Soft Comput (2015) 19:3331–3339
DOI 10.1007/s00500-015-1728-z

METHODOLOGIES AND APPLICATION

False and multi-secret steganography in digital images

Marek R. Ogiela[1] · **Katarzyna Koptyra**[1]

Published online: 12 June 2015

Abstract This paper presents the new concepts of multi-secret and false digital image steganography. The main idea of such approaches is to embed in a single container (digital image) more than one message. The hidden secrets are called *real* and *false messages*, respectively. The first one contains essential data which are intended to be securely transferred between different parties, the latter is a bait for focusing attention on an unimportant message. This false and multi-secret steganography will be broken when existence of the real message is revealed, it does not matter whether the false message is detected. Such concepts may find many different applications, especially in situations where communication channel between a sender and a receiver is closely monitored and the warden suspects that the steganography is used. In that case it is probable that the transmitted data will be analyzed in a very detailed way. The concepts described in this paper can help to overcome this problem by dropping a fabricated message and thereby deceiving the warden. The possibility of sending both real and false information at the same time can be seen as additional benefit. In fact, the presented idea allows to establish a kind of subliminal channel while transferring hidden information using digital images.

Communicated by V. Loia.

✉ Marek R. Ogiela
mogiela@agh.edu.pl

Katarzyna Koptyra
kkoptyra@agh.edu.pl

[1] Cryptography and Cognitive Informatics Research Group, AGH University of Science and Technology, Al. Mickiewicza 30, 30-059 Kraków, Poland

Keywords Cryptographic protocols · Digital image steganography · Information hiding · Intelligent information splitting · Multi-secret steganography

1 Introduction

Steganography is a technique of hiding information in a way that prevents detection of secret data by unintended recipients (Bailey and Curran 2005; Cheddad 2009; Cox et al. 2008; Subhedar and Mankar 2014). The usage of steganography is successful when existence of the secret message is not revealed; in the other case it is broken. Considering above, one of the most important aspects of any steganographic method is to provide high level of undetectability (Fridrich 2010; Katzenbeisser and Petitcolas 2000).

In the popular form of digital steganography, called container modification, the carrier is altered in embedding process and then sent to the receiver who extracts data from it. The problem is that every modification of the container changes slightly its statistics (e.g., histogram). If adversary (warden) gets access to carrier, he will try to check it for existence of the secret message. The actions taken by warden to find hidden information are called steganalysis and in many cases are based on statistical distribution, e.g., (Budhia et al. 2006; Cheddad 2009; Fridrich et al. 2003).

The length of embedded data affects number of changes made to container and, as a result, the likelihood of detection. Therefore, capacity and undetectability are competitive requirements (Fridrich 2010; Fridrich et al. 2003; Tang et al. 2014) and obtaining satisfactory level of both is a difficult problem in steganography.

A proper choice of carrier is crucial issue that affects system security. From a wide variety of digital containers, the most common are images, network packets, text, video and

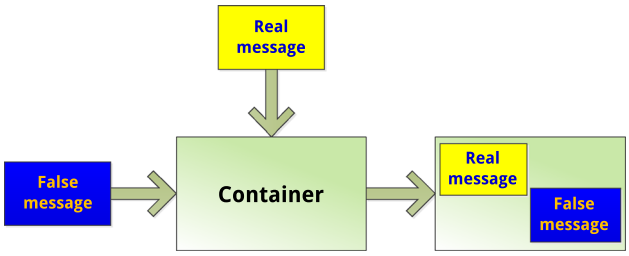

Fig. 1 An idea of multi-secret steganography

audio files because of its broad usage in network communication. However, there were also some proposals of utilizing different media for steganography: word documents (Castiglione et al. 2007, 2011b), e-mail messages (Castiglione et al. 2011c, 2012) or social networks (Castiglione et al. 2011a). Considering multimedia files, most of the techniques of hiding data are based on the limitations of the human senses. Some colors and sounds cannot be distinguished by a human brain and are seen as identical. This fact may be applied to conceal some information, e.g., in digital images.

2 Multi-secret and false steganography

As mentioned above, embedding secret message in a carrier file, in most cases results in modifications of the container. These changes can be detected by targeted or blind steganalysis algorithms (Castiglione et al. 2011c; Cox et al. 2008; Westfeld 2001). This is the reason why many steganographic methods are focused on minimizing impact of embedding. In the proposed approach some modifications of carrier are intentionally made to increase undetectability of some data at the expense of another.

In false or multi-secret steganography there are two types of messages:

- Real message (as a main secret information) which should be transferred inconspicuously from sender to receiver,
- Additional one or several artificial (false) messages that are not important at all for covert communication but its aim is to deceive adversary (unauthorized observer or parties that may supervise the communication channel).

Both messages, real and false, are embedded into a single container (Fig. 1). Any steganography method can be used but with assumption that detection of the real message should be much harder than the false message. Of course carrier modifications are larger than in case when one secret information is embedded but existence of the false message, which is easier to detect, divert attention from the real one. It is likely that if the false message is revealed, the warden will be convinced that the secret communication has been found and will not continue searching.

There are three possible situations:

1. All messages are not detected—false steganography is successful.
2. False message (one or more) is revealed but real message remained secret—false steganography is successful.
3. Real message is detected—false steganography is broken; detection of false message does not matter.

The above possibilities concern detecting data by the warden. On the other hand, the receiver should know the algorithm (and possibly the key) needed to extract real message. It is possible but not necessary for the real user to know method of obtaining false secret as it is not important to covert communication.

Disclosure of the false message is acceptable as its aim is to misinform adversary by change statistics of the container. From the perspective of steganalyst, carrier modifications are made in place where the data is hidden. With application of the rule that the real message should be harder to detect than the false one, it is highly likely that only false information will be found and real message will stay unseen. In fact it is an essence of false steganography.

It is also possible to hide the real message in a container and then use it as next message and embed it in an another container. This approach may be treated as multi-level steganography (Subhedar and Mankar 2014; Tang et al. 2014) where the real message is hidden on the last level. The main difference between two concepts is that in multi-level steganography extracting messages on all levels is required to get secret message. Both ideas can be used together but it should be noted that every level limits available capacity. Figures 2 and 3 show the presented approaches.

Described concept does not protect from situation when existence of secret data is revealed and, in consequence, the warden will block communication. However, it should be pointed out that steganography also does not do it. Detection of the false message gives adversary information that the covert communication takes place. This fact could be seen as disadvantage but it is possible to use it for benefits. False message can be prepared not only to focus attention but also to intentionally give warden incorrect information in case of detection.

3 Examples of application

The presented idea can be used with any steganographic method. This paper focuses on image steganography. Below there are described some experiments which were made with different algorithms.

A popular technique of steganography is image LSB, in which the secret information is embedded in least significant bits of pixels. Hence, this method will be used to hide

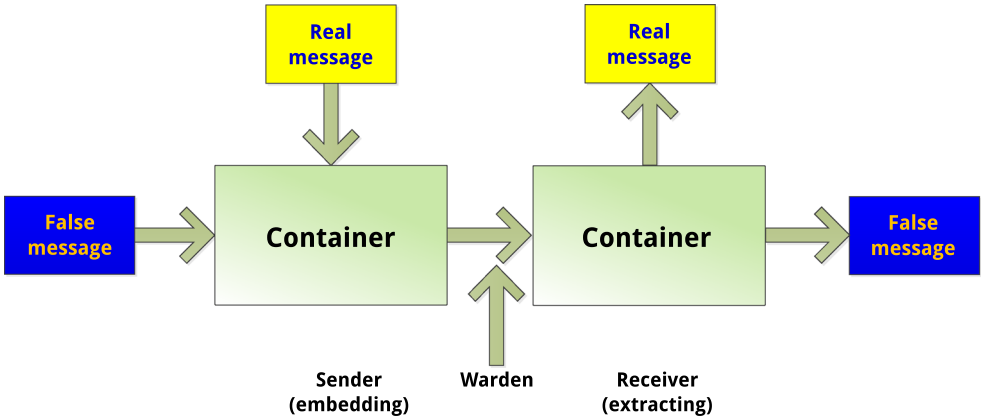

Fig. 2 General idea of false steganography

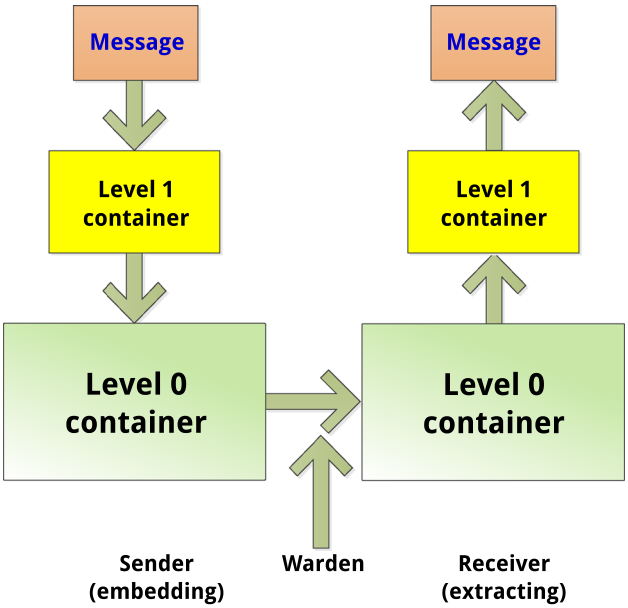

Fig. 3 Multi-level steganography

false message. The real message will be embedded in second LSB but for example only in red and blue component. This choice has been made because human eye is much more sensitive to green than other hues (Nagaraj et al. 2013; Qazanfari and Safabakhsh 2014; Thompson et al. 1992). As a result, the maximum capacity is reduced but undetectability of the real message will increase. The proper selection of carrier can improve undetectability even better. Mostly green images, for example forest or leaves, are a reasonable choice.

The process of embedding the real message is defined as follows. First of all, a secret key is used to create permutation of pixel indices. The aim of this step is to scatter modifications across image to avoid concentrating changes in the one part of the carrier file which is most visible when length of the secret message is smaller than container capacity. Let p_i be a i-th permutation element. A message of length n will be hidden in pixels located at positions $p_0, \ldots, p_{n-1}$. Depending on parity of p_i, the red or blue component is chosen. Then second LSB of selected component of actual pixel is replaced by ith message bit.

To extract real message, the receiver has to compute permutation from the same key to find modified pixels and then read second LSB of proper component. The secret key can be shared between the sender and the receiver or steganographically hidden in container. In presented implementation, a digest of the false message is used as a secret key and hashing algorithm is SHA-256. False message is embedded in LSB so there is no collision between embedding and extracting algorithm as they operate on different data.

The chosen secrets are digital images presented in Fig. 4 which were encrypted with Rijndael-256 before hiding. Figure 5 shows pure container and the result of embedding real, false and both messages with use of above technique.

The encryption causes that the occurrence probabilities of 0 and 1 in output data are equal (Qazanfari and Safabakhsh 2014). During the embedding, both real and false messages introduce some disruptions to the container but in a different way (Fig. 5). As mentioned earlier, the false message is hidden with LSB technique which is considered not very secure because of characteristic artifacts created in the histogram. The studies of this method show that adjacent bars (which differ only at the least significant bit position) are equalized as a result of embedding (Cox et al. 2008). These abnormalities do not occur normally in digital images, which leads to an obvious conclusion that the secret data are present. The shape of the histogram indicates LSB method; thus, there is probability that the false message might be detected during steganalysis. That is how an unauthorized user can obtain the image from Fig. 4b. In the presented example, the false secret was encrypted which may additionally suggest that the hidden information is important. From the attacker point of view the secret was found but for the participants it can be seen as a false-positive error.

Fig. 4 Secret images. **a** Real secret and **b** false secret

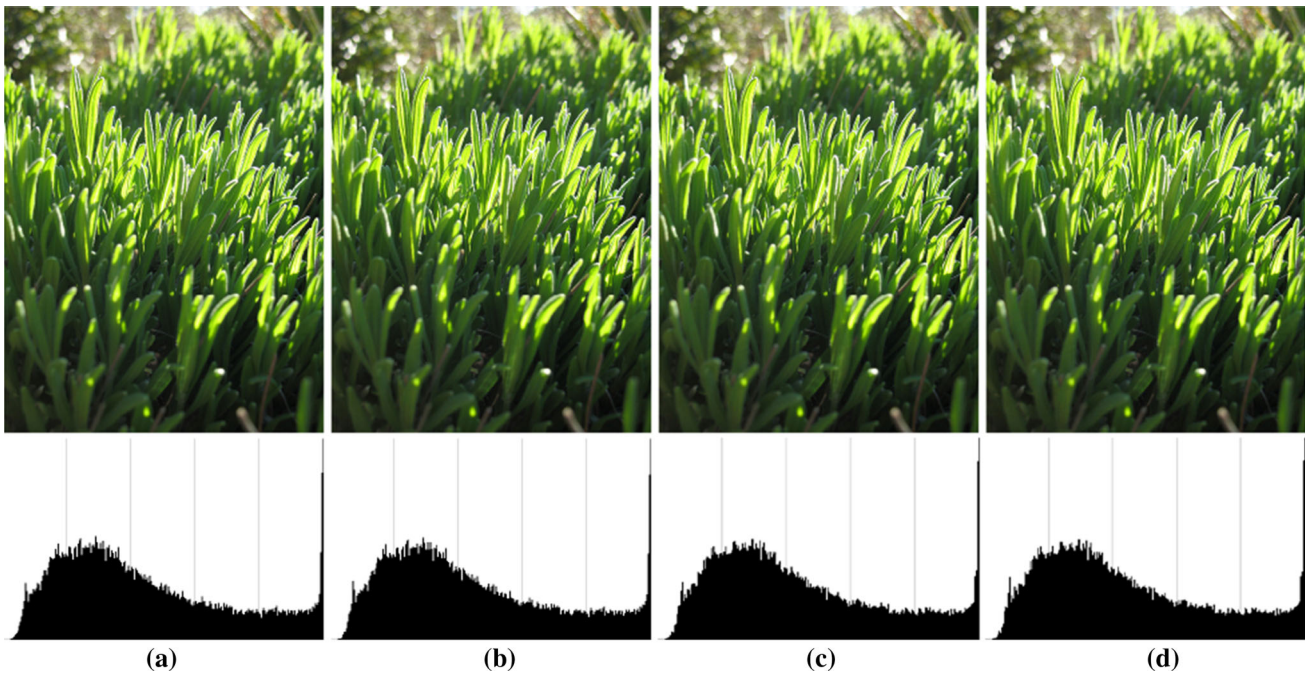

Fig. 5 Example of LSB multi-secret steganography application with histograms of values. **a** Original image, **b** container with false secret, **c** container with real secret and **d** container with real and false secrets

It is also possible to use various steganographic methods and hide real and false message in different ways. A good example could be JPEG steganography where:

- Real message can be embedded in frequency domain,
- False message can be hidden in file header.

JPEG file format (Fridrich et al. 2003; Ong et al. 2015) is intended for storing lossy compressed image data. The header of a file is divided into segments and each one starts with a pair of bytes (marker). First byte of marker is always 255 and the second, used for distinction, may vary. The one of the types of segments is COM (comment) identified with marker 254. It may contain, among others, information about quality and program used to create image. As the content length and structure is not defined, it can be used to hide some steganographic data. Thus, the false message will be placed in comment segment of the header.

Image data are stored as quantized frequency coefficients. Encoding is conducted in following steps. First, uncompressed image is divided into 8×8 pixel squares. Then every piece is transformed into frequency domain with discrete cosine transform. The next operation is quantization which results in rounding coefficients and, as a consequence, loss of some information. The obtained values are arranged in a zigzag order which causes concentrating zeros at the end of the stream. Finally, data are coded to compress redundant data.

The real message will be hidden in JPEG image with use of F5 algorithm (Fridrich et al. 2003; Westfeld 2001). The embedding process is performed after the quantization of coefficients. First, there is a computed permutation which uses strong random number generator based on a secret key.

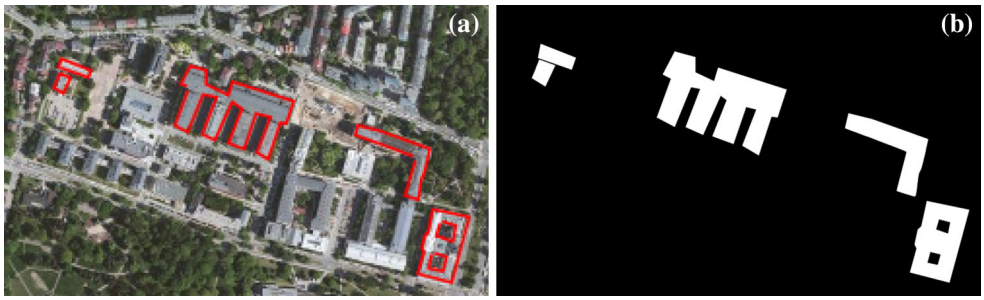

Fig. 6 Reducing secret message size. **a** Choice of interesting regions of image. **b** Real secret (vector image)

The intent of permutative straddling is to uniformly distribute modifications over the whole image. Then the message is hidden in nonzero coefficients with matrix encoding. This technique serves to improve the embedding efficiency and, in consequence, to reduce the number of necessary changes. The parameters of matrix encoding are related to capacity of the carrier medium and the length of the message. After those steps, the JPEG compression is continued.

As mentioned, the permutation depends on the secret key so sender and receiver should use the same sequence. As in first example, the use of key may be based on the false message digest. The aim was to hide image presented in Fig. 4a but its size exceeds carrier capacity. Thus, some strategic objects from picture were chosen (Fig. 6a), then saved as SVG (Scalable Vector Graphics) file (Fig. 6b) and encrypted. It helped to reduce the real message size to about 1/3 of initial value. The false message is injected into header, so ASCII text file (2407 characters) is used.

It should be noted that in JPEG files it is possible to manipulate quality of the image by setting a quantization factor. Its value affects not only quality, but also file size. Additionally, the latter is dependent on the false message size. Therefore, proper choice of quality factor and the false message length can help increasing undetectability. Figure 7 shows the results of experimentation with above technique (false message size = 2.4 KB; real message size = 3.6 KB).

In this case embedding the false secret has no influence on the pixel values, only on the file size. All modifications of image content are introduced by hiding the real message. This is the reason why containers with one message are not presented in separated figure as they are visually identical to pure carrier or one with both secrets.

F5 algorithm can also be applied to conceal two different secrets in the coefficients with interlacing. In proposed method both messages (longer and shorter) are combined before shuffling coefficients. This process is shown below (Algorithm 1 with example in Fig. 8). It is important to mention that a secret key is required—it has to be shared between the sender and the receiver. For security reasons the key used in coding algorithm should not be the same like the one in F5.

Algorithm 1 Coding

Input: *longMessage*, *shortMessage*, *key* (arrays of bytes)
Output: *data* (array of bytes)
Compute *lengthL* and *lengthS* (lengths of *longMessage* and *shortMessage*)
Use *key* to find permutation P from 0 to $lengthL-1$
$P1$ = *lengthS* first elements of P
$P2$ = sorted $P1$
Create empty *data* array of length $lengthL+lengthS$
for every $i=0,\ldots,lengthS-1$ **do**
    $index = P2[i]+i$
    $data[index] = shortMessage[i]$
Fill not used places in *data* array with bytes of *longMessage*
return *data*

The output data are then embedded in an identical way like in F5 method. Permutation in F5 spreads information across the image so byte order is not preserved. It is reason why basic shifting is satisfactory. In presented implementation lengthL is put on the beginning of the array as it is needed to decode secrets. Algorithm 2 depicts how to recover messages from data obtained from coefficients.

Algorithm 2 Decoding

Input: *data*, *key* (arrays of bytes), *lengthL*
Output: *longMessage*, *shortMessage* (arrays of bytes)
lengthS = length of *data* - *lengthL*
Create empty arrays:
    longMessage of length *lengthL*
    shortMessage of length *lengthS*
Use key to find permutation P from 0 to $lengthL-1$
$P1$ = *lengthS* first elements of P
$P2$ = sorted $P1$
for every $i=0,\ldots,lengthS-1$ **do**
    $index = P2[i]+i$
    $shortMessage[i] = data[index]$
Fill *longMessage* array with unused bytes from *data* array
return *longMessage*, *shortMessage*

In a single container, there were hidden two secrets which were encrypted images previously shown in Fig. 4b (longer message) and Fig. 6b (shorter message). The used keys were “Science” and “4511932” for F5 permutation and coding/decoding algorithms, respectively. The results of experiments are presented in Fig. 9. The chosen quality factor affects file size and, therefore, available capacity.

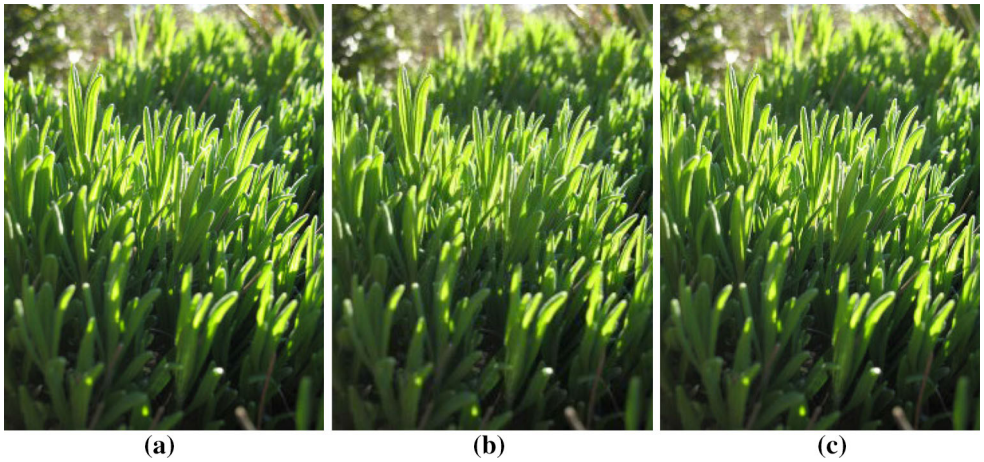

Fig. 7 Example of JPEG multi-secret steganography application. **a** Original image (43.4 KB), **b** stego image (30.0 KB), quality = 88 and **c** stego image (76.1 KB), quality = 100

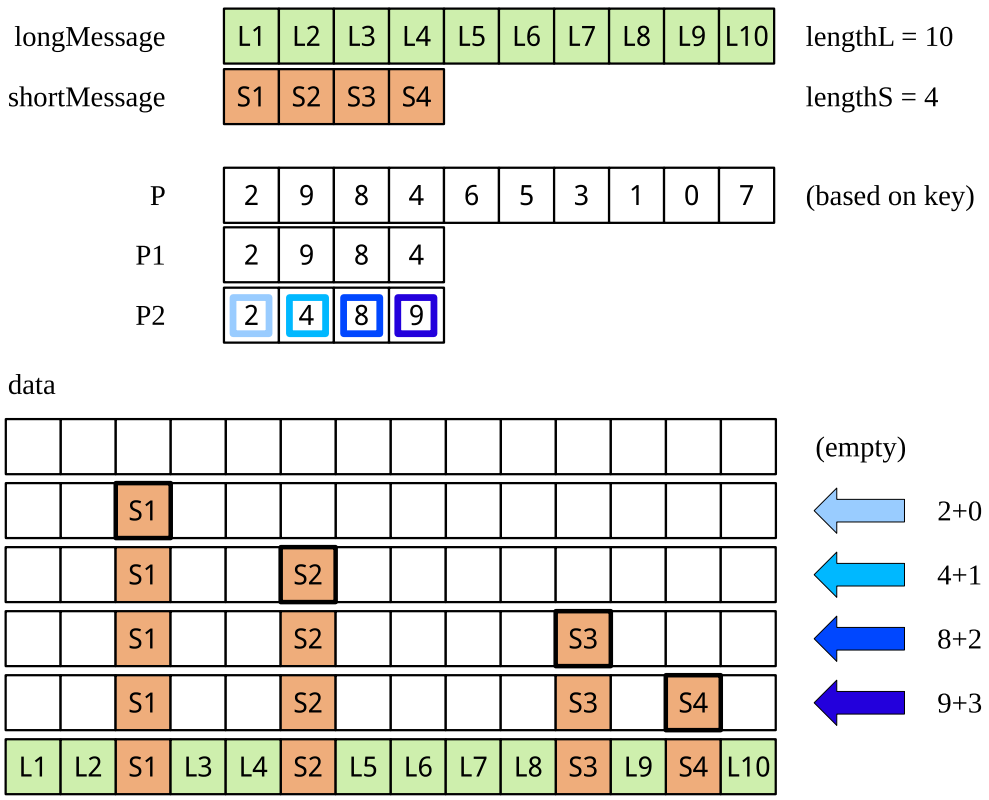

Fig. 8 Example of algorithm 1

Another approach is to use multi-level steganography and embed the secret message on higher level. This attempt was made with the following steps. First, encrypted SVG file from Fig. 6b was hidden in the digital image depicted in Fig. 5a. Then newly created object (Fig. 10b) was used as next secret and embedded in another container (Fig. 10a). The final result is shown in Fig. 10c. In comparison, in Fig. 10d is presented effect of omitting first step and using only Figs. 5a and 10a images.

It should be pointed out that every level of multi-level steganography limits capacity (Yuan 2014). In the presented example the message size is 3.6 KB and the sizes of carriers are 43.4 KB and 412.5 KB on level 1 and level 0, respectively.

4 Features of different approaches for multi-secret steganography

In this paper, there were presented two different approaches to multi-secret steganography called false steganography and

Fig. 9 Example of F5 multi-secret steganography application. **a** Original image (412.5 KB), **b** stego image (271.3 KB), quality = 88 and **c** stego image (735.5 KB), quality = 100

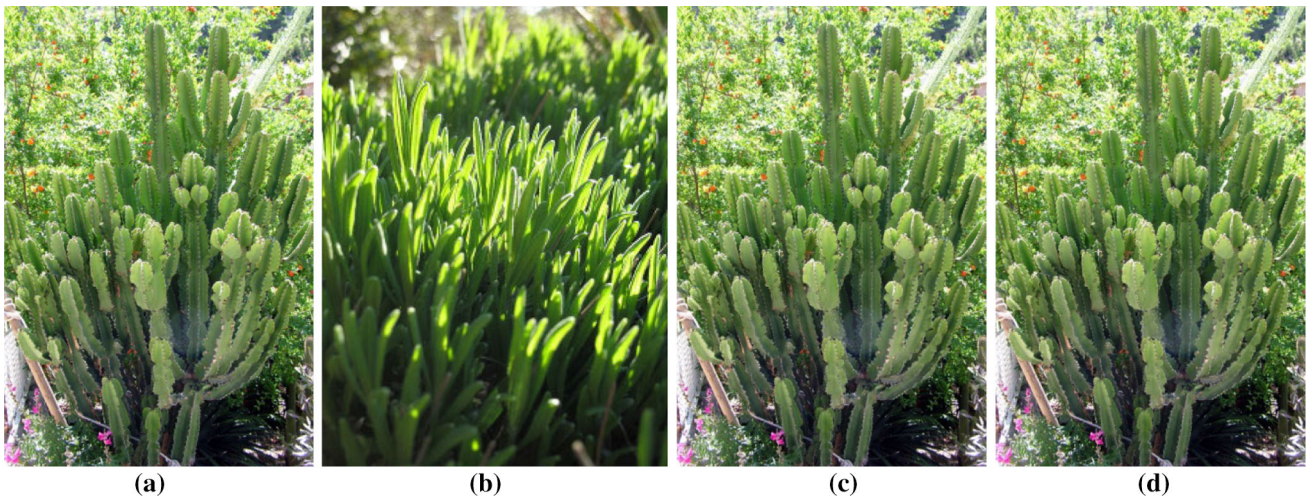

Fig. 10 Example of F5 multi-level steganography application. **a** Level 0 container, **b** level 1 container with secret message, **c** level 0 container with embedded (b) and **d** level 0 container with embedded 5a

multi-level steganography which are intended to hide more than one message in a single container.

In false steganography there is no theoretical limit of secrets amount. However, in practice, below condition has to be met:

$$\sum_{n=1}^{N} \text{secret}_n \text{ size} \leq \text{container capacity},$$

where N is amount of secrets.

Real and false message(s) are embedded independently and, as a consequence, any of them can be extracted without others. In this paper, extension was proposed which uses the false secret digest as a key but in general case messages are not related.

In multi-level steganography decoding all middle messages is required to obtain a final secret. It is unquestionable advantage; nevertheless, the more levels are used, the lower the capacity is. This approach has potential to transfer small messages but at some point it can be impractical. To successfully hide data, the sender has to assure the following condition:

$$\text{Secret size on level } n \leq n-1 \text{ level capacity} \quad \forall n \in 1, \ldots N,$$

where N is top level where real secret is hidden.

Carrier exchanged between the sender and the receiver is on level 0.

Below there are characteristics of the most important features of described techniques of multi-secret steganography.

Table 1 Carrier capacity for real and false secret

Method name	False message	Real message
False LSB steganography	12.5 % (1/8)	4.17 % (1/24)
False JPEG steganography	13.4 % (Fridrich et al. 2003; Westfeld 2001)	Unlimited
Multi-level F5 steganography	$(0.13)^{n+1} \times 100\,\%$ on level n (for both real and false messages)	

4.1 False LSB steganography

This method hides the false message continuously in LSB of all components. Real message is embedded in second LSB of red and blue component exclusively and with permutation. Therefore, available capacity is 1/8 of carrier size for the false secret and 1/24 for the real one (with assumption of 24-bit pixel). An adversary may try to destroy messages by replacing least significant bits or to reveal them with passive attacks. LSB is easy to implement and a popular method, but its security is low as it introduces statistical changes to container that can be detect by histogram analysis.

4.2 False JPEG steganography

Real secret is embedded with use of F5 algorithm. The capacity of this algorithm is about 13.4 % of carrier size (Westfeld 2001). There exists a complex attack which allows estimation of the message length (Fridrich et al. 2003). The false secret is injected into JPEG header. It is possible to use secret of unlimited length but it affects file size and leads to easy detection. Steganalysis techniques are simple, for example checking file header or replacing comment to destroy the message.

4.3 Multi-level F5 steganography

F5 algorithm is used on every level to hide current secret. Available capacity should be computed for every level independently. As only one method was used, estimated capacity on level n is $(0.13)^{n+1} \times 100\,\%$ of the container size. Possible attacks are the same as described for false JPEG steganography except second part as in this case the header remains untouched.

In Table 1 there is a short summary of available capacity of the presented methods.

5 Conclusions

The concept of false steganography presented in this article may find application in many specific situations. When communication channel between the sender and the receiver is closely monitored, every container modification is suspicious so embedding the false message can be an option. There is no need to use false steganography when probability of examination the transmitted data by warden is very small.

Below are compared current key topics in steganography and corresponding issues in the multi-secret and false steganography.

With regard to (Subhedar and Mankar 2014), the most important aspects that should be considered in designing a good steganographic algorithm are imperceptibility, maximum embedding capacity and acceptable level of security (eavesdropper's inability to detect hidden information). Modern methods are expected to satisfy these requirements as well as low computational complexity. Similar issues are essential in false steganography, however, there should be highlighted some meaningful differences between these discussed fields. As noted, the hiding capacity should be greater than or equal to the length of all secrets. Regarding security, the undetectability of the real message is crucial because its exposure denotes the failure of the whole system. On the other hand, protecting the false secret from being detected is not so important as its disclosure is acceptable. Another aspect indicated as significant in (Subhedar and Mankar 2014) is cover selection. In steganography (and multi-secret and false steganography as well) the user is free to select which carrier to use. Therefore, this choice should meet the requirements imposed on an up-to-date information hiding methods.

In watermarking, the situation is different because the container is more important than the message and there is no such control over the selection of the cover. This was worth to mention because there may be seen some similarities between presented idea and unauthorized embedding attack on digital watermarks (Cox et al. 2008). In both cases, additional data are added to the carrier but here the aim is not to forge a mark but to deceive adversary. If the warden detects only false message, the real information will remain securely hidden which is indeed the essence of steganography.

As a final conclusion, it should be indicated which techniques are most appropriate in specific situations. When warden suspects that steganography is used, it is likely that the carrier will be checked for possible presence of a hidden data. In this case the best choice is false steganography. This selection may be justified as follows. The largest modifications of the container are introduced by the false message embedding. Thus, during steganalysis statistical anomalies occur in that place which suggests that essential data were hidden there. If the secret is relatively small, multi-level

steganography may be considered. In situation where there is more than one important message, a reasonable solution is to mix secrets and hide them with hard to detect method. Then it is also possible to treat messages as real secret and then apply the above technique together with another approach presented in this paper. Algorithms of false and multi-secret steganography presented in this paper may be also applied with connection to the visual secret sharing protocols (Ogiela and Ogiela 2010, 2012; Yuan 2014). Obtained secret parts of visual information may be hidden over the communication channel and used for monitoring or authorization procedures in many different application, e.g., electronic currency exchange (Ogiela and Sulkowski 2014), medical images watermarking (Hachaj and Ogiela 2012, 2013; Ogiela and Ogiela 2011) or even preventing information leakage in large computer infrastructures used for homeland security purposes (Ogiela and Ogiela 2014).

References

Bailey K, Curran K (2005) Steganography—The Art of Hiding Information. Booksurge Publishing

Budhia U, Kundur D, Zourntos T (2006) Digital video steganalysis exploiting statistical visibility in the temporal domain. Inf Forensics Secur IEEE Trans 1(4):502–516

Castiglione A, De Santis A, Soriente C (2007) Taking advantages of a disadvantage: Digital forensics and steganography using document metadata. J Syst Softw 80(5):750–764

Castiglione A, De Santis A, Fiore U, Palmieri F (2012) An asynchronous covert channel using spam. Comput Math Appl 63(2):437–447

Castiglione A, D'Alessio B, De Santis A (2011a) Steganography and secure communication on online social networks and online photo sharing. In: Broadband and Wireless Computing, Communication and Applications (BWCCA), International Conference on. pp 363–368

Castiglione A, D'Alessio B, De Santis A, Palmieri F (2011b) New steganographic techniques for the OOXML file format. In: Proceedings of the IFIP WG 8.4/8.9 International Cross Domain Conference on Availability, Reliability and Security for Business, Enterprise and Health Information Systems. Springer, Berlin, Heidelberg, pp 344–358 **(ARES'11)**

Castiglione A, De Santis A, Fiore U, Palmieri F (2011c) E-mail-based covert channels for asynchronous message steganography. In: Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), Fifth International Conference on. pp 503–508

Cheddad A (2009) Digital Image Steganography: Concepts, Algorithms, and Applications. VDM Verlag Dr, Müller

Cox IJ, Miller ML, Bloom J, Fridrich J, Kalker J (2008) Digital watermarking and steganography. Morgan Kaufmann Publishers, Burlington

Fridrich JJ, Goljan M, Hogea D (2003) Steganalysis of JPEG images: breaking the F5 algorithm. Revised Papers from the 5th International Workshop on Information Hiding, IH '02. Springer, London, pp. 310–323

Fridrich J (2010) Steganography in Digital Media: Principles, Algorithms, and Applications. Cambridge University Press, Cambridge

Hachaj T, Ogiela MR (2012) Framework for cognitive analysis of dynamic perfusion computed tomography with visualization of large volumetric data. J Electron Imag 21(4):043017

Hachaj T, Ogiela MR (2013) Application of neural networks in detection of abnormal brain perfusion regions. Neurocomputing 122:33–42

Katzenbeisser S, Petitcolas FA (2000) Information hiding techniques for steganography and digital watermarking. Artech House, Norwood

Nagaraj V, Vijayalakshmi V, Zayaraz G (2013) Color Image Steganography based on Pixel Value Modification Method Using Modulus Function. IERI Procedia 4:17–24

Ogiela MR, Ogiela U (2010) The use of mathematical linguistic methods in creating secret sharing threshold algorithms. Comput Math Appl 60(2):267–271

Ogiela MR, Ogiela U (2012) Linguistic protocols for secure information management and sharing. Comput Math Appl 63(2):564–572

Ogiela L, Ogiela MR (2014) Cognitive Systems and Bio-inspired Computing in Homeland Security. J Netw Comput Appl 38:34–42

Ogiela MR, Sulkowski P (2014) Protocol for irreversible off-line transactions in anonymous electronic currency exchange. Soft Comput 18(12):2587–2594

Ogiela L, Ogiela MR (2011) Semantic analysis processes in advanced pattern understanding systems. In: Kim TH et al (eds) Communications in Computer and Information Science, vol. 195. Springer, Berlin, Heidelberg, pp 26–30

Ong SY, Wong KS, Qi X, Tanaka K (2015) Beyond format-compliant encryption for JPEG image. Signal Process Image Commun 31:47–60

Qazanfari K, Safabakhsh R (2014) A new steganography method which preserves histogram: Generalization of LSB^{++}. Inf Sci 277:90–101

Subhedar MS, Mankar VH (2014) Current status and key issues in image steganography: a survey. Comput Sci Rev 13–14:95–113

Tang M, Hu J, Song W (2014) A high capacity image steganography using multi-layer embedding. Optik 125:3972–3976

Thompson E, Palacios A, Varela FJ (1992) Ways of coloring: comparative color vision as a case study for cognitive science. Behav Brain Sci 15:1–26

Westfeld A (2001) F5–a steganographic algorithm: high capacity despite better steganalysis. In: 4th International Workshop on Information Hiding. Springer, pp 289–302

Yuan H-D (2014) Secret sharing with multi-cover adaptive steganography. Inf Sci 254:197–212