Security analysis of optical encryption

Yann Frauel,[a*] Albertina Castro,[b] Thomas J. Naughton,[c] Bahram Javidi [d]
[a] Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas, UNAM, Mexico.
[b] Instituto Nacional de Astrofísica, Óptica y Electrónica, Puebla, Mexico.
[c] Department of Computer Science, National University of Ireland, Maynooth, Ireland.
[d] Electrical and Computer Engineering Department, University of Connecticut, USA.

ABSTRACT

This paper analyzes the security of amplitude encoding for double random phase encryption. We describe several types of attack. The system is found to be resistant to brute-force attacks but vulnerable to chosen and known plaintext attacks.

Keywords: optical encryption, double random phase encoding, cryptanalysis.

1. INTRODUCTION

Optical information-processing systems have emerged as a very promising means of encryption, securing and validation of data [1-17]. They exploit the advantage of the inherent parallelism of optical systems, the processing of information at the speed of the light, and its versatility for manipulating the information in the spatial domain or in the spatial frequency domain. Under these circumstances, optical systems are considered to be a good solution for real-time secure 3D television and video-conference systems, where a digital communication channel transmits encrypted digital holograms [6-8] and then this information is retrieved and displayed by means of secure 3D display systems [9]. To encrypt the data, one of the most popular techniques is the double random phase encryption scheme [10,11]. This scheme uses a classical 4-f correlator and two random phase masks. The first mask is placed in the input plane immediately after the image to be encrypted. The second mask is located in the Fourier domain. Up to now, a complete analysis regarding the security of this technique has not been performed.

In this paper, we present such an analysis of the numerical implementation of amplitude encoding for double random phase encryption. There are other types of double random phase encoding that are more secure. For example, fully phase encoded approaches as described in Ref. 11 may be more difficult to attack. Also, double random phase encoding becomes much more secure when employed in optical systems. The phase codes could be stored in thick optical materials such as photo-polymers. If the keys are recorded on photo-polymers, there are nonlinear effects of the optical materials as well as the Bragg condition. The nonlinear effects need to be known depending on the particular parameters used during the recording of the keys. The Bragg regime requires that the angle of incident light corresponding to the optical code satisfies the Bragg condition.

In case of optical implementation, the keys can be placed in the Fresnel domain which provides a multi-dimensional space to encode the information [14]. The correlation length of the optical keys can be in micron range which requires a substantially large number of random searches in 3D space for each trial of a 2D phase key. If wavelength [15] and polarization encoding [16] are utilized, they will add additional degrees of difficulty. In addition, implementations with optical storage are more secure because the encrypted data is not easily available in digital form for processing. In general, these optical properties and multi-dimensional topologies cannot be easily simulated by digital computers and it is difficult to determine the reliability of these methods by digital techniques.

Section 2 describes the principle of the double random phase encryption and its theoretical security. In section 3 we demonstrate that it is not necessary to know the complete keys in order to get an approximate decrypted image. In section 4 we describe two attacks that require an attacker to be able to choose some images to encrypt. In section 5, we present an attack that uses two known – but not chosen – images. Conclusions are given in section 6.

*yann@leibniz.iimas.unam.mx; phone +52-55-5622-3573; fax +52-55-5622-3620

Unmanned/Unattended Sensors and Sensor Networks II,
edited by Edward M. Carapezza, Proc. of SPIE Vol. 5986
598603, (2005) · 0277-786X/05/$15 · doi: 10.1117/12.633677

2. DOUBLE RANDOM PHASE ENCRYPTION

In this paper, we concentrate on the analysis of the original double random phase optical encryption schemes that were described in [10] and [11]. However, identical or similar techniques could be applied to variants of the schemes or even to other optical encryption techniques.

2.1 Principle and notations

The considered encryption scheme is sketched in Fig. 1. Figure 1(a) shows the optical implementation of the algorithm and Fig. 1(b) sums up the performed operations. Following common cryptographic nomenclature, let us call *P* the plain (original) image and *C* the ciphered (encrypted) image. Let us denote *X* and *Y* the first and second phase keys respectively. All of *P*,*C*,*X* and *Y* are two-dimensional functions. For now, we make no assumption concerning the nature of the plain image *P*; it can be real- or complex-valued. On the other hand, the keys *X* and *Y* are pure phase functions, that is, their magnitude is constant and equal to 1. As can be seen in Fig. 1, the encryption process consists of multiplying – point by point – the plain image *P* by the first key *X*. The result of this product is Fourier transformed and the obtained spectrum is multiplied – again point by point – by the second key *Y*. The result of these operations is the ciphered image *C*. The complete process can be written as

$$C = Y \cdot \mathcal{F}\left(P \cdot X\right) \quad , \tag{1}$$

where $\cdot$ denotes point-by-point product and $\mathcal{F}$ denotes the two-dimensional Fourier transform (FT) operation. Note that the encryption process described in [10] performs an additional inverse FT, so that the actual ciphered image is the inverse FT of *C*. However this final step does not add anything to the security of the system and, knowing this final image, it is always possible to compute its inverse FT to obtain *C*. For this reason we ignore this final transformation.

For the needs of the analysis, we consider that *P*,*C*,*X* and *Y* are discrete (pixelized) and spatially bounded. This is always the case in practice, given that even continuous images and phase keys can be adequately sampled. We therefore represent these four two-dimensional functions by four $r \times c$ arrays, where *r* and *c* are the numbers of rows and columns respectively. The total number of pixels of each array is then $N = r \times c$. In addition, we re-arrange each array into a vector with *N* components; this vector is formed by sequentially appending each column of the original array. The four vectors are called P,C,X and Y.

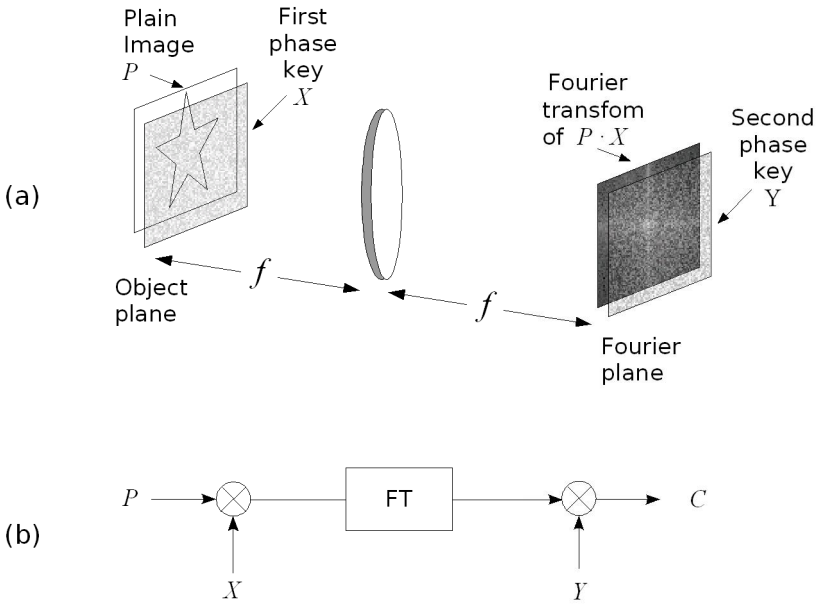


Figure 1: Double random phase encryption scheme. (a) Optical implementation. (b) Operation summary.

2.2 Theoretical security and weaknesses of the scheme

The encryption model in [10] uses keys with continuous values of phase. However a practical system will be limited to a discrete number of phase levels L. If the system were perfect, an attacker who only knows a ciphered image C would have to try every possible key in order to retrieve the plain image P. Since there are two N-pixel keys with L phase levels for each pixel, so this brute-force attack would require trying L^{2N} combinations. For any practical values of L and N, this number is huge and – more importantly – it increases exponentially with the number of pixels N. This degree of complexity is deemed computationally intractable. In order to give an idea of the involved difficulty, let us take an example with $L = 16$ phase levels and $N = 100 \times 100$ pixels in each key. Then the number of keys to try would be $16^{20000} \sim 10^{24000}$, which is inconceivable. However, these L^{2N} combinations concern the general – and ideal – case. This number does not always measure the actual security of the system. Under certain conditions, the number of necessary attempts can be somewhat reduced. In section 3, the actual strength of the encryption technique is given is some particular cases.

Another important aspect of a cryptosystem is its capacity to resist a known plaintext or a chosen plaintext attack [18,19]. Namely, let us assume that the attacker knows a plain image and its corresponding ciphered image, or even that an authorized user can be induced to cipher a plain image chosen by the attacker. Then the knowledge of this particular pair – or even of many pairs – of plain image and ciphered image must not help the attacker to decrypt other ciphered images. In particular it should not be possible to recover the keys from one or several pairs of plain/ciphered images. Unfortunately the encryption scheme considered in this paper has a property that facilitates this type of attack. This property is linearity. Indeed, the system is a composition of point-by-point multiplications and Fourier transforms, which are all linear operations. If P_1, P_2 are two plain images with corresponding ciphered images C_1, C_2, and α, β are two scalar coefficients, it is straightforward to show using Eq. (1) that $\alpha P_1 + \beta P_2$ is encrypted to $\alpha C_1 + \beta C_2$. Considering the discrete case, this linearity means that the transformation between the plain image vector P and the ciphered image vector C can be represented by a matrix-vector product as follows

$$\mathsf{C} = \boldsymbol{T}\mathsf{P} \quad , \tag{2}$$

where $\boldsymbol{T}$ is the $N \times N$ transformation matrix that characterizes the encryption process. We show in sections 4 and 5 how this linearity property can be used to attack the system.

3. APPROXIMATE DECRYPTION USING REDUCED KEYSPACES

In this section, we show that it is not always necessary to know the complete keys in order to decrypt – at least partially – a ciphered image. We compute the strength of the encryption technique in a few particular cases. Throughout this section, we assume that the plain image P is a pure magnitude function with a zero (or constant) phase.

3.1 Ignoring the first key

If the plain image P has a zero phase, it is not necessary to know the first key X in order to be able to decrypt a ciphered image C. Indeed, if we only know the second key Y, we can easily recover the product $P \cdot X$ by

$$P \cdot X = F^{-1}\left(C \div Y\right) \quad , \tag{3}$$

where F^{-1} stands for the inverse FT and $\div$ stands for the point-by-point division. Now because P is a pure magnitude function and X is a pure phase function, the plain image is simply recovered by

$$P = \left|P \cdot X\right| = \left|F^{-1}\left(C \div Y\right)\right| \quad , \tag{4}$$

where $|\ |$ is the modulus.

Since the first key X is not necessary for the decryption, a brute-force attack only needs to try every possible instance of the second key. The number of combinations is therefore L^N instead of L^{2N}. This represents a great simplification since the number of trials is reduced by a factor L^N. In the case of the example used in section 2.2, the number of combinations

to try is $16^{10000} \sim 10^{12000}$. Fortunately, this number remains huge. Note that this simplification is not applicable in the case of a phase-coded plain image [11]. For the next two subsections, we still assume that the plain image is magnitude-coded and we therefore ignore the first key.

3.2 Reducing the number of phase levels

We have assumed that the keys are coded with L phase levels. For a brute-force attack, it is possible to reduce the number of phase levels to try, while accepting some loss of quality in the decrypted image. For instance, we can look for keys with only two phase levels. The number of possible combinations is then 2^N instead of L^N. For the example of section 2.2, the number of combinations to try would be $2^{10000} \sim 10^{3000}$.

In order to show the effect of reducing the number of phase levels of the key during decryption, we encrypt a 100 × 100 image with continuous phase keys ($L = \infty$; practically, 64-bit floating point representation for each phase value). Figure 2 shows the result of decrypting with the original key Y and with various reductions of phase levels of the same key. It can be seen that the original image is recognizable even with a binarized phase key Y. However, the fewer phase levels, the more noise is introduced in the reconstruction. This technique is therefore only usable if the plain image does not possess small details.

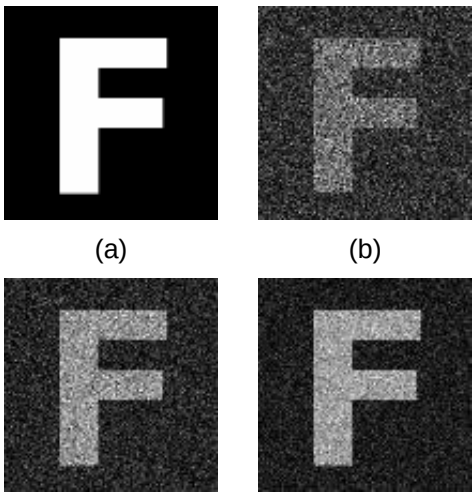


Figure 2: Decryption of an image encrypted with a continuous phase key. (a) Continuous decryption key. (b) Two-level decryption key. (c) Three-level decryption key. (d) Four-level decryption key.

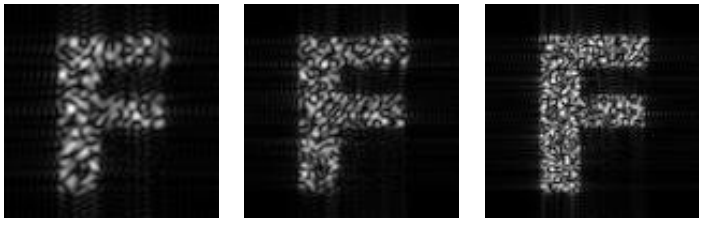


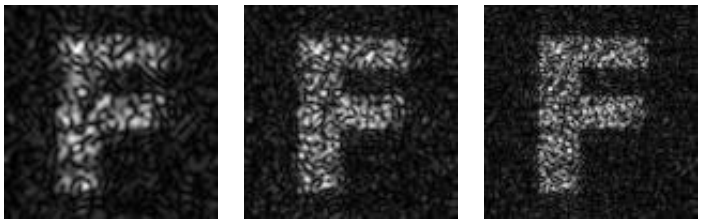
Figure 3: Decryption using partial windows of the ciphered image. (a) 30x30 window (b) 40x40 window (c) 50x50 window.

3.3 Using a sub-window of the second key

Because of the first random key X and the FT performed during the encryption, the information of the plain image is spread over the whole ciphered image. It is sometimes possible to recover enough information about the plain image by decrypting only a partial window of the ciphered image. For instance, Figs. 3(a)-(c) present decryptions of a 100 × 100

image using partial windows of the ciphered image with various sizes. Since an approximate reconstruction of the plain image is possible from only a window of the ciphered image, it is sufficient to look for the corresponding window of the second key Y instead of the full key. If the window has a total of $N_r<N$ pixels, then the number of combinations to try is L^{Nr} instead of L^N. In the case of Fig. 3(a), it is enough to find a 30×30 window of the second key. The number of combinations is thus reduced to $16^{900} \sim 10^{1080}$.

This technique of finding a partial window of the second key can be combined with the phase-level reduction technique described in subsection 3.2. If the number of phase levels is $L_r<L$, then the number of combinations is L_r^{Nr}. Figure 4 gives decrypted images using four phase levels for the second keys and various window sizes. In the case of Fig. 4(b), the number of trials would be $4^{1600} \sim 10^{960}$.



(a) (b) (c)

Figure 4: Decryption using partial windows of the ciphered image and reduction to four phase levels. (a) 30x30 window (b) 40x40 window (c) 50x50 window.

As with the reduction of phase levels, the use of a partial window loses some of the information and therefore introduces noise in the recovered image. For this reason, it can only be used when the plain image does not have small details. Note that in all the cases we have presented in this section the number of keys to consider is vastly reduced compared to the original number of keys, and yet the remaining number of possibilities is still intractable. In the following sections, we present different approaches to attack the encryption technique.

4. CHOSEN PLAINTEXT ATTACK

In this section, a different type of attack is considered. We assume that the attacker has the ability to trick a legitimate user of the system into encrypting particular images.

4.1 Using a base of the input space

As explained in section 2.2, the encryption operation is linear and can be represented by a matrix-vector product as in Eq. (2). As an operation in a vector space of dimension N, the encryption is fully defined by its operation on a base of the vector space. In other terms, let us suppose that the attacker can obtain the ciphered images $C_1,...C_N$ of N plain images $P_1,...P_N$ that constitute a base of the vector space. Then an $N \times N$ matrix $\mathbf{P}$ can be formed, using as columns the vectors $\mathsf{P}_1,...\mathsf{P}_N$ that correspond to the N plain images. Similarly, an $N \times N$ matrix $\mathbf{C}$ can be formed with the vectors corresponding to the ciphered images. The process of encrypting all the plain images can be written as

$$\boldsymbol{C}=\boldsymbol{T}\boldsymbol{P} \quad , \tag{5}$$

where $\mathbf{T}$ is the same encryption matrix as in Eq. (2). Now, $\mathbf{P}$ and $\mathbf{C}$ are known to the attacker, and $\mathbf{P}$ is invertible because its columns are linearly independent. So, the encryption matrix can be computed by

$$\boldsymbol{T}=\boldsymbol{C}\boldsymbol{P}^{-1} \quad , \tag{6}$$

with $\mathbf{P}^{-1}$ the inverse of $\mathbf{P}$. At this stage the system is compromised because knowing the encryption matrix $\mathbf{T}$ is equivalent to knowing the keys. Indeed, from Eq. (2), we see that any unknown plain image P can be obtained from the ciphered image C by

$$\mathrm{P} = \boldsymbol{T}^{-1}\,\mathrm{C} \quad , \tag{7}$$

where $\mathbf{T}^{-1}$ is the inverse of $\mathbf{T}$.

To sum up, with the knowledge of *N* linearly independent plain images (a base of the vector space) and of the corresponding ciphered images, an attacker is able to retrieve the keys of the system and to decrypt any subsequent ciphered image. We have classified this attack as "chosen plaintext" because of the necessity that the *N* images be a base. In theory, the *N* images do not have to be specifically chosen by the attacker. It suffices that the attacker wait to know *N* linearly independent images. Note that the attack presented in this subsection is quite general and would work for any linear encryption technique [20,21]. In practice, this attack is not very useful because it requires one to know a huge number of plain images, for instance 10000 images if they have 100 × 100 pixels. In the next subsection we present a technique that requires much fewer images.

4.2 Using an impulse input image

Let us examine what happens if the attacker succeeds in having encrypted an image with a single bright pixel. We will assume that all the pixels of the plain image are turned off except the central pixel (note that the same analysis can be done if the bright pixel is not in the center of the image). The plain image is then

$$P(\mu,\nu) = \delta(\mu,\nu) \quad , \tag{8}$$

where (μ,ν) are the spatial coordinates and $\delta(\mu,\nu)$ is the impulse distribution that is equal to 1 in (0,0) and to 0 otherwise. Using Eq. (8) in Eq. (1), we obtain

$$\begin{aligned} C(\mu,\nu) &= Y(\mu,\nu)\cdot F\left[\delta(\mu,\nu)\cdot X(\mu,\nu)\right] \\ &= Y(\mu,\nu)\cdot F\left[\delta(\mu,\nu)\,X(0,0)\right] \\ &= X(0,0)\;Y(\mu,\nu) \end{aligned} \quad , \tag{9}$$

where X(0,0) is the central value of the first key. Equation (9) means that, to a constant phase factor, the ciphered image is equal to the second key (that it has a different constant amplitude is not important). Thus, by encrypting a single chosen plain image, the attacker can easily recover the second key.

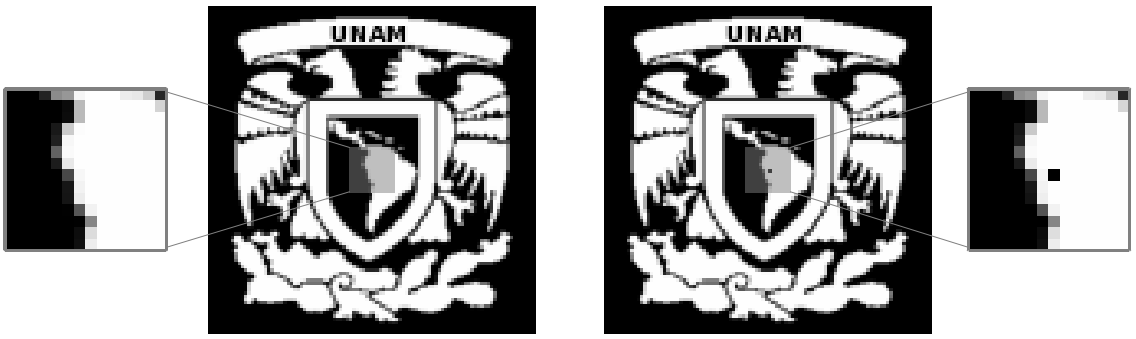


Figure 5: Key recovery using two plain images differing by one pixel.

Actually, there is a variant of this attack that could appear less suspicious to the authorized user. The attacker can provide two plain images P_1 and P_2 to encrypt in a way that

$$P_1(\mu,\nu) - P_2(\mu,\nu) = \delta(\mu,\nu). \tag{10}$$

Concretely, Eq. (10) means that P_1 and P_2 only differ by one pixel: the central pixel of P_1 is bright while that of P_2 is dark. Figure 5 presents an example of two such images. Now because of the linearity of the encryption scheme, we find that

$$C_1 - C_2 = Y \cdot F\left[(P_1 - P_2) \cdot X\right] \tag{11}$$

and using Eq. (10) we obtain

$$C_1 - C_2 = X(0,0)Y \quad . \tag{12}$$

The second key can thus be found by subtracting the two ciphered images.

If the plain images are amplitude images, then the knowledge of the second key Y is sufficient to decrypt the ciphered images (see section 3.1). On the other hand, if the plain images are phase coded [11], then it is not enough to know Y to perform the decryption. However, the remaining key X can be recovered with a single additional plain-ciphered image pair. Indeed, if P_3 is a uniform (spatially constant) image, with its corresponding ciphered image C_3, then Eq. (3) gives

$$P_3 \cdot X = F^{-1}\left(C_3 \div Y\right) \quad , \tag{13}$$

and since P_3 is constant, the first key is directly recovered as

$$X \propto F^{-1}\left(C_3 \div Y\right) \quad . \tag{14}$$

To sum up, with a maximum of three chosen plain-ciphered image pairs, it is possible to recover the two encryption keys and break the system. Yet, this attack has the disadvantage that the attacker must be able to induce an authorized user to encrypt some particular images. In the next section, we study an attack that operates with more relaxed conditions.

5. KNOWN PLAINTEXT ATTACK

In this sections, we assume that the attacker knows two plain images P and P' with their corresponding ciphered images C and C'. The plain images do not have to have very specific forms so they do not have to be chosen by the attacker. Once again, we use the vector representations of the plain images P, P', ciphered images C, C', and keys X, Y (see sections 2.2 and 4.1). We are now interested in the expression of the encryption process in terms of p_i, p'_i, c_i, c'_i, x_i and y_i, the elements of the vectors P, P', C, C', X and Y, respectively. For the first image, we can write

$$c_i = y_i \sum_{j=1}^{N} F_{ij}\, x_j\, p_j \quad \text{with } 1 \le i \le N, \tag{15}$$

where F_{ij} is the matrix that characterizes the FT operation. Note that, although it operates on vectors, it does not represent a one-dimensional FT but rather a two-dimensional FT on the images before they are re-arranged in vector form. For the second image, we can write similarly

$$c'_i = y_i \sum_{j=1}^{N} F_{ij}\, x_j\, p'_j \quad \text{with } 1 \le i \le N. \tag{16}$$

Now, it is possible to eliminate the elements of the second key y_i by cross-multiplying Eqs. (15) and (16). We get

$$c_i \sum_{j=1}^{N} F_{ij}\, x_j\, p'_j = c'_i \sum_{j=1}^{N} F_{ij}\, x_j\, p_j \quad \text{with } 1 \le i \le N, \tag{17}$$

hence

$$\sum_{j=1}^{N} F_{ij}\, x_j \left(c_i\, p'_j - c'_i\, p_j\right) = 0 \quad \text{with } 1 \le i \le N. \tag{18}$$

In Eq. (18), the only unknown variables are the components of the first key x_j. We thus rewrite this equation as

$$\sum_{j=1}^{N} S_{ij}\, x_j = 0 \quad \text{with } 1 \le i \le N, \tag{19}$$

where $S_{ij} = F_{ij}(c_i p'_j - c'_i p_j)$ is known. Equation (19) represents a system of N linear equations with N unknown variables. This system has a trivial solution $x_i = 0$ for every i. This solution is not the one we are looking for because we know that the first key is not zero. Therefore there has to be another solution to this system. This is only possible if the determinant of the matrix $\mathbf{S}$ is zero, that is if the N equations are not linearly independent. This interpretation of this property is that the keys are defined up to a constant phase factor. We can thus arbitrarily choose the value of one pixel of the key and this will fix the values of the rest of the pixels. We decide to fix $x_N = 1$. We can then eliminate the last equation of the system, because it is a linear combination of the other equations. With these modifications, Eq. (19) is transformed to

$$\sum_{j=1}^{N-1} S_{ij}\, x_j = -S_{iN} \quad \text{with } 1 \le i \le N\text{-}1. \tag{20}$$

This new system of N-1 equations with N-1 variables can be solved by classical system solving techniques such as Gauss elimination or LU decomposition [22]. Of course, the resolution is only possible if the determinant of this new system is not zero. In practice, this condition is not very restrictive. Once the first key is known, the second key is easily retrieved using for instance Eq. (15) by

$$y_i = \frac{c_i}{\sum_{j=1}^{N} F_{ij}\, x_j\, p_j} \quad \text{with } 1 \le i \le N. \tag{21}$$

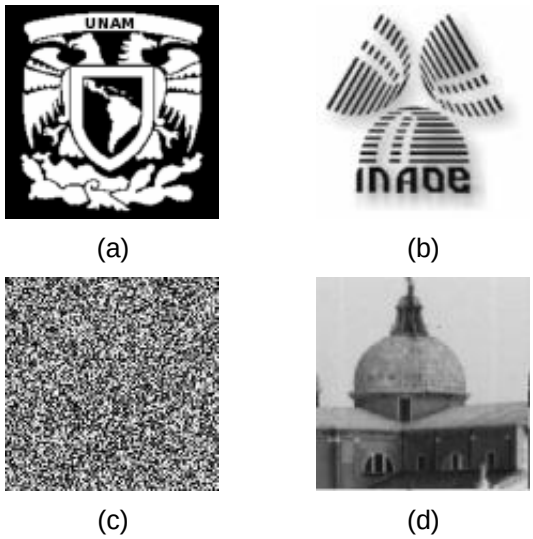


Figure 6: Decryption by known plaintext attack. (a) and (b) Two known plain images. (c) Unknown ciphered image and (d) its decryption using the keys retrieved thanks to the two known images.

An example of this attack is presented in Fig. 6. The known plain images P and P' are shown in Figs. 6(a) and 6(b) respectively. From these two images and their corresponding ciphered images, the keys X and Y are found. These recovered keys are then used to decrypt the unknown ciphered image shown in Fig. 6(c). The result of the decryption is given in Fig. 6(d). As can be seen, the plain image is successfully decrypted.

The attack presented in this section requires solving a system of about N equations. The complexity of this attack is thus $O(N^2)$ in space and $O(N^3)$ in time. This polynomial complexity makes the problem tractable. For the 100 × 100 images of Fig. 6, the keys were found in 6 hours on a 1.6 Mhz Pentium M computer using Gaussian elimination with backsubstitution.

6. CONCLUSION

In this paper, we have analyzed the security of the double random phase encryption scheme and we have described possible attacks. We first considered the feasibility of brute-force attacks and we showed that, although some tricks can be used to lessen the security, the remaining number of trials is prohibitive. However, we then described several possible chosen or known plaintext attacks and showed that the system is vulnerable to these forms of attack. With as few as one chosen image, it might be possible to break the system. Alternatively, two non-chosen but known plain images are sufficient to retrieve the keys. In this latter case, the process of computing the keys is computationally intensive but nevertheless feasible. An important weakness of the encryption scheme resides in its linearity. To minimize the risks, it is recommendable to use large keys, at least with 1000 × 1000 pixels. If possible, the re-use of keys should be avoided: new keys should be used for each encryption, as in a one-time pad approach [18,19].

In this paper, we have investigated the numerical implementation of amplitude encoding for double random phase encryption. There are other types of double random phase encoding that are more secure. For example, fully phase encoded approaches [11]. Also, double random phase encoding becomes much more secure when employed in optical systems. If the keys are recorded on photo-polymers, there are nonlinear effects of the optical materials as well as the Bragg condition. In case of optical implementation, the keys can be placed in the Fresnel domain which provides a multi-dimensional space to encode the information [14]. If wavelength [15] and polarization encoding [16] are utilized, they will add additional degrees of difficulty. In general, implementations with optical storage are more secure because the encrypted data is not easily available in digital form for processing, and the optical properties and multi-dimensional topologies can prove difficult to simulate digitally.

ACKNOWLEDGMENTS

The authors wish to thank César Franscisco Gamboa Verduzco for his help with the computational implementation of the decryption algorithm. Support is acknowledged from Enterprise Ireland and Science Foundation Ireland.

REFERENCES

1 J. L. Horner and B. Javidi, Opt. Eng. **38**, Special issue on Optical security, 1999.
2 B. Javidi and J. L. Horner, "Optical pattern recognition for validation and security verification," Opt. Eng. **33**, pp. 1752-1756, 1994.
3 H. Fielding, J.L. Horner, and C.K. Makekau, "Optical fingerprint identification by binary joint transform correlator," Opt. Eng. 30, pp 1958-1961 (1991).
4 B. Javidi and E. Ahouzi, "Optical security with Fourier plane encoding," Appl. Opt. **37**, pp. 6247-6255, 1998.
5 J.-L. de Bougrenet de la Tocnaye, E. Quémener, and G. Keryer, "Principles of pattern-signature synthesis and analysis based on double optical correlators," Appl. Opt. **39**, pp. 199-211, 2000.
6 E. Tajahuerce and B. Javidi, "Encrypting three-dimensional information with digital holography," Appl. Opt. **39**, pp. 6595-6601, 2000.
7 O. Matoba and B. Javidi, "Optical retrieval of encrypted digital holograms for secure real-time display," Opt. Lett. **27**, pp. 321-323, 2002.
8 T.J. Naughton and B. Javidi, "Compression of encrypted three-dimensional objects using digital holography," Opt. Eng. **43**, pp. 2233-2238, 2004.
9 O. Matoba and B. Javidi, "Secure three-dimensional data transmission and display," Appl. Opt. **43**, pp. 2285-2291, 2004.

10 Ph. Réfrégier and B. Javidi, "Optical image encryption based on input plane and Fourier plane random encoding ," Opt. Lett. **20**, pp. 767-769, 1995.
11 N. Towghi, B. Javidi, and Z. Luo, "Fully phase encrypted image processor," J. Opt. Soc. Am. A **16**, pp. 1915-1927, 1999.
12 Y. Li, K. Kreske, and J. Rosen, "Security and encryption optical systems based on a correlator with significant output images," Appl. Opt. **39**, pp. 5295-5301, 2000.
13 H.T. Chang, W.C. Lu, and C.L. Kuo, "Multiple-phase retrieval for optical security systems by use of random-phase encoding,", App. Opt. **41**, 4825-4834, 2002.
14 O. Matoba and B. Javidi, "Encrypted optical memory using multi-dimensional keys," Opt. Lett. **24**, pp. 762-765, 1999.
15 O. Matoba and B. Javidi, "Encrypted optical storage with wavelength key and random codes," Appl. Opt. **38**, pp. 6785-6790, 1999.
16 B. Javidi and T. Nomura, "Polarization encoding for optical security systems," Opt. Eng. **39**, pp.2439-2443, 2000.
17 B. Javidi ed., *Optical and Digital Techniques for Information Security*, Springer Verlag, New York, 2005.
18 B. Schneier, *Applied cryptography*, John Wiley and Sons, New York, 1996.
19 A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, Boca Raton, 1997.
20 S. R. Blackburn, S. Murphy, and K. G. Paterson, "Comments on 'Theory and applications of cellular automata in cryptography'," IEEE Trans. Comp. **46**, pp. 637-638, 1997.
21 H. Beker and F. Piper, *Cipher systems*, Van Nostrand, London, 1982.
22 W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical recipes in C*, Cambridge University Press, Cambridge, 1992.