دانشگاه شهید مدنی

مجموعه مقالات

کنفرانس توسعه کاربردهای صنعتی اطلاعات، ارتباطات و محاسبات

Extending Industrial Application of Information, Communications and Computations (EIA-ICC) Conference

دانشگاه شهید مدنی
۱۸ بهمن ماه ۱۳۹۰

فهرست

پوستر مقالات

A Novel Patch-Based Digital Signature

M. Saadatmand-Torzjan
Electrical Engineering Department, Ferdowsi University of Mashhad, Mashhad, Iran

S. Etezad
Electrical Engineering Department, Ferdowsi University of Mashhad, Mashhad, Iran

Abstract:
In this paper a new patch-based digital signature (DS) is suggested. The proposed approach similar to steganography methods hides the secure message in a host image. However, it uses a patch-based key to code/decode the data like cryptography approaches. Both the host image and key patches are randomly initialized. The proposed approach consists of coding and decoding algorithms. The coding algorithm converts the characters stream of the secret message to the patches stream of the DS image. The final image can be further distorted by noise to hide the source patches. Nevertheless, the decoding algorithm uses the connectivity and compactness properties of patches to decode the secret message. Experimental results demonstrated that it is significantly robust against noise. The proposed approach has been successfully used for digital signature generation.

Keywords: Digital Signature, Patch-Based Image Processing, Steganography, Watermarking

1. Introduction

The demand of secure transmission of information has been an issue since the first invented communication methods. Security has a top priority in any organization dealing with confidential data. Whatever is the method we choose for the security purpose, the major issue is the degree of security [1].

The problem of unauthorized copying is of great concern especially to the music, film, book, and, software publishing industries. To overcome this problem, some invisible information can be embedded in the digital media in such a way that it could not be easily extracted without a specialized technique [2]. Information hiding is an emerging research area, which encompasses applications such as copyright protection for digital media, watermarking, steganography, fingerprinting, graphical passwords, digital signature (DS), and copy deterrence of software [3].

For more details, in watermarking, the message, a low-level signal including owner identification and a digital time stamp, is directly placed into the host data set (*e.g.* an image) [4]. The buyer-seller watermarking protocol is a good example of using invisible watermarks for copy deterrence applications [5]. Various techniques have been proposed for public-key watermarking systems, which allow two parties who have never met or exchanged a secret, to send hidden messages over public channels [6, 7].

Steganography, which is also considered as invisible watermarking, imperceptibly hides the secret message within the host data [8, 9]. For example, substituting the message data with the least significant bits of the host data is a well-known frequently-used approach [10-14]. As an example, Aly [11] used the least significant bit method to embed the secret message in the motion vectors of the P- and B-frames in a compressed video.

By using the fingerprint approach, the owner can embed a serial number in the data set. It uniquely indicates the copyright information and thus, any unauthorized use of the data set can be traced [3].

Moreover, one of the most important topics in information security is the user authentication. There is a good security by using the text-based strong password schemes, but memorizing the password is often so difficult and users usually write them down on a piece of paper or save inside the computer. The graphical user authentication (or graphical password), known as an alternative solution, is actually based on this fact that the human tend to remember images better. However, the graphical passwords are vulnerable to five attacks including brute force, dictionary, spyware, shoulder surfing, and social engineering attacks [15].

Besides, by using DS in electronic government, confidentiality, integrity, authenticity, and anti-denial of circulating documents in network environment can be solved [16, 17]. Jarusombat *et al* proposed location based DS on mobile devices which increases the security and also compromises with low-computation capability and limited battery life of these devices [18].

However, with the advent of illegal copying of software and pervasive access to the Internet, software protection has gained increasing importance. Therefore, a complete solution is required to provide the software piracy protection capability. It should allow one to protect his/her software applications against piracy, illegal use, or illegitimate copy [19].

Generally speaking, copy-protection techniques can be separated into token-based and token-less categories. Although token-based techniques are most common, they simply refuse to run software until a specified token is present [20]. However, they can be simply cracked by removing the checks of tokens from the application code while the software itself remains fully functional. In contrast, typical token-less techniques (such as obfuscation, encryption, watermarking, fingerprinting, and so forth) try to authenticate the user.

In this paper, we propose a new DS generation and verification method which can provide a breakthrough in all the mentioned applications. The proposed approach, called PBIS (patch-based image signature), includes coding and decoding stages. First, the proposed coding algorithm (referred to as C-PBIS) decomposes the secret message to its fundamental characters while the DS image is randomly initialized. Then, for each message character, the corresponding patch in the codex (*i.e.* a set of randomly generated primary patches) is copied to a randomly-chosen position of the image. After encryption of some essential coding parameters, whole the image is distorted by additive noise.

software licence agreement, secure communication, and so forth.

Acknowledgements

This work is partially supported by *Binatooth Machine Vision and Image Processing* Co. affiliated to *Khorasan Science and Technology Park*. Besides, it is a part of the project *Intelligent Industrial Camera* with the grant No. 15/13644 of *Ferdowsi University of Mashhad*.

Appendix A

Let's M be a matrix of size $e\times f$ whose components are also matrixes of size $g\times h$. The component located in the rth row and cth column of M is represented by $M_{(r,c)}$. Moreover, the component of pth row and qth column of $M_{(r,c)}$ is also indicated by $M_{(r,c)}(p,q)$.

Furthermore, we can equivalently use the index-based method to uniquely indicate each component of M. In more detail, the ith component of M, shown by $M_{(i)}$ (where i is the index), corresponds to $M_{(r,c)}$, if and only if:

$$i=(c-1)\times e+r$$

(35)

or equivalently:

$$(r,c)=(\text{mod}(i-1,e)+1,\left[\frac{i-1}{e}\right]+1)$$

(36)

where [.] returns the integer fraction of a number and mod(a,b) computes a modulo b. Similarly, the jth component of M_i is given by:

$$M_{(i)}(j)=M_{(r,c)}(\text{mod}(j-1,g),\left[\frac{j-1}{g}\right])$$

(37)

References

[1] S. K. Bandyopadhyay, D.Bhattacharyya, D.Ganguly, S. Mukherjee and P. Das, "A Tutorial Review on Steganography ", in *Proc. 2008 Int'l Conf. Contemporary Computing*.

[2] N. Sharma, J. Bhatia, and N. Gupta, "An encrypto- stego technique based secure data transmission system director," CDAC Mohali, India. [Online] Available: http://www.scribd.com/doc/63422680/encrypto

[3] M. M Amin, M. Salleh. S. Ibrahim, M. R. Katmin, and M. Z. I. Shamsuddin, "Information Hiding using Steganography", *in Proc. 2003 of the 4th National Conf. Telecommunication Technology*, Shah Alam, Malaysia. pp. 21–25.

[4] M. Chandra, S. Pandey, and R. Chaudhary,"Digital Watermarking Technique for Protecting Digital Images", *in Proc. 2010 of the 3rd IEEE Int'l Conf. Computer Science and Information Technology*, vol. 7, pp. 226–233.

[5] N. Memon and P. W. Wong, "A Buyer–Seller Watermarking Protocol", *IEEE Transactions on Image Processing*, vol. 10, no. 4, pp. 643-649. 2001.

[6] Q. Wang, F. Sun, and F. Liu, "Research on Public-Key Digital Watermarking System", *in Proc. 2011 IEEE 3rd Int'l Conf. Communication Software and Networks*, pp. 158–162.

[7] R. Xie, K. Wu, J. Du, and C. Li, "Survey of Public Key Digital Watermarking Systems", *in Proc. 2007 of 8th ACIS Int'l Conf. Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing*, vol. 2, pp. 439–443.

[8] M.-C. Chen, S. S. Agaian, and C. L. P. Chen , "Generalized collage steganography on images", in *Proc. 2008 IEEE Int'l Conf. Systems, Man and Cybernetics*, pp. 1043–1047.

[9] S. Singh and G. Agarwal, "Use of image to secure text message with the help of LSB replacement", *Int'l Journal of Applied Engineering Research Dindigul*, vol. 1, no. 1, pp. 200–205, 2010.

[10] D. Artz, "Digital steganography: hiding data within data", *IEEE Internet Computing*, vol. 5, no. 3, pp. 75–80, 2001.

[11] A. H. Aly, "Data Hiding in Motion Vectors of Compressed Video Based on Their Associated Prediction Error", *IEEE Trans. onInformation Forensics and Security*, vol. 6, no. 1, pp. 14–18, 2011.

[12] K. B. Raja, C.R.Chowdary, K. R. Venugopal, and L. M. Patnaik, "A Secure Image Steganography using LSB, DCTand Compression Techniques on Raw Images", *in Proc. 2005 3rd Int'l Conf. Intelligent Sensing and Information Processing*, pp. 170–176.

[13] M. Nosrati , R. Karimi, H. Nosrati , and A. Nosrati, "Embedding stego-text in cover images using linked list concepts and LSB technique", J. of American Science, vol 7, no 6, pp. 97-100, 2011.

[14] R. Sharma, E.Walia, and D. Sharma, "Analysis of non-adaptive and adaptive edge based LSB steganographyfor colored images", *Int'l Journal of Computing and Business Research*, vol. 2, no. 1, 2011.

[15] A. H. Lashkari, M .Masrom, and A. A.Manaf, "A Secure Recognition Based Graphical Password by Watermarking", *in Proc. 2011 of the 11th IEEE Int'l Conf. Computer and Information Technology*, pp. 164–170.

[16] N. Nikolaidis and I. Pitas, "Copyright Protection of Images Using Robust Digital Signatures", *in Proc. 1996 IEEE Int'l Conf.*

Acoustics, Speech, and Signal Processing, vol. 4, pp.2168–2171.

[17] N. Zhu and G. X. Xiao, "The Application of a Scheme of Digital Signature in Electronic Government", *in Proc. 2008 Int'l Conf. Computer Science and Software Engineering*, vol. 3, pp. 618–621.

[18] S. Jarusombat and S. Kittitornkun, "Digital Signature on Mobile Devices based on Location", *in Proc. 2006 Int'l Symp. Communications and Information Technologies*, pp. 866–870.

[19] S. Mumtaz, S. Iqbal, and E. I. Hameed, "Development of a Methodology for Piracy Protection of Software Installations", *in Proc. 2005 of the 9th Int'l Multitopic Conference*, pp. 1–7.

[20] P. Djekicand and C. Loebbecke, "Software Piracy Prevention through Digital Rights Management Systems", *in Proc. 2005 of the 7th IEEE Int'l Conf. E-Commerce Technology*, pp. 504–507.

[21] S. Theodoridis, K. Koutroumbas, *Pattern Recognition*. Elsecier Academic Press, 2nd ed., 2003.

[22] D. Y. Downham and F. D. K. Roberts, "Multiplicative congruential pseudo-random number generators," *The Computer Journal*, vol. 10, no. 1, pp. 74–77, 1967. doi:10.1093/comjnl/10.1.74.

دانشگاه تربیت معلم آذربایجان

اولین کنفرانس ملی

توسعه کاربردهای صنعتی

اطلاعات، ارتباطات و محاسبات

18 بهمن ماه 1390

کنفرانس کاربردهای
اطلاعات، ارتباطات و محاسبات

پژوهشگر گرامی، صنعتگر ارجمند،

جناب آقای مهدی سعادتمند

با سلام و احترام،

دبیرخانه کنفرانس در کمال مسرت و افتخار به اطلاع گرامی می‌رساند مقاله شما تحت عنوان

A Novel Patch-Based Digital Signature

جهت سخنرانی شفاهی در کنفرانس پذیرفته شده است.

جهت تکمیل موارد ثبت نام، ارائه مقاله، چاپ کامل مقاله در کتابچه مقالات کنفرانس، صرف نهار و پذیرایی، انجام موارد ذیل حداکثر تا تاریخ 1390/11/12 ضروری می‌باشد:

1- **ارسال نسخه نهایی مقاله مطابق با فرمت مقاله کنفرانس**

2- **ارسال تصویر فیش بانکی ثبت نام به ایمیل** eiaicc@azaruniv.edu

دبیر علمی کنفرانس

دکتر محسن حیدریان

A Novel Patch-Based Digital Signature

M. Saadatmand-Torzjan*, S. Etezad**
* Electrical Engineering Department, Ferdowsi University of Mashhad, Mashhad, Iran,
saadatmand@kiaeee.org
** Electrical Engineering Department, Ferdowsi University of Mashhad, Mashhad, Iran,
Sahba.etezad@gmail.com

Abstract: *In this paper a new patch-based digital signature (DS) is suggested. The proposed approach similar to steganography methods hides the secure message in a host image. However, it uses a patch-based key to code/decode the data like cryptography approaches. Both the host image and key patches are randomly initialized. The proposed approach consists of coding and decoding algorithms. The coding algorithm converts the characters stream of the secret message to the patches stream of the DS image. The final image can be further distorted by noise to hide the source patches. Nevertheless, the decoding algorithm uses the connectivity and compactness properties of patches to decode the secret message. Experimental results demonstrated that it is significantly robust against noise. The proposed approach has been successfully used for digital signature generation.*

Keywords: Digital Signature, Patch-Based Image Processing, Steganography, Watermarking

1. Introduction

The demand of secure transmission of information has been an issue since the first invented communication methods. Security has a top priority in any organization dealing with confidential data. Whatever is the method we choose for the security purpose, the major issue is the degree of security [1].

The problem of unauthorized copying is of great concern especially to the music, film, book, and, software publishing industries. To overcome this problem, some invisible information can be embedded in the digital media in such a way that it could not be easily extracted without a specialized technique [2]. Information hiding is an emerging research area, which encompasses applications such as copyright protection for digital media, watermarking, steganography, fingerprinting, graphical passwords, digital signature (DS), and copy deterrence of software [3].

For more details, in watermarking, the message, a low-level signal including owner identification and a digital time stamp, is directly placed into the host data set (*e.g.* an image) [4]. The buyer-seller watermarking protocol is a good example of using invisible watermarks for copy deterrence applications [5]. Various techniques have been proposed for public-key watermarking systems, which allow two parties who have never met or exchanged a secret, to send hidden messages over public channels [6, 7].

Steganography, which is also considered as invisible watermarking, imperceptibly hides the secret message within the host data [8, 9]. For example, substituting the message data with the least significant bits of the host data is a well-known frequently-used approach [10-14]. As an example, Aly [11] used the least significant bit method to embed the secret message in the motion vectors of the P- and B-frames in a compressed video.

By using the fingerprint approach, the owner can embed a serial number in the data set. It uniquely indicates the copyright information and thus, any unauthorized use of the data set can be traced [3].

Moreover, one of the most important topics in information security is the user authentication. There is a good security by using the text-based strong password schemes, but memorizing the password is often so difficult and users usually write them down on a piece of paper or save inside the computer. The graphical user authentication (or graphical password), known as an alternative solution, is actually based on this fact that the human tend to remember images better. However, the graphical passwords are vulnerable to five attacks including brute force, dictionary, spyware, shoulder surfing, and social engineering attacks [15].

Besides, by using DS in electronic government, confidentiality, integrity, authenticity, and anti-denial of circulating documents in network environment can be solved [16, 17]. Jarusombat *et al* proposed location based DS on mobile devices which increases the security and also compromises with low-computation capability and limited battery life of these devices [18].

However, with the advent of illegal copying of software and pervasive access to the Internet, software protection has gained increasing importance. Therefore, a complete solution is required to provide the software piracy protection capability. It should allow one to protect his/her software applications against piracy, illegal use, or illegitimate copy [19].

Generally speaking, copy-protection techniques can be separated into token-based and token-less categories. Although token-based techniques are most common, they simply refuse to run software until a specified token is present [20]. However, they can be simply cracked by removing the checks of tokens from the application code while the software itself remains fully functional. In contrast, typical token-less techniques (such as obfuscation, encryption, watermarking, fingerprinting, and so forth) try to authenticate the user.

In this paper, we propose a new DS generation and verification method which can provide a breakthrough in all the mentioned applications. The proposed approach, called PBIS (patch-based image signature), includes coding and decoding stages. First, the proposed coding algorithm (referred to as C-PBIS) decomposes the secret message to its fundamental characters while the DS image is randomly initialized. Then, for each message character, the corresponding patch in the codex (*i.e.* a set of randomly generated primary patches) is copied to a randomly-chosen position of the image. After encryption of some essential coding parameters, whole the image is distorted by additive noise. Finally, the DS image is reshaped to hide its original size from the user.

In the suggested decoding algorithm (referred to as D-PBIS), after decryption of the coding parameters, for each patch in the image, the corresponding character is specified by using the nearest neighbor measure.

Experimental results demonstrated significant robustness of PBIS against noise. Furthermore, the computational burden of both the coding and decoding algorithms is small. They are easy and straightforward to implement. Therefore, PBIS can be used in a wide variety of security applications such as user authentication, software license agreement, and so forth.

This paper is organized as follows: the proposed algorithm is explained in Section 2. Experimental results are given in Section 3. Finally, Section 4 is devoted to concluding remarks.

Notations used in this paper are fairly standard. Both sets and matrixes are represented by upper-case letters while the later can be distinguished by italicface symbols. Boldface symbols are also used for vectors. For more details about the index-based representation of matrix/set components, see Appendix A.

2. Digital Image Signature

Both C- and D-PBIS, require some primary information including the symbols set (S), codex set (P), and permitted movements matrixes along *x*-axis (D^x) and *y*-axis (D^y) to properly work.

In more detail, the symbols set S, given by the DS images provider (so-called the security administrator), includes all the characters necessary to write the text-based secret message such as alphabets, digits, space, dash, under-line, comma, and so forth, as follows:

$$\mathrm{S}=\left\{\mathrm{S}_{(k)}\middle| k\in \mathrm{Z}, 0<k\leq L_{\mathrm{S}}\right\} \tag{1}$$

where $\mathrm{S}_{(k)}$ represents the *k*th component of S and $L_{\mathrm{S}}=\|\mathrm{S}\|$ in which ||.|| returns the cardinality of a set or number of components within a matrix.

The matrixes D_x (L_x=||D_x||) and D_y (L_y=||D_y||) are used to locate each coding patch in the DS image with respect to the previous patch position.

The codex is a set of $L_{\tilde{\mathrm{S}}}L_xL_y+1$ randomly-initialized matrixes of size *w*×*w* (so-called patches) with three-dimensional vectorial components. Therefore, we can write:

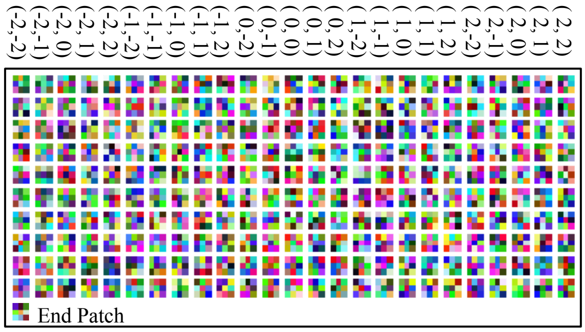

Fig. 1. A sample codex: each row corresponds to a digit and every column matches to a doublet (Δ*x*,Δ*y*). The single patch in the last row is the end-patch.

$$\mathrm{P}=\left\{P_{(i)}\middle| i\in \mathrm{Z}, 0<i\leq L_{\mathrm{S}}L_xL_y+1\right\} \tag{2}$$

where $P_{(i)}$ is the *i*th patch of the codex. Specifically, the last patch of P (called the end-patch) will be used to indicate the end of a patch stream (see Section 2.1.2). For example, a sample codex, including 10×5×5+1 patches, with $\mathrm{S}=\{0,1,...,9\}$ and D_x=D_y=[-2,-1,0,1,2], is illustrated in Fig. 1.

Excluding the end-patch, P can be simplified to a matrix of size L_{S}×L_xL_y, as follows:

$$P=\begin{bmatrix} P_{(1,1)} & P_{(1,2)} & \cdots & P_{(1,L_xL_y)} \\ P_{(2,1)} & P_{(2,2)} & \cdots & P_{(2,L_xL_y)} \\ \vdots & \vdots & & \vdots \\ P_{(L_{\mathrm{S}},1)} & P_{(L_{\mathrm{S}},2)} & \cdots & P_{(L_{\mathrm{S}},L_xL_y)} \end{bmatrix} \tag{3}$$

where

$$P_{(p,q)}=P_{((q-1)L_{\mathrm{S}}+p)} \tag{4}$$

Furthermore, the patch $P_{(p,q)}$ corresponds to the triplet ($\mathrm{S}_{(k)}$,Δ*x*,Δ*y*) only when *p*=*k* and,

$$q=L_y(\Delta x-1)+\Delta y \tag{5}$$

or equivalently:

$$\begin{cases} \Delta x=\left[\dfrac{q-1}{L_y}\right]+1 \\ \Delta y=\mathrm{mod}(q-1,L_y)+1 \end{cases} \tag{6}$$

where $\Delta x\in D_x$ and $\Delta y\in D_y$. For example, in Fig. 1, $P_{(2,9)}$ corresponds to ($\mathrm{S}_{(2)}$=1,Δ*x*=-1, Δ*y*=1).

2.1 Coding Algorithm

Fig. 2 shows the block diagram of the proposed coding algorithm. C-PBIS treats the secret message as a stream of consequence characters. Indeed, it converts this characters stream to a patches stream within the DS image as follows.

2.1.1 Initialization

First, all pixels of the DS image (*I*) are randomly initialized in the range [0,255]. Indeed, *I* is an image of size *b*×*αb*×3 which can be considered as a matrix of size *b*×*αb* with three-dimensional vectorial components. The parameters *b* and *α* are integer constants, separately specified for each secret message.

The secret message can be decomposed to its fundamental characters, as follows:

$$\mathrm{C}=\{\mathrm{C}_{(k)}| 1\leq k\leq L_{\mathrm{C}}\}\subset \mathrm{S} \tag{7}$$

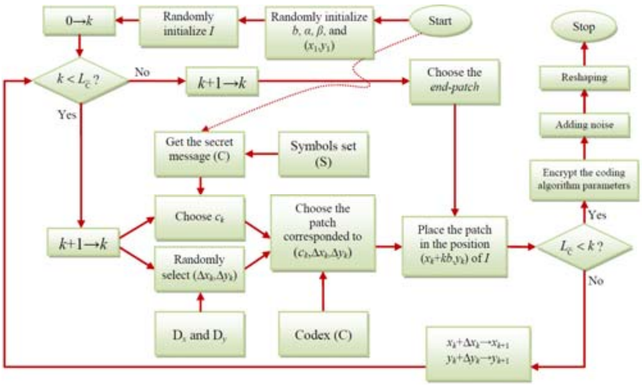
Start
Randomly initialize b, α, β, and (x_1,y_1)
Randomly initialize I
$0\rightarrow k$
$k < L_{\mathrm{C}}$?
No
Yes
$k+1\rightarrow k$
Choose the end-patch
Get the secret message (C)
Symbols set (S)
Choose c_k
Randomly select $(\Delta x_k,\Delta y_k)$
Choose the patch corresponded to $(c_k,\Delta x_k,\Delta y_k)$
$k+1\rightarrow k$
D$_x$ and D$_y$
Codex (C)
Place the patch in the position (x_k+kb,y_k) of I
$L_{\mathrm{C}} < k$?
Yes
No
Encrypt the coding algorithm parameters
Adding noise
Reshaping
Stop
$x_k+\Delta x_k\rightarrow x_{k+1}$
$y_k+\Delta y_k\rightarrow y_{k+1}$

Fig. 2. The block diagram of C-PBIS

where C is the set of message characters, C$_{(k)}$ indicates its kth component, and $L_{\mathrm{C}} = \| \mathrm{C} \|$.

As shown in Fig. 3, I is divided into α square blocks (including B_0, B_1, …, $B_{\alpha\text{-}1}$), all of size b×b. Except the first block (B_0) which is reserved to code C-PBIS parameters (see Section 2.1.3), every remaining block can include only one patch. Thus, to preserve the possibility of using all components of D_x and D_y for localization of patches (see Section 2.1.2), we should have:

$$w + \max\left(\max_{\Delta x \in D_x}(|\Delta x|), \max_{\Delta y \in D_y}(|\Delta y|) \right) \le b \tag{8}$$

where |.| computes the first-order norm (so-called Manhattan distance measure) [21].

Besides, the patches stream should be finished by the end-patch. Thus, α should observe the following constraint:

$$L_{\tilde{\mathrm{C}}} + 2 \le \alpha \tag{9}$$

2.1.2 Patches Stream

Then, the integer parameters x_1 and y_1 are randomly initialized in the range [1,b-w] as follows (Let's k=1):

$$1 \le x_1, y_1 \le b - w \tag{10}$$

They are used to locate the first stream patch in B_1. Generally, in the kth step ($1 \le k \le L_{\tilde{\mathrm{C}}} + 1$), the position of the top-left corner of the current patch within B_k is specified by (x_k,y_k).

To increase the security of the coding algorithm, every (x_k,y_k) should be randomly initialized. For achieving this purpose, the parameters Δx_k and Δy_k are randomly selected from D_x and D_y, respectively, with observing the following constraints:

$$\begin{cases} -x_k < \Delta x_k \le b - w - x_k \\ -y_k < \Delta y_k \le b - w - y_k \end{cases} \tag{11}$$

In each step of the coding algorithm, (Δx_k,Δy_k) is provided to specify the position of the next patch in the image.

Next, if all message characters have been already coded in the image (*i.e.* $L_{\mathrm{C}} \le k$), the end-patch is selected to specify the end of the stream. Otherwise, the patch corresponded to the triplet (c_k,Δx_k,Δy_k) is chosen to specify both the kth message character and the position of the next patch.

Subsequently, the selected patch is copied to the corresponding components of B_k as follows:

$$\boldsymbol{B}_k(i,j) = \boldsymbol{P}_{(p, L_y(\Delta x_k - 1) + \Delta y_k)}(i,j), \qquad 1 \le i, j \le w \tag{12}$$

such that,

$$\boldsymbol{B}_k(i,j) = \boldsymbol{I}(x_k + kb + i - 1, y_k + j - 1) \tag{13}$$

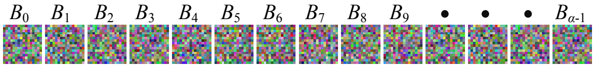

Fig. 3. The DS image I is divided into α blocks, each of size b×b.

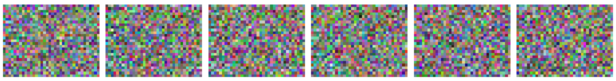

Fig. 4. Six DS images provided for the secret message 0123456789 with the codex of Fig. 1 and η=100%, all of size 36×27.

For example, according to the above equations, the component located on the top-left corner of the kth patch is copied to the position (x_k+kb,y_k) of I, equalivalent to the position (x_k,y_k) of B_k.

Afterwards, the position of the next patch is explicitly computed as follows:

$$\begin{cases} x_{k+1} = x_k + \Delta x_k \\ y_{k+1} = y_k + \Delta y_k \end{cases} \tag{14}$$

Both x_{k+1} and y_{k+1} are also in the range [1,b-w], because of observing the constraints of Eq. (11) by Δx_k and Δy_k.

After increasing k by one, the above process is repeated until the patch stream finishes with the end-patch.

2.1.3 Coding Algorithm Parameters

Then, some essential parameters of C-PBIS including b, x_1, and y_1 should be encrypted within B_0, because they separately alter for each secrete message. For this aim, we take advantage of the index-based representation (see Appendix A) for components of P and I.

In more detail, for encrypting b, x_1, and y_1, whole components of the patches P_b, P_{x_1}, and P_{y_1}, respectively, are consecutively copied to the corresponding components of B_0 as follows:

$$\boldsymbol{B}_0(i) = \begin{cases} \boldsymbol{P}_b\left(\mathrm{mod}(i-1, w^2)+1\right) & 0 < i \le w^2 \\ \boldsymbol{P}_{x_1}\left(\mathrm{mod}(i-1, w^2)+1\right) & w^2 < i \le 2w^2 \\ \boldsymbol{P}_{y_1}\left(\mathrm{mod}(i-1, w^2)+1\right) & 2w^2 < i \le 3w^2 \end{cases} \tag{15}$$

For the sake of importance of coding parameters, the above copy procedure is repeated for η (1≤η) times. Thus, we should have:

$$3\eta w^2 \le b^2 \Rightarrow \sqrt{3\eta}\, w \le b \tag{16}$$

Combining the above equation with Eq. (8), we can write:

$$w + \max\left(\max_{\Delta x \in D_x}(|\Delta x|), \max_{\Delta y \in D_y}(|\Delta y|), (\sqrt{3\eta} - 1)w \right) \le b \tag{17}$$

This equation provides a lower limit for the height of I.

2.1.4 Distortion by Noise

Next, I is altered by additive noise as follows:

$$\hat{\boldsymbol{I}}(i) = \boldsymbol{I}(i) + \gamma \mathrm{rand}(), \qquad 1 \le i \le \alpha b^2 \tag{18}$$

where the function rand() randomly generates a three-dimensional vector (with scalar components in the range [0,255]) based on the multiplicative congruential approach [22]. The coefficient γ also indicates the strength of noise. As demonstrated in experimental results (see Section 3.2), D-PBIS is fairly robust against noise. Therefore, for increasing the security of the algorithm, in this step, I is distorted by noise to hide source patches from the user.

2.1.5 Image Reshaping

By copying each component of $\hat{I}$ to the corresponding one with the same index, the final DS image $\bar{I}$ of size $b\beta \times \alpha b/\beta$ is produced as follows:

$$\bar{\boldsymbol{I}}(i) = \hat{\boldsymbol{I}}(i), \qquad 1 \le i \le \alpha b^2 \tag{19}$$

Here, β is a randomly-chosen divisor of α as follows:

$$\beta \in \{ n \in \mathrm{N} | \, 1 < n, \ \mathrm{mod}(\alpha, n) = 0 \} \tag{20}$$

The final image is reshaped to hide its true width (*i.e.* b) from the user and in turn, improve security of the digital signature.

2.1.6 PBIS Key

The primary parameters of the proposed algorithm including P, S, D_x, D_y, w, and η should be specified by the security administrator. Both C-PBIS and D-PBIS should use these parameters with exactly the same values to correctly work. At first glance, it may seem like a constraint, but that can provide a secure method to separate DS images produced by different administrators. In other words, the above primary parameters totally form the encryption key of PBIS. To correctly decode the DS image, this key should be entirely known.

Similar to watermarking and steganography methods, PBIS hides the secure message in a host image. However, it provides a coding key similar to cryptography algorithms.

Anyway, for its very long length, the PBIS key is significantly complex. In more detail, the main part of the key is the codex. It consists of $L_{\mathrm{S}} L_x L_y + 1$ patches. Each patch includes w^2 three-dimensional vectorial components whose elements range between 0 and 255. Therefore, the number of all possible codexes is given by:

$$N(\mathrm{P}) = 256^{3w^2(L_{\mathrm{S}} L_x L_y + 1)} \tag{21}$$

For example, when $L_{\mathrm{S}} = 10$ (*e.g.* once S includes only digits), L_x=L_y=3 (*e.g.* for D_x=D_y=[-1,0,1]), and w=3, we have $N(\mathrm{P}) = 2^{19656} \approx 1.11 \times 10^{5917}$. If a computer evaluates each codex in one nanosecond, it takes more than 2.77×10^{5901} years to check all possible cases.

Furthermore, C-PBIS uses a number of parameters, including b, α, β, and (x_1,y_1), which should be separately initialized for each secret message, with observing Eqs. (17), (9), (20), and (11), respectively (see Section 2.1.3). Therefore, as illustrated in Fig. 4, even for the same secret message, each time, C-PBIS provides a different DS image because of randomly initialization of the DS image, using different values for the above parameters, and altering the final DS image with noise.

2.2 Decoding Algorithm

Once the PBIS key is given, decoding of the DS image is straightforward. Generally, for the candidate patch $\bar{P}$, the most similar patch in the codex is given by using the first-order norm as follows:

$$m = \mathrm{index}\left(\min_{j} \left(\mathop{\mathrm{sum}}_{i=1}^{w^2} (|\bar{\boldsymbol{P}}(i) - \boldsymbol{P}_j(i)|) \right) \right) \tag{22}$$

Although we take advantage of the above similarity function for its simplicity, higher order norms or cross-correlation measure, also, can be alternatively used.

2.2.1 Initialization

Initially, to decrypt the exclusive coding parameters, including b, α, x_1, and y_1, the first $3\eta w^2$ components of $\bar{I}$ (*i.e.* $\bar{\boldsymbol{I}}(1)$ to $\bar{\boldsymbol{I}}(3\eta w^2)$) are divided into 3η patches (including $\bar{P}_1$, $\bar{P}_2$, …, and $\bar{P}_{3\eta}$) such that each consists of w^2 components. According to Section 2.1.3, all the patches $\bar{P}_1$, $\bar{P}_4$, …, $\bar{P}_{3\eta-2}$ correspond to P_b. Thus, b can be obtained as the median of indexes of all matched patches as follows:

$$b = \mathop{\mathrm{median}}_{k=1}^{\eta}\left(\mathrm{index}\left(\min_{j} \left(\mathop{\mathrm{sum}}_{i=1}^{w^2} (|\bar{\boldsymbol{P}}_{3k-2}(i) - \boldsymbol{P}_j(i)|) \right) \right) \right) \tag{23}$$

Similarly, we have:

$$x_1 = \mathop{\mathrm{median}}_{k=1}^{\eta}\left(\mathrm{index}\left(\min_{j} \left(\mathop{\mathrm{sum}}_{i=1}^{w^2} (|\bar{\boldsymbol{P}}_{3k-1}(i) - \boldsymbol{P}_j(i)|) \right) \right) \right) \tag{24}$$

$$y_1 = \mathop{\mathrm{median}}_{k=1}^{\eta}\left(\mathrm{index}\left(\min_{j} \left(\mathop{\mathrm{sum}}_{i=1}^{w^2} (|\bar{\boldsymbol{P}}_{3k}(i) - \boldsymbol{P}_j(i)|) \right) \right) \right) \tag{25}$$

Obviously, α is given by:

$$\alpha = \|\bar{I}\| / b \tag{26}$$

Therefore, $\hat{I}$ can be obtained by reshaping $\bar{I}$ to a matrix of size b×αb, according to Eq. (19). It is worth to note that there is no way to obtain I from $\hat{I}$, because of distortion by unknown additive noise. Instead, we can decrypt the secret message through the similarity measure of Eq. (22).

2.2.2 Characters Stream

Similar to the approach used in C-PBIS, we should convert the patch stream of the DS image to the characters stream of the secret message.

Generally, in the kth step (initially, set k=1) of the decoding algorithm, the position of the top-left corner of the current patch, of size $w \times w$, in $\hat{I}$ (shown by $\hat{P}_k$) is given by (kb+x_k,y_k). In this case, the kth character of the secret message can be specified by obtaining the most similar patch in the codex as follows:

$$(p_k, q_k) = \mathrm{index}\left(\min_{(p,q)} \left(\mathop{\mathrm{sum}}_{i=1}^{w^2} (|\hat{\boldsymbol{P}}_k(i) - \boldsymbol{P}_{(p,q)}(i)|) \right) \right) \tag{27}$$

Obviously, S_{p_k} specifies the kth character of the secret message. Furthermore, according to Eq. (6), to obtain the position of the next patch in $\hat{I}$, we should have:

$$(x_{k+1}, y_{k+1}) = (\left\lceil \frac{q_k - 1}{L_y} \right\rceil + 1, \mathrm{mod}(q_k - 1, L_y) + 1) \tag{28}$$

After augmenting k by one, the above process is repeated until the patch stream finishes with the end-patch.

3. Experimental Results

The performance of DIS was evaluated using three different codexes with w=3, 5, and 7. All the experimental results were obtained by an Intel Core2Duo 2.0-GHz Laptop with 3-GB main memory using MATLAB environment.

3.1 Producing the Codex

In all experimental evaluations, the symbols set included only digits and D_x=D_y=[-2,-1,0,1,2] Consequently, each codex includes 251 patches.

Furthermore, in each codex, the patches should be as diverse as possible to improve the performance of D-PBIS against noise. In other words, more dissimilar patches can be better distinguished from each other.

Therefore, we provided a simple and effective algorithm to produce diverse patches. In more detail, after initialization of the first patch, a new one is randomly produced. The patch will be added to the codex, if it observes the following constraint:

$$\theta < \min_{j=1}^{k-1}\left(\frac{1}{w^2}\operatorname{sum}_{i=1}^{w^2}\left(\left|\boldsymbol{P}_k(i)-\boldsymbol{P}_j(i)\right|\right)\right) \tag{29}$$

where θ is the threshold of minimum acceptable difference between patches. Otherwise, a new patch is randomly generated. The above procedure repeated until all required patches are initialized.

We experimentally obtained θ=62. By using the above approach, three sets of patches with w=3, 5, and 7 were made.

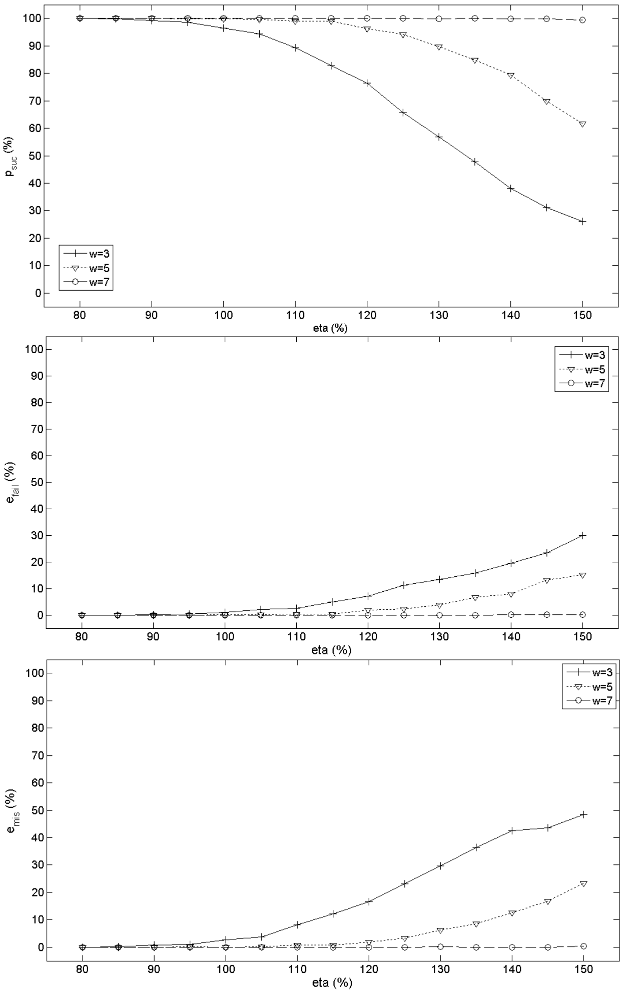

Fig. 5. Variations of (top) p_{suc}, (centre) e_{fail}, and (bottom) e_{mis} versus η for different values of w.

3.2 Robustness against Noise

The proposed algorithm is significantly secure, because of:

- Using very long key
- Randomly initialization of some parameters of C-PBIS for each secure message
- Distortion of the DS image by additive noise

Indeed, C-PBIS produces different DS images for even the same secure message. This may be the most important and unparalleled property of the proposed algorithm. For example, Fig. 4 illustrates six different DS images provided for the secure code 0123456789 with the codex of Fig. 1 and η=100%.

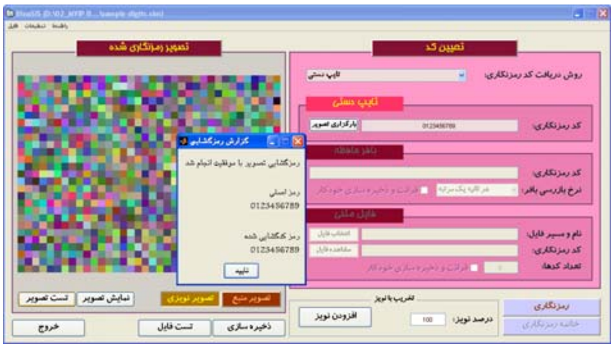

Fig. 6. The developed software for PBIS (GUI texts are in Farsi).

Furthermore, the statistical analysis methods are not useful to discover the PBIS key because; both the host image and coding patches are generated by using the same random-number generation method. Moreover, it is also employed to distort the final DS image.

On the other hand, although patch-based image processing methods are well-known and frequently-used, PBIS can be counted as the first patch-based algorithm to generate digital signatures. D-PBIS takes advantage of compactness and connectivity of patches to decode the DS image. Thus, it should properly work even for an altered DS image.

To test performance of D-DIS against noise, three different codex sets with w=3, 5, and 7 were employed. D_x, D_y and S were also chosen as the same as those used in Fig. 1. η was increased by the step of 5% ranging from 80% to 150%. In each step, the same secure code (*i.e.* 123456789) was coded and then decoded for 1000 times. In decoding of the secure message, three situations may take place:

- The secure message is successfully decoded.
- The decoding algorithm fails because of wrong encryption of the coding parameters.
- Some characters of the decrypted message mismatch with those of the source message.

In order to evaluate the performance of D-PBIS, the following evaluation measures were employed:

$$p_{suc}(w,\eta)=\frac{\text{Number of successes}}{1000}\times 100\% \tag{30}$$

$$e_{fail}(w,\eta)=\frac{\text{Number of failures}}{1000}\times 100\% \tag{31}$$

$$e_{mis}(w,\eta)=\frac{\text{Number of mismatches}}{1000}\times 100\% \tag{32}$$

$$e_{mis}^{k}(w,\eta)=\frac{\text{Number of mismatches of the }k\text{th digit}}{1000}\times 100\% \tag{33}$$

where p_{suc}, e_{fail}, e_{mis}, and e_{mis}^{k} computes the percents of successful decoding attempts, decoding failures, whole message mismatches, and mismatches of the kth character, respectively. Obviously, we have:

$$p_{suc}(w,\eta)=100\%-e_{fail}(w,\eta)-e_{mis}(w,\eta) \tag{34}$$

$$e_{mis}(w,\eta)=\sum_{k=1}^{L_{\mathrm{C}}} e_{mis}^{k}(w,\eta) \tag{35}$$

Fig. 5 shows variations of p_{suc}, e_{fail}, and e_{mis} versus η for different values of w. For all values of w with η≤80%, D-PBIS was quite successful to decode all DS images (p_{suc}=100%). Although p_{suc} decreases by increasing η, its sensitivity to noise significantly improved by augmenting w; such that for η=150%, we have p_{suc}=99.4% with w=7.

For every values of η and w, e_{mis} was approximately 2 times greater than e_{fail}. In other words, from every three unsuccessful decoding attempts, D-PBIS failed only for once.

As demonstrated, D-PBIS may sometimes fail to decode the distorted DS image. Therefore, after finishing C-PBIS, it is necessary to check correct decoding of the image. In this case, if D-PBIS fails, a new DS image should be produced. Fig. 6 illustrates the developed software for PBIS.

4. Conclusion

In this paper, a new patch-based algorithm for digital signature generation and verification was proposed. It uses a set of randomly generated patches to code a set of character symbols. Indeed, C-PBIS converts the characters stream of the secret message to the patches stream of the DS image. C-DIS uses a very long key which considerably improves the security of provided DS images. Besides, both the host image and codex patches are initialized by using the same random-number generator method. It is also employed to further distort the final DS image. For all these reasons, the proposed algorithm is very secure against statistical analysing methods.

D-PBIS decodes the DS image by using a simple similarity measure. Actually, it takes advantage of the compactness and connectivity of patches. For this reason, D-PBIS was significantly robust against noise. Experimental results demonstrated that D-PBIS successfully decoded 99.4% of DS images generated by C-PBIS (with w=7 and η=150%).

Although DIS is originally proposed for digital signature applications, it can be used for a wide variety of security applications such as user authentication, software licence agreement, secure communication, and so forth.

Acknowledgements

This work is partially supported by *Binatooth Machine Vision and Image Processing* Co. affiliated to *Khorasan Science and Technology Park.* Besides, it is a part of the project *Intelligent Industrial Camera* with the grant No. 15/13644 of *Ferdowsi University of Mashhad.*

Appendix A

Let's M be a matrix of size $e\times f$ whose components are also matrixes of size $g\times h$. The component located in the rth row and cth column of M is represented by $M_{(r,c)}$. Moreover, the component of pth row and qth column of $M_{(r,c)}$ is also indicated by $M_{(r,c)}(p,q)$.

Furthermore, we can equivalently use the index-based method to uniquely indicate each component of M. In more detail, the ith component of M, shown by $M_{(i)}$ (where i is the index), corresponds to $M_{(r,c)}$, if and only if:

$$i=(c-1)\times e+r \qquad (35)$$

or equivalently:

$$(r,c)=(\mathrm{mod}(i-1,e)+1,\left[\frac{i-1}{e}\right]+1) \qquad (36)$$

where [.] returns the integer fraction of a number and mod(a,b) computes a modulo b. Similarly, the jth component of M_i is given by:

$$M_{(i)}(j)=M_{(r,c)}(\mathrm{mod}(j-1,g),\left[\frac{j-1}{g}\right]) \qquad (37)$$

References

[1] S. K. Bandyopadhyay, D.Bhattacharyya, D.Ganguly, S. Mukherjee and P. Das, "A Tutorial Review on Steganography ", in *Proc. 2008 Int'l Conf. Contemporary Computing*.

[2] N. Sharma, J. Bhatia, and N. Gupta, "An encrypto- stego technique based secure data transmission system director," CDAC Mohali, India. [Online] Available: http://www.scribd.com/doc/63422680/encrypto

[3] M. M Amin, M. Salleh, S. Ibrahim, M. R. Katmin, and M. Z. I. Shamsuddin, "Information Hiding using Steganography", *in Proc. 2003 of the 4th National Conf. Telecommunication Technology*, Shah Alam, Malaysia, pp. 21–25.

[4] M. Chandra, S. Pandey, and R. Chaudhary,"Digital Watermarking Technique for Protecting Digital Images", *in Proc. 2010 of the 3rd IEEE Int'l Conf. Computer Science and Information Technology*, vol. 7, pp. 226–233.

[5] N. Memon and P. W. Wong, "A Buyer–Seller Watermarking Protocol", *IEEE Transactions on Image Processing*, vol. 10, no. 4, pp. 643-649, 2001.

[6] Q. Wang, F. Sun, and F. Liu, "Research on Public-Key Digital Watermarking System", *in Proc. 2011 IEEE 3rd Int'l Conf. Communication Software and Networks*, pp. 158–162.

[7] R. Xie, K. Wu, J. Du, and C. Li, "Survey of Public Key Digital Watermarking Systems", *in Proc. 2007 of 8th ACIS Int'l Conf. Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing*, vol. 2, pp. 439–443.

[8] M.-C. Chen, S. S. Agaian, and C. L. P. Chen , "Generalized collage steganography on images", in *Proc. 2008 IEEE Int'l Conf. Systems, Man and Cybernetics*, pp. 1043–1047.

[9] S. Singh and G. Agarwal, "Use of image to secure text message with the help of LSB replacement", *Int'l Journal of Applied Engineering Research Dindigul*, vol. 1, no. 1, pp. 200–205, 2010.

[10] D. Artz, "Digital steganography: hiding data within data", *IEEE Internet Computing*, vol. 5, no. 3, pp. 75–80, 2001.

[11] A. H. Aly, "Data Hiding in Motion Vectors of Compressed Video Based on Their Associated Prediction Error", *IEEE Trans. onInformation Forensics and Security*, vol. 6, no. 1, pp. 14–18, 2011.

[12] K. B. Raja, C.R.Chowdary, K. R. Venugopal, and L. M. Patnaik, "A Secure Image Steganography using LSB, DCTand Compression Techniques on Raw Images", *in Proc. 2005 3rd Int'l Conf. Intelligent Sensing and Information Processing*, pp. 170–176.

[13] M. Nosrati , R. Karimi, H. Nosrati , and A. Nosrati, "Embedding stego-text in cover images using linked list concepts and LSB technique", J. of American Science, vol 7, no 6, pp. 97-100, 2011.

[14] R. Sharma, E.Walia, and D. Sharma, "Analysis of non-adaptive and adaptive edge based LSB steganographyfor colored images", *Int'l Journal of Computing and Business Research*, vol. 2, no. 1, 2011.

[15] A. H. Lashkari, M .Masrom, and A. A.Manaf, "A Secure Recognition Based Graphical Password by Watermarking", *in Proc. 2011 of the 11th IEEE Int'l Conf. Computer and Information Technology*, pp. 164–170.

[16] N. Nikolaidis and I. Pitas, "Copyright Protection of Images Using Robust Digital Signatures", *in Proc. 1996 IEEE Int'l Conf. Acoustics, Speech, and Signal Processing*, vol. 4, pp.2168–2171.

[17] N. Zhu and G. X. Xiao, "The Application of a Scheme of Digital Signature in Electronic Government", *in Proc. 2008 Int'l Conf. Computer Science and Software Engineering*, vol. 3, pp. 618–621.

[18] S. Jarusombat and S. Kittitornkun, "Digital Signature on Mobile Devices based on Location", *in Proc. 2006 Int'l Symp. Communications and Information Technologies*, pp. 866–870.

[19] S. Mumtaz, S. Iqbal, and E. I. Hameed, "Development of a Methodology for Piracy Protection of Software Installations", *in Proc. 2005 of the 9th Int'l Multitopic Conference*, pp. 1–7.

[20] P. Djekicand and C. Loebbecke, "Software Piracy Prevention through Digital Rights Management Systems", *in Proc. 2005 of the 7th IEEE Int'l Conf. E-Commerce Technology*, pp. 504–507.

[21] S. Theodoridis, K. Koutroumbas, *Pattern Recognition*. Elsecier Academic Press, 2nd ed., 2003.

[22] D. Y. Downham and F. D. K. Roberts, "Multiplicative congruential pseudo-random number generators," *The Computer Journal*, vol. 10, no. 1, pp. 74–77, 1967. doi:10.1093/comjnl/10.1.74.