A Novel JPEG Steganography Method Based on Modulus Function with Histogram Analysis

Vladimír BÁNOCI[1], *Gabriel BUGÁR*[1], *Dušan LEVICKÝ* [1], *Zita KLENOVIČOVÁ*[1]

[1]Dept. of Electronics and Multimedia Communications, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Letná 9, 042 00 Košice, Slovak Republic

vladimir.banoci@tuke.sk, gabriel.bugar@tuke.sk, dusan.levicky@tuke.sk, zita.klenovicova@tuke.sk

Abstract. *In this paper, we present a novel steganographic method for embedding of a secret data in still grayscale JPEG image. In order to provide large capacity of pro–posed method while maintaining good visual quality of stego-image, the embedding process is performed in quan–tized transform coefficients of Discrete Cosine transform (DCT) by modifying coefficients according to modulo function, what gives to the steganography system blind extraction predisposition. After-embedding histograms of proposed Modulo Histogram Fitting (MHF) method is analysed to secure steganography system against stegana–lysis attacks. In addition, AES ciphering was implemented to increase security and improve histogram after-embed–ding characteristics of proposed steganography system as experimental results show.*

Keywords

2D-DCT, modulo function, JPEG, steganography, histogram, AES encryption.

1. Introduction

Information hiding, steganography, and watermarking are three closely related fields that have a great deal of overlap and share many technical approaches. However, there are fundamental philosophical differences that affect the requirements, and thus the design of a technical solution. Digital watermarking is mainly used in copyright protection; steganography is a method of embedding the secret message into a camouflage media to ensure that unintended recipients will not be aware of the existence of the embedded secret data in cover media [6]. Additionally, steganography is different from cryptography as a part of secret communication based on cryptography techniques may fail since a cipher text has meaningless form and thus easily arouses the curiosity of malicious attackers who are willing to consume the substantial amount of time and energy to recover or destroy data. Unlike cryptography, steganography conceals the fact that there is secret communication going on and still image may be represented as well suited camouflage media for embedding. Even more, the advantage of cryptography techniques could be implemented in steganography systems to be executed on secret data prior embedding into still image; to strengthen security level and also to suppress the energy disproportion of secret data.

Steganalysis on the other side is the science of detecting hidden information. The main objective of steganalysis is to break steganography system and that condition is met if steganalysis algorithm can judge whether a given image contains a secret message.

There are three main types of steganalysis. Firstly, *visual attacks* try to reveal the presence of hidden information through inspection with naked eye or with the assistance of a computer, which can separate the image into bit planes for further analysis. Secondly, *statistical attacks* are more powerful and successful, because they reveal the smallest alterations in an image´s statistical behaviour. These attacks can be further classified as (i) passive attacks and (ii) active attacks. Passive attacks deal with identifying the presence or absence of a covert message or the embedding algorithm used etc., whereas the goal of active attacks is to estimate the embedded message length or location of the hidden message or the secret key used in embedding. Thirdly, *structural attacks* are based on fact that format of the data files often changes as the data to be hidden are embedded; on identifying these characteristic structure changes can detect the existence of a secret data. Generally, the modification of redundant bits changes the statistical properties of the cover-image what could reveal the hidden message [8].

The JPEG compression uses discrete cosines transform (DCT) which reduces the visual redundancy in order to achieve good compression performance. Even though the more advanced standard JPEG2000 has been introduced, JPEG images are still the most common image format for web and local usage. Therefore JPEG images are also widely used as cover-images in stenoganography. Due to less visual redundancy, the embedding capacity is relatively smaller comparing to steganographic methods in other uncompressed image formats like BMP format [1], [2]. There are two requirements which need to be fulfilled in design of steganography system in JPEG. Firstly, a secret message embedded in cover-image should be visually imperceptible along with solid transfer capacity that generally depends on quality factor of JPEG image file which gives proportional visual redundancy for embedding. The

second one expects that receiver can fully recover a secret message without the knowledge of cover-image.

In this paper, authors focus on the JPEG steganography, where several methods already have been presented. Those methods can be distinguished according to their embedding approach in two categories. One group changes quantization table of JPEG file in order to embed a secret data e. g. JQTM method [5], and another group manipulates directly transform coefficients of DCT such as JPHS [10], J-Steg [12], [4] that embeds the secret message by sequentially flipping the least significant bit (LSB) of the quantized DCT coefficients (except 0's and 1's) without causing detectable artificial distortion. There are several other methods as F5 [11] that decreases coefficients' absolute value by 1 and randomly chooses quantized coefficients for embedding instead of flipping LSB. The Outguess method (and Outguess 0.2) [9] embeds a secret data in similar way as J-Steg, however it uses just a part of usable coefficients. Outguess method after embedding matches global after-embedding histogram of DCT coefficients to a histogram of the cover-image.

However, the modern steganalysis examines statistical properties of stego-image to detect the presence of a secret message. Therefore, many present steganographic techniques seem to be insecure once the modern steganalysis is employed to test their security. To those general steganalysis methods can be included by χ^2-attack and histogram analysis of stego-image and cover-image DCT coeficients. Methods J-Steg and JHPS are detectable by general χ^2-attack which can reliably determine LSB placement. F5 algorithm successfully defends against both the chi-square and extended chi-square attacks. However, Fridrich et. al [7] presented a specific technique of analysing DCT coefficient histogram that can detect F5.

In this paper we propose a novel JPEG steganography method secure against histogram-based attacks, which allows hiding a secret data in still greyscale images while maintaining good visual quality and high transfer capacity. Another advantage of this method is full reconstruction of secret data without having original image present on the recipient side due to modulo embedding concept. The organization of the paper is as follows. In Section 2, authors discussed the concept of JPEG steganography, histogram analysis and modulo function which was used for embedding a secret message. Section 3 describes the proposed method's algorithm, where experimental results with comparison are given in Section 4. In Section 5 the discussion and contribution of the proposed method to the image steganography is given.

2. Modulus Steganography Framework

The secret message is embedded to DCT domain that is represented by 2D-DCT transform applied to cover-image in 8x8 pixel processing. Each transformed block with DCT coefficient $C^B_{m,n}; m,n \in <1,8>$ is rearranged by well known "Zig-zag" algorithm that aligns coefficients in frequency ascending order starting with DC (AC_1) value and it continues with higher frequency components, where B represents the block from all N blocks. Thus, each aligned DCT coefficient from blocks N with the same frequency belongs to one of frequency components $AC_k; k \in <1,64>$. The second phase consists of modulus function calculation for each transform coefficient from frequency component AC_k according to the following formula $r_i = C_i^k \bmod n$, where $n = \{2^s; s \in N\}$, $i = <1, N>$. Sets of remainders r_i altogether with operations + and · creates commutative ring $(Z_n; +, \cdot)$, where the interest is focused on operation of summation on commutative group $(Z_n; +), n \in N, n > 1$. The embedding of secret message is carried out by conforming of DCT coefficient's remainder r_i to secret data symbol M_i. The secret data are coded from its digital stream of zeros and ones to form of m-symbol alphabet, where $m = \{0,1,\cdots,2^s - 1\}$. Thereby, cardinality of the commutative group $(Z_n;+)$ created by remainder of modulo n must be equal to the number of m-symbol alphabet of secret message M_i. The following example in Tab. 1 shows the embedding of secret message to DCT coefficients using modulo $n=4$ with given conditions

$$r_i' = C_i^k' \bmod n, \tag{1}$$

$$r_i' == M_i, \tag{2}$$

where modified coefficient C_i^k' equals to

$$C_i^k' = C_i^k + x_i, \tag{3}$$

C_i^k	$r_i = C_i^k mod 4$	M_i	x_i	C_i^k'	$r_i' = C_i^k' mod 4$
-5	3	3	+0	-5	3
-10	2	1	+3	-7	1
3	3	0	+1	4	0
1	1	2	+1	2	2
8	0	3	+3	11	3
-1	3	2	+3	2	2

Tab. 1. An example of embedding message with modulo operator.

Parameter x_i represents the value that must be added to C_i^k DCT coefficient to obtain C_i^k' modified DCT coefficient, which fulfills the condition (2). If we take into account that DCT coefficient equals C_i^k = -10 and secret decimal number is equal to M_i = 1 as an example, the desired change of DCT coefficient C_i^k that have to meet condition (2) could also be realized by extracting the value x_i = -1 as well as x_i = +3 in order to obtain the same remainder result $r_i' = C_i^k'$mod4 as M_i, due to cyclic group attribute of modulo remainder. Authors introduce *Modulo Window* (*MW),* which delineates the range of possible values x_i (further referred as x_{mw}) that specifies the way of change of DCT coefficients. The modulo windows is given as follows

$$MW = \left\langle -\text{floor}\left(\frac{n-1}{2}\right), \text{ceil}\left(\frac{n-1}{2}\right)\right\rangle \tag{4}$$

where n is parameter of modulo function and *floor* is rounding function that rounds the expression in brackets to the nearest integers towards infinity and *ceil* function towards minus infinity. The receiver of stego-image simply calculates the remainder r_i' from modulo function of modified coefficient $C_i^k{}'$mod4 to obtain part of secret message M_i. The general scheme of the proposed algorithm is shown in Fig. 1.

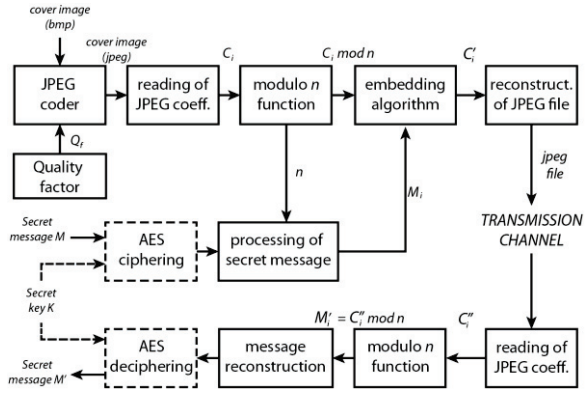


Fig. 1. General scheme of the proposed JPEG steganography system.

An additional block of symmetric ciphering could be applied to a secret message to achieve better results in matter of better histogram preserving as it will be shown later in the proposed method.

2.1 Histogram Analysis and Quality Stego Image

One of important aims of the proposed method, besides increasing the capacity and imperceptibility, is to follow the security level of created steganography system. The concept of histogram analysis was used to prove proposed method's security against steganalysis. The individual frequency components of DCT coefficients were analyzed before and after the change. The difference between the histograms is objectively measured by Kullback-Leibler diverge as it is proposed in [3], which formula is given as

$$D_{KL} = \sum_i P_i . \log \frac{P_i}{P_i'} = \sum_i \frac{h_i}{N_{total}} . \log \frac{h_i}{h_i'}, \tag{5}$$

where P_i and P_i' are probabilities of occurrence for transform DCT coefficients C_i^k prior and after embedding $C_i^{k\prime}$ and N_{total} represents the number of coefficients for each frequency component AC_k. If the input image size is 512x512 pixels, the 2D-DCT transform processing the image in block of 8x8 what means that N_{total}= 4086. The imperceptibility of secret message embedded into cover-image is measured by *PSNR* [dB] (Peak Signal Noise Ratio), which objectively measures the degradation of stego-image compared to cover-image.

3. Proposed MHF Method with Histogram Preserving Scheme

The Modulo Histogram Fitting (MHF) method uses previously discussed principle of modulo embedding. However, to improve the after-change histogram characteristics, the modulo window is modified according to formula

$$MW_{MHF} = \left\langle -\text{ceil}\left(\frac{n-1}{2}\right), \text{ceil}\left(\frac{n-1}{2}\right)\right\rangle. \tag{6}$$

As it was shown the regular modulo MW with its x_i values is not symmetric around zero value. Hence, the applied changes by this modulo window could significantly affect the resulting histogram. This drawback could be suppressed by flipping the modulo window for each consecutive embedding to preserve symmetric histogram mean value of DCT coefficients for each frequency component. The authors call this method *Modulo Window Flipping method (MWF)* and with execution it is close to simple LSB JPEG embedding method. Another alternative, which shows better results, is to widen the modulo window that induce an ambiguity of addressing the value x_{mw} which changes DCT coefficients. As a result, an additional decision making algorithm was implemented that analyzes the cover-image histogram of frequency component and refers which ambiguous value x_{mw} of modulo window MW_{MHF} is going to be used. The algorithm is executed for each embedded DCT coefficient as follows.

At the beginning of embedding, the flag mark of all DCT coefficients is set to null value.

flag(C_i^k) = flag(C_i^k)+1

if abs(x_{mw}) = ceil $\left(\frac{n-1}{2}\right)$

  x_{mw}= abs(x_{mw})

  if flag($C_i^k - x_{mw}$) ≥ flag($C_i^k + x_{mw}$)

    $C_i^{k\prime} = C_i^k - x_{mw}$

    flag($C_i^{k\prime}$) = flag($C_i^{k\prime}$) - 1

  else

    $C_i^{k\prime} = C_i^k + x_{mw}$

    flag($C_i^{k\prime}$) = flag($C_i^{k\prime}$) - 1

  end

else

  $C_i^{k\prime} = C_i^k + x_{mw}$

end

1. The flag of present DCT coefficient is increased by one what means DCT coefficient was used for embedding.
2. If x_{mw} value reaches ambiguous value of modulo window MW_{MHF} that equals to ±ceil[(n-1)/n] then x_{mw} value is set to its absolute value. Thereafter the flags of DCT coefficients, which are obtained by subtracting and adding of value x_{mw}, are compared.

3. If the flag from previous iterations of DCT coefficient $C_i^k - x_{mw}$ is higher or equal to $C_i^k + x_{mw}$, then the resulting coefficient $C_i^{k\prime}$ is given as $C_i^{k\prime} = C_i^k - x_{mw}$. Otherwise, the modified DCT coefficient is equalled to $C_i^{k\prime} = C_i^k + x_{mw}$.
4. In the next step, the flag of modified coefficient $C_i^{k\prime}$ is decreased by one in order to compensate executed change.
5. If x_{mw} is different than ambiguous value from modulo window MW_{MHF} than modified DCT coefficient is given by $C_i^{k\prime} = C_i^k + x_{mw}$.

The provided algorithm allows compensate changes on initial histogram by recording previously embedding changes of DCT coefficients. Another criterion that significantly affects distorting of histogram characteristics is the conformity of probability. The histogram preserving embedding is realized under condition that the probability of symbol in *m*-symbol secret data to be embedded is equal to the probability of quantized DCT coefficients of cover-image that are belonging to representatives of one of *m*-state quantizers. Hence, there are two possibilities to overcome this problem. The first one reckons on probability change of a secret message to comply the probability of DCT coefficient assigned to *m*-state quantizer. The probability change is performed by entropy encoder that is applied to a secret message for each frequency component individually. However, it is needed to remark that entropy encoder could significantly increase the message length depending on histogram of DCT coefficients for each frequency component AC_k. The second possibility of overcoming the problem of histogram preserving uses specific amount of selected coefficients from all DCT coefficients for embedding to fulfil the condition of equal probability. The proposed method uses the first alternative or its variation, where the probability change of secret message symbols is processed by implementing symmetric encryption AES 128-bit cipher with the secret key of the same size. The probability is uniform distribution for all symbols of a secret message therefore it is not adequate mapping to DCT coefficient probability that are assigned to *m*-state quantizer. However, this approach significantly enhances post-change histogram characteristic without enlarging message size (except ciphering overhead). Another positive contribution of implementing ciphering is advancing of security level to proposed steganography system.

4. Experimental Results

The proposed method does not have any limitation regarding transfer capacity due to modulo embedding. If a higher value of *n* modulo is used (embedding the same length of the secret message), less frequency components and thus less DCT coefficients are modified during the embedding what also influences quality of stego-image. On the other hand, the higher value shift of DCT coefficients is executed due to modifying value x_{mw} that ranges from wider modulo window *MW*. Fig. 2 depicts relation between *n* modulo values and capacity of the method for cover-image Lena.jpg (512x512 pixels), where 63 frequency components were used for embedding except DC coefficient. The proposed method's capacity increases exponentially with higher *n* modulo value.

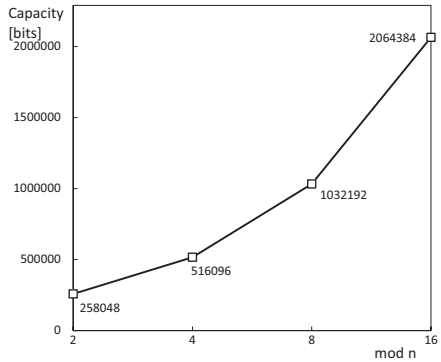


Fig. 2. The capacity in [bits] of the method depending on value of *n* modulo.

As an example of before and after-embedding histograms, histograms of AC_2 and AC_5 freq. component among 64 components for Lena.jpg image (with size of 512x512 pixels, 8bpp, $Q_f = 80$) could be shown. The embedding message was size of 64800 bits. Fig. 3 and Fig. 4 show examples of after-embedding histograms of the proposed methods Modulo Window Flipping (MWF), Modulo Histogram Fitting (MHF) and MHF with AES ciphering (MHF-AES) using modulo function *n*=2.

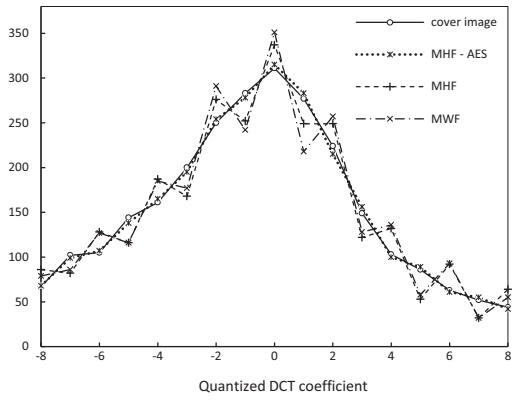


Fig. 3. After-embedding histogram of AC_2 with *mod 2* embedding.

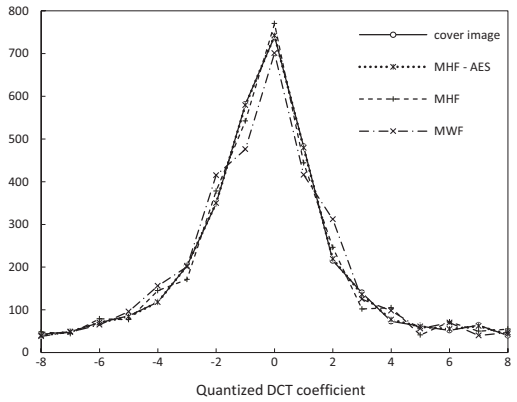


Fig. 4. After-embedding histogram of AC_5 with *mod 2* embedding.

The embedding using mod n=2 with MHF without ciphering accounts for better results as MWF especially in AC_5, where MWF shows asymmetric after-embedding characteristics. Even though, MHF and also MWF show undesirable saw tooth run of after-embedding histogram due to previously mentioned condition of probability, which was not analyzed and considered during methods' implementation. The proposed method MHF-AES by reason of changed probability of secret message's symbol accounts for more nearer after-embedding what is also expressed by divergence coefficient D_{KL} in Tab. 2. Fig. 5 shows embedding using mod n=4, where the significant deviation of after-embedding histogram for MWF and MHF methods is plotted. Even though higher modulo n accounts for better or equal quality of stego-image with two times higher capacity than n=2, after-embedding histogram shows significant distortion. Despite of notable results of MHF-AES comparing to MHF and MWF, embedding with modulo $n \geq 4$ cannot be considered as secure steganography system due to significant difference between after-embedding and before-embedding histograms.

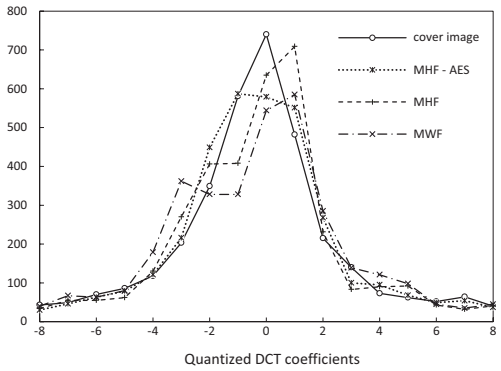


Fig. 5. After-embedding histogram of AC_5 freq. component with *mod 4* embedding

Tab. 2 also shows all results of stego-image quality measured by PSNR [dB] and general D_{KL} divergence coefficient after-embedding histogram in dependence on modulo n={4,8,16}, where Lena.jpg (512x512 pixels) cover-image was used and the embedded secret message was size of 64800 bits, which also represents 25.11% of the utilized capacity in case of modulo n=2.

	mod2		mod4		mod8		mod16	
Method	D_{KL}	PSNR	D_{KL}	PSNR	D_{KL}	PSNR	D_{KL}	PSNR
MWF	0.0663	36.12	0.0882	36.65	0.1773	35.07	0.5502	31.76
MHF	0.0412	36.26	0.0709	36.70	0.1728	35.08	0.5477	31.76
MHF-AES	0.0352	36.17	0.0669	36.68	0.1468	35.11	0.3906	31.80

Tab. 2. Proposed method's results of quality of stego-image and histogram.

Experimental results of the proposed methods do not account for significant differences in stego-image quality or any in transfer capacity; they only differ in D_{KL} divergence of after-embedding histograms.

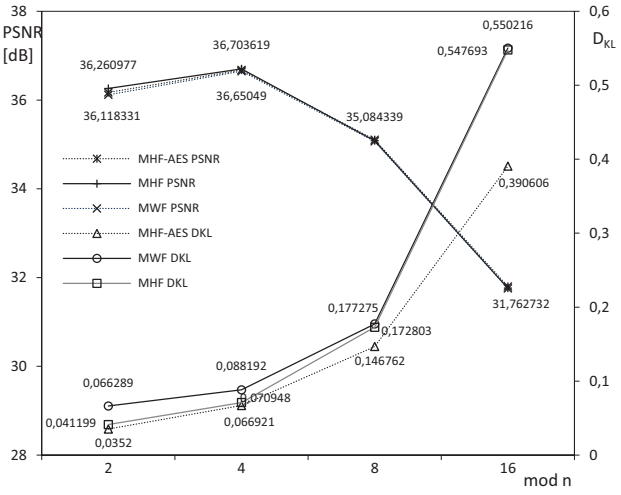


Fig. 6. The quality of stego-image and D_{KL} coefficient in relation to n modulo.

The results also show that the best stego-image quality is obtained by embedding with mod n={2,4}. In case of n=2, the proposed method can be considered as secure against histogram attacks what cannot be assessed about n=4 or higher n modulo due to the shown histogram difference and D_{KL} results from Tab. 2. Fig. 6 shows that each increasing of modulo n doubles the capacity of the steganography system, however at the expense of quality of stego-image and security level of steganography system. Fig. 7 shows testing different detail level cover-images (Mandrill512.jpg, Cameraman512.jpg and Lena512.jpg), where images were created from BMP image with JPEG coder and given input parameter i.e. quality factor Q_f=80. Hence results of the proposed methods do not differ significantly in PSNR [dB], mean values are given in the figure.

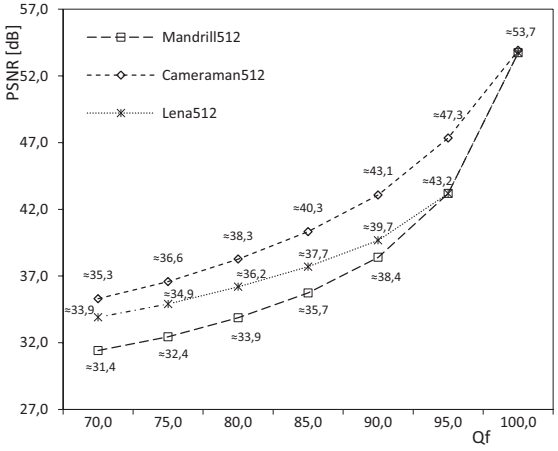


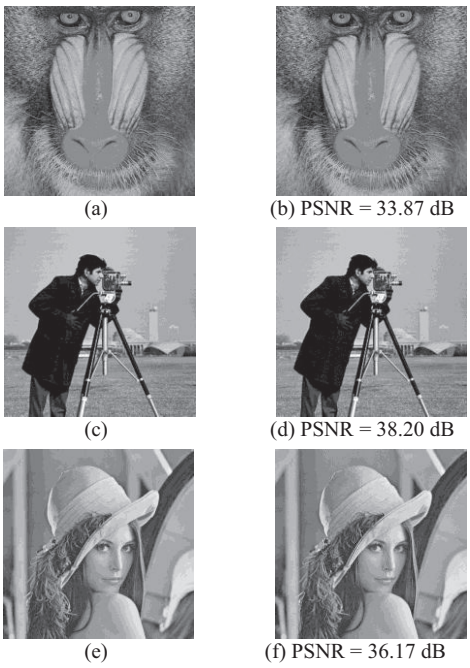
Fig. 7. The quality of stego-image for different cover-images used for embedding with variation of quality factor Q_f, mod n=2.

Tab. 3 shows a comparison of the proposed MHF-AES method with the well-known JPEG steganography F5 method in JPEG file in manner of preserving local histograms and quality of stego-images. MHF-AES accounts for better results in before and after-embedding histograms expressed by D_{KL} divergence.

Cover-image	Method	Embedded data size [bits]	PNSR[dB]	D_{KL}
Lena	F5	35174	37.260	0.04783
	MHF-AES	35376	37.756	0.00942
Mandrill	F5	61481	31.890	0.04543
	MHF-AES	61248	33.979	0.00543

Tab. 3. Results of F5 and proposed MHF-AES method with embedding an equal amount of secret data.

Testing covers with size of 512x512 as well as stego-images are illustrated in Fig. 8.



(a) (b) PSNR = 33.87 dB

(c) (d) PSNR = 38.20 dB

(e) (f) PSNR = 36.17 dB

Fig. 8. (a), (c) and (e) are cover-images; (b), (d) and (f) are stego-images obtained by embedding with MHF-AES method.

5. Conclusion

The proposed method MHF with histogram preserving algorithm uses modulo arithmetic for embedding a secret message in JPEG file format. Embedding of a secret message consists in alternation of DCT coefficients in such a manner that it's calculated modulo value equals a secret symbols. An application of modulo arithmetic creates a blind steganography system, which does not need cover-image in process of extraction.

The capacity of the proposed method is not restrained. However, using higher modulo $n > 2$ makes the proposed method not secure against histogram attacks of steganalysis. The quality of stego-image also depends on selected modulo n as it was previously discussed and experientially proven. Implementation of AES ciphering on secret data before embedding improves after-embedding histograms of DCT frequency components due to condition of probability conformity between secret data and DCT coefficients of frequency components. The proposed method was analyzed and compared in term of distortion of after-embedding histogram. Kullback-Leibler divergence was introduced to objectively measure histogram distortion due to embedding of the secret message. The comparison between MHF method and its variation MWF or simple LSB embedding in JPEG was carried out to show contribu-tion of implemented histogram preserving algorithm of MHF.

The MHF method offers variability in transfer capacity with desired stego-image quality as well as steganalysis security against histogram attacks. Implemented 128bit AES ciphering of the secret message also increases security from cryptographic perspective. The majority of experiments were carried out on JPEG files with quality factor Q_f=80 that is widely used. If higher quality factor is applied during the compression of JPEG coder, more redundancy is at disposal what creates predisposition of more secure stego-image with better quality in a manner of PSNR value and vice versa as it was shown.

Acknowledgements

The paper was supported by the Ministry of Education of the Slovak Republic VEGA Grant No. 1/0386/12 (50%) and EU FP7 project INDECT No. 218086 (50%).

References

[1] BÁNOCI, V., BUGÁR, G., LEVICKÝ, D. Steganography system using by CDMA techniques. In *Proc. of the 19st International Conference Radioelektronika.* Bratislava (SR), 2009, p. 183-186.

[2] BANOCI, V., BUGAR, G., LEVICKY, D. A novel method of image steganography in DWT domain. In *Proc. of the 21st International Conf. Radioelektronika.* Brno (Czechia), 2011, p.1-4.

[3] BÖHME, R. *Advanced Statistical Steganalysis.* Springer, 2010.

[4] BRAZIL, A. L., SANCHEZ, A., CONCI, A., BEHLILOVIC, N. Hybridizing genetic algorithms and path relinking for steganography. In *Proc. of ELMAR*. Zadar (Croatia), 2011, p. 285–288.

[5] CHANG, C.C., CHEN, T.S., CHUNG, L.Z. A steganographic method based upon JPEG and quantization table modification. *Information Science,* 2002, vol. 141, p. 123–138.

[6] COX, I. J., MILLER, M. L., BLOOM, J. A., FRIDRICH, J., KALKER, T. *Digital Watermarking and Steganography.* USA, 2008. ISBN 978-0-12-372585-1.

[7] FRIDRICH, J., GOLJAN, M., HOGEA, D. New methodology for breaking steganographic techniques for JPEGs. *Proc. SPIE*, 2003, vol. 5020, p. 143–155.

[8] JOHNSON, N., JAJODIA, S. Exploring steganography: Seeing the unseen. *Computer,* 1998, vol. 31, no. 2, p. 26-34.

[9] PROVOS, N. Defending against statistical steganalysis. In *Proc. of the 10th USENIX Security Symposium*. 2001, p. 323–335.

[10] Steganography software for Windows, July, 2004. [Online]. <http://members.tripod.com/steganography/stego/ software.html>.

[11] WESTFELD, A. F5—A steganographic algorithm: High capacity despite better steganalysis. *Lect. Notes Comput. Sci*., 2001, vol. 2137, p.289–302.

[12] UPHAM, D. *JPEG-JSTEG Steganography Method*, 1997. Available at: <http://ftp.funet.fi/pub/crypt/steganography/jpeg-jsteg-v4.diff.gz>.