36	58.36	58.38	58.37
	EMD n = 2	56.88	56.89	56.88	56.89	58.89

4.2 Robustness

The evaluation of robustness is done by applying the χ^2 attack in Figure 2. To prove that the proposed method provides high robustness, the results should be compared with another method in the same payload and image. Therefore, Figures 3–6 are employed to benchmark the results of robustness.

Figure 2. Man Image.

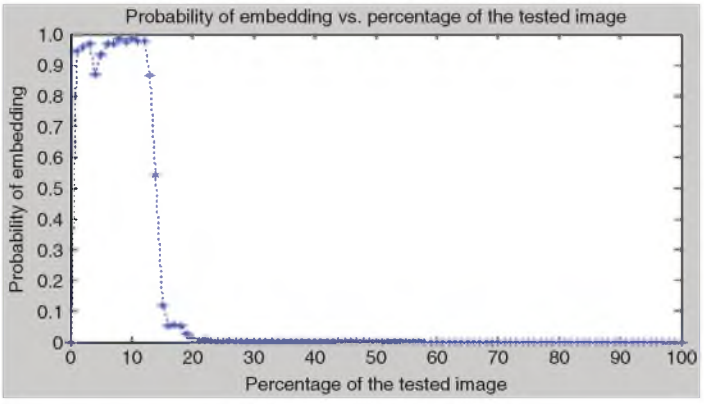

Figure 3. Results for χ^2 Attack on Original Man Image.

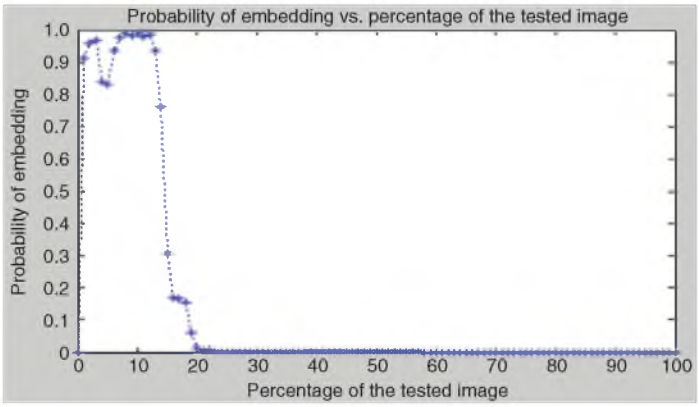

Figure 4. Results for χ^2 Attack on Man Stego-image That Has Been Embedded by the Proposed Method Using 5-Pixel Group and Tested on 100,000 Bits of the Secret Message.

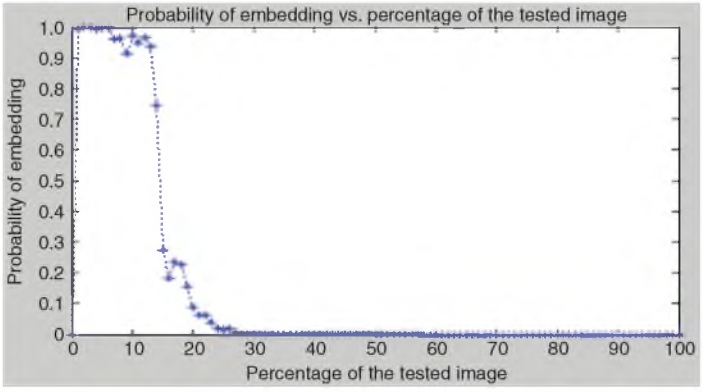

Figure 5. Results for χ^2 Attack on Man Stego-image That Has Been Embedded by the Original EMD Method Using 2-Pixel Group and Tested on 100,000 Bits of the Secret Message.

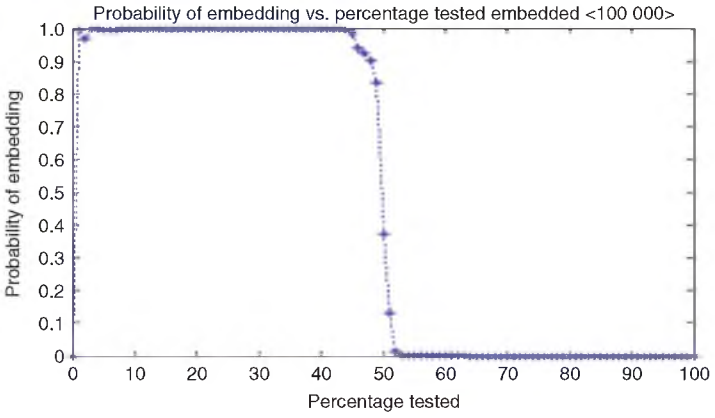

Figure 6. Results for χ^2 Attack on Man Stego-image That Has Been Embedded by Simple LSB Method and Tested on 100,000 Bits of the Secret Message.

The proposed method is compared with the original EMD and LSB method using the same cover image and payload. As shown in Figures 4–6, the probability of containing a hidden message in the proposed method as well as in original EMD are almost zero except for the first 15% of the test where the probability varies between 1 and 0.8; during this 15%, the results revealed that the proposed method is similar to the original image (Figure 3) rather than to the original EMD. From this fact, it can be deduced that the proposed

method is a successful method for embedding secret messages because it does not leave clear-cut effective when a hidden message is embedded inside the stego-image. Furthermore, the probability of the LSB method is almost continuously 1 until 50% of the tested image, and this result is very different from the original image in Figure 3; therefore, the attacker can easily detect that the image contains hidden information. Thus, as can be clearly seen, the proposed method in Figure 4 and the original EMD in Figure 5 outperformed the LSB method in Figure 6.

5 Conclusion

In this article, an improved EMD method and Huffman coding is proposed. The secret messages are encrypted, compressed, and embedded in random groups inside cover images, with high secrecy but without compromising the quality of the stego-images. Experimental results have revealed that the proposed method has successfully outweighed the well-known methods in PSNR, payload, and robustness.

Bibliography

[1] E. Adelson, Digital signal encoding and decoding apparatus, United States Patent, 4 (939, 515), pp. 1–13, 1990.
[2] W. Bender, D. Gruhl, M. Morimoto and A. Lu, Techniques for data hiding, *IBM Systems Journal* **35** (1996), 313–336.
[3] M. Elarbi-Boudihir, A Rehman and T Saba, Video motion perception using optimized Gabor filter, *International Journal of Physical Sciences* **6** (2011), 2799–2806.
[4] M. Emami and G. B. Sulong, A statistical method based on L2Norm technique for EISB information watermarking scheme, in: *Proc. of International Conference on Future Information Technology IPCSIT*, vol. 13, pp. 139–143, 2011.
[5] A. Habes, Information hiding in BMP image implementation, analysis and evaluation, *Information Transmissions in Computer Networks* **6** (2006), 1–10.
[6] L. Kai Yung, H. Wien, J. Chen, C. Tung Shou and C. Wen Chin, Data hiding by exploiting modification direction technique using optimal pixel grouping, in: *Proceedings of the 2010 Education Technology and Computer (ICETC)*, 2010 2nd International Conference on 22–24 June 2010, V3-121–V123-123.
[7] B. A. Mehemed, T. E. A. El-Tobely, M. M. Fahmy, M. E. L. Said Nasr and M. H. A. El-Aziz, Robust digital watermarking based falling-off boundary in corners board-MSB-6 gray scale images, *International Journal of Computer Science and Network Security* **9** (2009), 227–240.
[8] Z. F. Muhsin, A. Rehman, A. Altameem, T. Saba and M. Uddin, Improved quadtree image segmentation approach to region information, *The Imaging Science Journal* **62** (2014), 56–62.
[9] K. Neamah, D. Mohamad, T. Saba and A. Rehman, Discriminative features mining for offline handwritten signature verification, *3D Research* **5** (2014), DOI 10.1007/s13319-013-0002-3.
[10] S. M. Perumal and V. V. Kumar, A wavelet based digital watermarking method using thresholds on intermediate bit values, *International Journal of Computer Applications* **15** (2011), 29–36.
[11] K. Rabah, Steganography: the art of hiding data, *Information Technology Journal* **3** (2004), 245–269.
[12] A. Rehman and T. Saba. Neural network for document image preprocessing: state of the art, *Artificial Intelligence Review* (2012), DOI: 10.1007/s10462-012-9337-z.
[13] A. Rehman and T. Saba, Features extraction for soccer video semantic analysis: current achievements and remaining issues, *Artificial Intelligence Review* **41** (2014), 451–461, DOI: 10.1007/s10462-012-9319-1.
[14] A. Rehman, S. Alqahtani, A. Altameem and T. Saba, Virtual machine security challenges: case studies, *International Journal of Machine Learning and Cybernetics* (2013), DOI 10.1007/s13042-013-0166-4.
[15] T. Saba and A. Rehman, Effects of artificially intelligent tools on pattern recognition, *International Journal of Machine Learning and Cybernetics* **4** (2012), 155–162, DOI: 10.1007/s13042-012-0082-z.
[16] Z. Xinpeng and W. Shuozhong, Efficient steganographic embedding by exploiting modification direction, *Communications Letters, IEEE* **10** (2006), 781–783.
[17] D. Yan, R. Yang, Y. Yu and H. Xin, Blind digital image watermarking technique based on intermediate significant bit and discrete wavelet transform, in: *Proceedings of the 2009 Computational Intelligence and Software Engineering*, 2009, CISE 2009, International Conference on 11–13 December 2009, pp. 1–4.
[18] A. A. Zaidan, B. B. Zaidan, Y. A. Taqa, M. K. Sami, G. M. Alam and A. H. Jalab, Novel multi-cover steganography using remote sensing image and general recursion neural cryptosystem, *International Journal of Physical Sciences* **5** (2010), 1776–1786.
[19] A. M. Zeki and A. A. Manaf, A novel digital watermarking technique based on ISB (intermediate significant bit), *International Journal of Information Technology* **5** (2009), 989–996.