Fast numerical generation and encryption of computer-generated Fresnel holograms

P. W. M. Tsang,[1,*] T.-C. Poon,[2] and K. W. K. Cheung[1]

[1]Department of Electronic Engineering, City University of Hong Kong, 83 Tat Chee Avenue, Kowloon, Hong Kong

[2]The Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, Virginia 24061, USA

*Corresponding author: eewmtsan@cityu.edu.hk

Received 23 July 2010; revised 7 November 2010; accepted 10 November 2010;
posted 11 November 2010 (Doc. ID 132144); published 20 December 2010

In the past two decades, generation and encryption of holographic images have been identified as two important areas of investigation in digital holography. The integration of these two technologies has enabled images to be encrypted with more dimensions of freedom on top of simply employing the encryption keys. Despite the moderate success attained to date, and the rapid advancement of computing technology in recent years, the heavy computation load involved in these two processes remains a major bottleneck in the evolution of the digital holography technology. To alleviate this problem, we have proposed a fast and economical solution which is capable of generating, and at the same time encrypting, holograms with numerical means. In our method, the hologram formation mechanism is decomposed into a pair of one-dimensional (1D) processes. In the first stage, a given three-dimensional (3D) scene is partitioned into a stack of uniformed spaced horizontal planes and converted into a set of hologram sublines. Next, the sublines are expanded to a hologram by convolving it with a 1D reference signal. To encrypt the hologram, the reference signal is first convolved with a key function in the form of a maximum length sequence (also known as MLS, or M-sequence). The use of a MLS has two advantages. First, an MLS is spectrally flat so that it will not jeopardize the frequency spectrum of the hologram. Second, the autocorrelation function of an MLS is close to a train of Kronecker delta function. As a result, the encrypted hologram can be decoded by correlating it with the same key that is adopted in the encoding process. Experimental results reveal that the proposed method can be applied to generate and encrypt holograms with a small number of computations. In addition, the encrypted hologram can be decoded and reconstructed to the original 3D scene with good fidelity.

OCIS codes: 090.0090, 090.1760, 100.4998.

1. Introduction

Numerical generation and encryption (coding) of holographic data are two important topics in the realm of holography that have instigated numerous research works in the past two decades. The former area, commonly known as computer-generated holography (CGH), has enabled holograms to be generated numerically from three dimensional (3D) objects or synthetic models that do not actually exist in the real world [1]. The latter one, encryption, allows a hologram to be converted to a form which can only be reconstructed with prior knowledge, such as the availability of a key signal, of the encoding process. The integration of these two technologies, hereafter referred to as the holographic encryption, has also enabled 3D optical images to be encrypted with more degrees of freedom than classical approaches which operate on the image pixels directly [2–8]. With the advancement of the computing technology, the holographic encryption process can be conducted with numerical means or a combination of numerical and optical means, which are more flexible and easier to implement than the classical optical methods. Along this direction, successful attempts have been made and published on the digital holographic processing techniques for information security [9–13]. Despite the success achieved to date on digital holography encryption,

0003-6935/11/070B46-07$15.00/0

there are a number of problems which have become identified as areas of investigation in recent years. Basically, heavy computation is involved in the hologram generation process, and the addition of numerical encryption will further lower the computation efficiency. Some of the representative works that provide a moderate amount of reduction on the computation load in hologram generation include [14–23]. Second, the decoding and reconstruction process is also computationally demanding as an encrypted hologram has a wider space-bandwidth product than the source data. Third, it is difficult to find a set of key signals to encrypt each hologram in a unique manner (so that each candidate can only be recovered with its own key), and which guarantee decryption on all the coded holograms. Fourth, but not the least, the key signals should not impose excessive disturbance on the frequency spectrum of a hologram. Otherwise, the reconstructed image may be contaminated with noticeable visual artifacts.

In this paper, we propose a method to address the aforementioned problems. First, we integrated the hologram generation and encryption mechanism into a single entity and realized it with a pair of one-dimensional (1D), finite impulse response (FIR) filtering processes. This results in a substantial reduction in the computation loading. Second, we employed the set of 1D maximum length sequence (M-sequence) [24] as the key signals. The latter are spectrally flat, which does not impose excessive distortion on the frequency spectrum of the holographic data. According to [24], the autocorrelation function of each member in the M-sequence is close to a train of Kronecker delta function. As a result, the encrypted hologram can be decoded by simply correlating it with the same key that is adopted in the encoding process. The decryption process can be achieved with numerical means with a small amount of arithmetic operations. Third, the correlation between different members in the M-sequence is very low so that each of them can be employed as a unique encryption key for encoding each source hologram.

The paper is organized as follows. After the introduction, the proposed method for fast generation and encryption of the Fresnel hologram is reported. We shall explain the decomposition of the hologram formation mechanism into a pair of 1D processes, and how encryption can be integrated without additional computation overhead. Experimental evaluation is given in Section 3. The vast difference between the reconstructed form of an encrypted hologram that has been decoded with, and without, the right encryption key is illustrated. In Section 4, we present a conclusion summarizing the essential findings.

2. Fast Generation and Encryption of Fresnel Holograms

A. Fast Generation of Fresnel Holograms

Given a set of 3D, self-illuminating object points $P = [p_0(x_0,y_0,z_0),p_1(x_1,y_1,z_1),\ldots\ldots,p_{N-1}(x_{N-1},y_{N-1},z_{N-1})]$, the diffraction pattern $D(x,y)$ on a vertical plane can be generated numerically with the following equation:

$$D(x,y)=\sum_{i=0}^{N-1}\frac{a_i}{r_i}\exp(jkr_i)=\sum_{i=0}^{N-1}\left[\frac{a_i}{r_i}\cos(kr_i)+j\frac{a_i}{r_i}\sin(kr_i)\right], \tag{1}$$

where a_i and r_i represent the intensity of the "ith" point and the distance between the object point and a point (x,y) on the diffraction plane, respectively. $k=\frac{2\pi}{\lambda}$ is the wavenumber and λ is the wavelength of the object wave.

The horizontal and vertical extents of the hologram is given by X and Y, respectively. Without loss of generality, we assumed that the source image has identical size and resolution as the hologram.

From Eq. (1), it can be seen that the number of multiplicative operations involved in the hologram generation process, assuming that all the sine and cosine terms are precomputed in advance (and therefore so as r_i), is

$$2\times N\times X\times Y. \tag{2}$$

In our proposed method in hologram generation, the object scene can be partitioned into a vertical stack of horizontal planes as shown in Fig. 1.

We further assume that the range of depth of the object points is small and centered at $z=z_o$. Suppose there are $N(\tau)$ object points on the $y=\tau$ plane, the diffraction pattern according to Fresnel diffraction, contributed by the plane is given by

$$\begin{aligned}D(x,y)_\tau &\approx \sum_{i=0}^{N(\tau)}\frac{a_i}{r_i}\exp\left(jk\frac{(x-x_i)^2}{2z_i}\right)\exp\left(jk\frac{(y-\tau)^2}{2z_o}\right)\\ &=\exp\left(jk\frac{(y-\tau)^2}{2z_o}\right)\sum_{i=0}^{N(\tau)}\frac{a_i}{r_i}\exp\left(jk\frac{(x-x_i)^2}{2z_i}\right)\\ &=R(y-\tau)O(x,\tau).\end{aligned} \tag{3}$$

The total object beam is then equal to the superposition of the diffraction pattern in each scan plane as

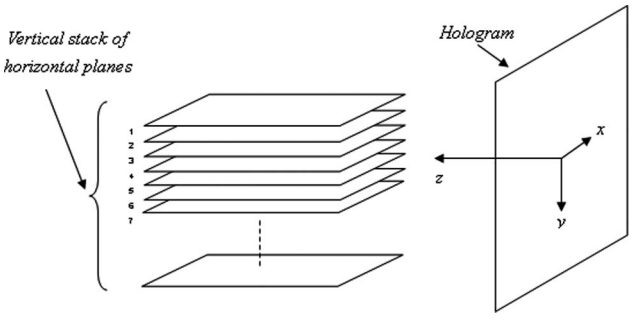

Fig. 1. Partitioning an object scene into a vertical stack (along y direction) of horizontal planes.

$$D(x,y) = \sum_{\tau} D(x,y)_{\tau} = \sum_{\tau} O(x,\tau) R(y-\tau). \quad (4)$$

Equation (4) is equivalent to a convolution process which can be represented by

$$D(x,y) = O(x,y) * R(y). \quad (5)$$

For clarity of explanation we refer to $R(y)$, which is employed to convert the sublines into a Fresnel hologram, as the "conversion signal." Compared with the direct implementation in Eq. (1), the alternative hologram generation method based on Eqs. (4) and (5) is significantly faster as it only involves a pair of one-dimensional processes, i.e., the formation of the diffraction pattern $O(x,\tau)$ in each scan plane and the subsequent generation of the overall diffraction pattern $D(x,y)$. We further noted that $O(x,\tau)$ can be derived with an economical hardware solution based on field programmable gate array (FPGA) at a throughput of over 100 M pixels per second [25].

Subsequently, a hologram can be generated from the diffraction pattern $D(x,y)$ by adding a planar or a spherical reference beam.

B. Fast Encryption of Holograms

We now propose to encrypt the hologram generated with Eq. (5) by convolving the conversion signal $R(y)$ with a one-dimensional binary M-sequence signal. Mathematically, the latter can be expressed as the primitive (prime) polynomial [24]

$$S(y) = 1 + g_1 y^1 + g_2 y^2 + \ldots + g_M y^M, \quad (6)$$

where all mathematical operations are performed in modulo-2, in other words, $+$ denotes the modulo-2 addition, and the value of $S(y)$ and $g_i|_{1\le i\le M}$ is either 1 or 0. For a given M in Eq. (6), there are $2^M - 1$ binary bits in the M-sequence.

And there exist certain settings for the coefficients $g_i|_{1\le i\le M}$ where one or more M-sequences can be generated. The M-sequence can be generated by a linear feedback shift register (LFSR) as shown in Fig. 2, where a cascade series of M stages of a single binary memory is shown.

Let $\Psi_{k;M}$ denote the kth output for a sequence of length $2^M - 1$. The output signals at the selected binary memory (commonly referred to as taps) are combined with the "exclusive or" (XOR) operation (which is basically the modulo-2 addition), and the result is feedback to the input of the LFSR.

For example with $M = 6$, two prime polynominals can be formed from Eq. (6) with

$$g_i = \begin{cases} 1 & (i = 5, 6) \\ 0 & \text{otherwise} \end{cases}, \quad (7a)$$

for the first polynominal, and

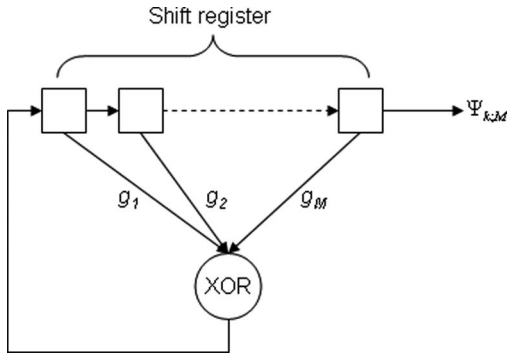

Fig. 2. M-stage LFSR for the realization of M-sequence.

$$g_i = \begin{cases} 1 & (i = 1, 6) \\ 0 & \text{otherwise} \end{cases}, \quad (7b)$$

for the second polynominal.

In other words with $M = 6$, a pair of M-sequences can be generated with a sequence length of $2^6 - 1$. In fact, with the prime polynomial, we can generate the M-sequence. For instance, with $g_i = 1$ for $i = 5, 6$; $g_i = 0$ otherwise given above, and assuming all the binary memories have been loaded with 1's, we take the fifth and sixth binary memory outputs and XOR them to obtain a new bit for the input of the shift register. This bit, as well as the rest of the contents in the shift register, are then shifted right at the command of a clock. Subsequently, a new bit value is shifted to the output of the shift register. This process is repeated through the control of the clock to generate $2^6 - 1$ "pseudorandom" bits before repeating the same sequence.

We now have the M-sequence, denoted as $\Psi_M(y) = (\Psi_{1;M}, \Psi_{2;M}, \ldots \Psi_{2^{M-1};M})$, where the argument y represents that we are applying the sequence along the y direction.

The diffraction pattern encrypted with the M-sequence $\Psi_M(y)$ is then given by

$$D_E(x,y) = O(x,y) * R(y) * \Psi_M(y). \quad (8)$$

Taking the Fourier transform on both sides of Eq. (8) along the vertical direction, we have,

$$\tilde{D}_E(x,e^{j\omega}) = \tilde{O}(x,e^{j\omega})\tilde{R}(e^{j\omega})\tilde{\Psi}_M(e^{j\omega}) = \tilde{O}(x,e^{j\omega})\tilde{E}(e^{j\omega}), \quad (9)$$

where $\tilde{E}(e^{j\omega}) = \tilde{R}(e^{j\omega})\tilde{\Psi}_M(e^{j\omega})$.

From Eq. (9), it can be seen that the generation and encryption of the hologram can be merged into a single process and conducted in the frequency domain. In other words, no extra computation load is required in the encryption process. Subsequently, the diffraction pattern can be obtained by inverse Fourier transforming $\tilde{D}_E(x,e^{j\omega})$ as

$$D_E(x,y) = \mathrm{FT}^{-1}\{\tilde{D}_E(x,e^{j\omega})\}. \quad (10)$$

A breakdown of the number of complex multiplication operations involved in Eqs. (10) (formulated based on Eqs. (8) and (9)) is given in Table 1. As the formation of $O(x,y)$ can be realized with the hardware solution, the computation loading on this part of the process is not taken into account.

According to Table 1, the amount of complex multiplication operation involved in the generation of the encrypted hologram $D_E(x,y)$ is given by

$$(X \times Y \log_2 Y) + (X \times Y) + (X \times Y \log_2 Y) = (X \times Y)(1 + 2Y \log_2 Y). \quad (11)$$

However, one complex multiplication is composed of four scalar multiplication operations. Hence in Eq. (11), the total number of scalar multiplication operations is equivalent to

$$4(X \times Y)(1 + 2Y \log_2 Y). \quad (12)$$

Comparing with Eq. (2), the computation advantage (CA) of the proposed method over the direct hologram generation process based on Eq. (1) is given by

$$\mathrm{CA} = \frac{2 \times N \times X \times Y}{4 \times X \times Y[1 + 2\log_2 Y]} = \frac{N}{2[1 + 2\log_2 Y]}. \quad (13)$$

Suppose the source image is of identical size and resolution as the hologram, and all the pixels in the source image are object points (i.e., $N = X \times Y$), we have

$$\mathrm{CA} = \frac{XY}{2[1 + 2\log_2 Y]}. \quad (14)$$

For a small hologram the size of 1024×1024 pixels, our proposed method is over 2×10^4 times faster than the direct hologram generation method based on Eq. (1).

The decryption process can be conducted in the frequency space by inverse filtering $\tilde{D}_E(x, e^{j\omega})$ as given by

$$D(x,y) = \mathrm{FT}^{-1}\left\{\frac{\tilde{D}_E(x, e^{j\omega})}{\tilde{E}(e^{j\omega})}\right\}. \quad (15)$$

Equation (15) indicates that the encrypted hologram can be perfectly reverted to its original form if it is decrypted with the same key.

Table 1. Breakdown of Number of Complex Multiplications for Each Step in Eq. (10)

Operation	Number of Complex Multiplications
$O(x,y) \rightarrow \tilde{O}(x, e^{j\omega})$	$X \times Y \log_2 Y$
$\tilde{D}_E(x, e^{j\omega}) = \tilde{O}(x, e^{j\omega})\tilde{E}(e^{j\omega})$	$X \times Y$
$\tilde{D}_E(x, e^{j\omega}) \rightarrow D_E(x,y)$	$X \times Y \log_2 Y$

Fig. 3. Test image Lenna positioned at 0.5 m from the hologram.

3. Experimental Evaluation

We shall evaluate our method with the standard image "Lenna" shown in Fig. 3. The image is positioned at a distance of 0.5 m from the hologram and the optical setting is given in Table 2.

The image "Lenna" is converted into a hologram without encryption according to Eq. (5), and the numerical reconstruction is shown in Fig. 4(a). It can be seen that apart from a slight blurriness of the image and minor ringing patterns near the boundary which is caused by the finite size of the hologram, the result is close to the original image.

Next, we apply our proposed method to generate, and concurrently encrypt the hologram with an M-sequence derived from a six-taps LFSR representing the polynominal $S_1(y) = 1 + y^5 + y^6$ [see Eq. (6) with $M = 6$ and the g_i given in Eq. (7a) for $(1 \le i \le 6)$]. For simplicity of explanation, we refer the polynominals representing the M-sequence as the "encryption keys."

The image reconstructed from the encrypted hologram is shown in Fig. 4(b). The result is barely visible and the original content is completely lost. The mean square error (MSE) is 3196, and the peak signal to noise ratio (PSNR) is 13.1 dB, indicating an extremely low coding fidelity.

Next the encrypted hologram is decrypted with the same encryption key $S_1(y)$, and the reconstructed image is shown in Fig. 4(c). It can be seen that the reconstructed image is identical to the reconstructed image derived from the unencrypted hologram shown in Fig. 4(a). The MSE is 0 which is in accordance with the theoretical deduction in Eq. (15).

Table 2. Optical Setting for Generating the Hologram

Wavelength	650 nm
Pixel size of hologram	$10.58\,\mu\mathrm{m} \times 10.58\,\mu\mathrm{m}$
Image/hologram dimension	$X = Y = 1024$
z_o of Eq. (3)	0.5 m

To demonstrate the effect of employing a wrong key to decrypt the encrypted hologram, the latter is decrypted with another encryption key $S_2(y)$ given by [see Eq. (6) with $M = 6$ and the g_i given in Eq. (7b) for $(1 \le i \le 6)$]

$$S_2(y) = 1 + y^1 + y^6,$$

and the reconstructed image is shown in Fig. 4(d). It can be seen that the original image is beyond recognition. The MSE is 1519, and the PSNR is 16.3 dB, indicating an extremely low coding fidelity.

The encryption method can be conducted with an encryption key of longer M-sequences. To demonstrate this, the proposed method is applied to generate and encrypt the hologram with an M-sequence derived from a eight-taps LFSR representing the polynominal $S_3(y) = 1 + y^1 + y^6 + y^8$. The eight-taps LFSR generates an M-sequence of length 256, which is four times the one generated with a six-taps LFSR. Subsequently, the encrypted hologram is decrypted with the same key $S_3(y)$ and the result is shown in Fig. 4(e). It can be seen that the reconstructed image is identical to the one corresponding to the unencrypted hologram shown in Fig. 4(a). The MSE is 0 demonstrating that the quality of the reconstructed image is not affected by the length of the M-sequence.

Next, we apply our proposed method to generate and encrypt a hologram generated from the image "peppers" as shown in Fig. 5(a). The horizontal and vertical extent of the test image is 512×512, and the image is evenly divided into an upper and a lower portion located at a distance of z_1 and z_2 from the hologram as shown in Fig. 5(b). A hologram is first generated without encryption with $z_1 = 0.502$ m, $z_2 = 0.498$ m, and $z_o = 0.50$ m according to the setting given in Table 1. The reconstructed images at distances of z_1 and z_2 are shown in Figs. 6(a) and 6(b), respectively.

It can be seen that apart from a slight blurriness, together with minor ringing at the boundary and the junction between the upper and lower sections, the reconstructed images at each section are similar to the original one.

(a) (b)

Fig. 5. (a) Test image evenly divided into an upper and lower section, each locating at a different depth from the hologram as shown in Fig. 5(b). (b) Depth of each section of the image in Fig. 5(a) to the hologram.

We then apply the proposed method to generate and encode the hologram for the source image in Fig. 5(a). Subsequently, the encoded hologram is decoded, and the reconstructed images at $z = 0.502$ m and $z = 0.498$ m are shown in Figs. 6(c) and 6(d), respectively. For both cases, the reconstructed images are identical to the ones obtained with the unencrypted holograms and the MSE is 0.

A similar test is performed on the source image in Fig. 5(a) with $z_1 = 0.49$ m, $z_2 = 0.51$ m, and $z_o = 0.50$ m. The reconstructed images of the unencrypted hologram at these two depth planes are shown in Figs. 6(e) and 6(f). It can be seen that when z_1 and z_2 are further away from z_0, the reconstructed images are slightly inferior than those shown in Figs. 6(c) and 6(d). The reason is that in applying the proposed method in generating the hologram [i.e., Eq. (3)], it has been assumed that the range of depth of the object points is within a close neighborhood of z_o. As a result, the quality of the reconstructed images will decrease as the deviation from the assumption escalates.

An encoded hologram based on the key $S_3(y)$ is then generated with the proposed method. The encrypted hologram is decoded, and the reconstructed

(a) (b) (c) (d) (e)

Fig. 4. (a) Reconstructed image from a hologram generated from the test image in Fig. 3. (b) Reconstructed image from a hologram generated from the test image in Fig. 3 and encrypted with the M-sequence generated by $S_1(y)$. PSNR = 13.1 dB. (c) Reconstructed image from a hologram generated from the test image in Fig. 1 and that has been encrypted and decrypted with the same encryption key, $S_1(y)$. The MSE, as compared with the reconstructed image of the unencrypted hologram, is 0. (d) Reconstructed image from a hologram generated from the test image in Fig. 1 and that has been encrypted with the encryption key $S_1(y)$ and decrypted with the key $S_2(y)$. PSNR = 16.3 dB. (e) Reconstructed image from a hologram generated from the test image in Fig. 3 and that has been encrypted and decrypted with the same encryption key, $S_3(y)$. The key is derived from an eight-taps M-sequence. The MSE, as compared with the reconstructed image of the unencrypted hologram in Fig. 4(a), is 0.

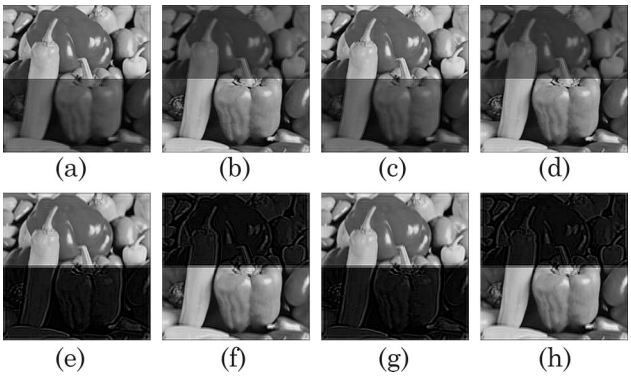

Fig. 6. (a) Reconstructed image from a hologram generated from the test image in Fig. 5(a) at a depth of $z = 0.502\,\mathrm{m}$ (i.e., z_1). The hologram is not encrypted. (b) Reconstructed image from a hologram generated from the test image in Fig. 5(a) at a depth of $z = 0.498\,\mathrm{m}$ (i.e., z_2). The hologram is not encrypted. (c) Reconstructed image from a hologram that is generated and encrypted with the proposed method from the test image in Fig. 5(a), at a depth of $z = 0.502\,\mathrm{m}$ (i.e., z_1). The encrypted hologram is decrypted with the correct key $S_3(y)$ prior to reconstruction. The MSE, as compared with the reconstructed image of the unencrypted hologram in Fig. 6(a), is 0. (d) Reconstructed image from a hologram that is generated and encrypted with the proposed method from the test image in Fig. 5(a) at a depth of $z = 0.498\,\mathrm{m}$ (i.e., z_2). The encrypted hologram is decrypted with the correct key $S_3(y)$ prior to reconstruction. The MSE, as compared with the reconstructed image of the unencrypted hologram in Fig. 6(b), is 0. (e) Reconstructed image from a hologram generated from the test image in Fig. 5(a) at a depth of $z = 0.51\,\mathrm{m}$ (i.e., z_1). The hologram is not encrypted. (f) Reconstructed image from a hologram generated from the test image in Fig. 5(a) at a depth of $z = 0.49\,\mathrm{m}$ (i.e., z_2). The hologram is not encrypted. (g) Reconstructed image from a hologram that is generated and encrypted with the proposed method from the test image in Fig. 5(a), at a depth of $z = 0.51\,\mathrm{m}$ (i.e., z_1). The encrypted hologram is decrypted with the correct key $S_3(y)$ prior to reconstruction. The MSE, as compared with the reconstructed image of the unencrypted hologram in Fig. 6(e), is 0. (h) Reconstructed image from a hologram that is generated and encrypted with the proposed method from the test image in Fig. 5(a) at a depth of $z = 0.49\,\mathrm{m}$ (i.e., z_2). The encrypted hologram is decrypted with the correct key $S_3(y)$ prior to reconstruction. The MSE, as compared with the reconstructed image of the unencrypted hologram in Fig. 6(f), is 0.

images at $z = z_1$ and $z = z_2$ are shown in Figs. 6(g) and 6(h), respectively. For both cases, the reconstructed images are identical to the ones obtained with the unencrypted holograms and the MSE is 0.

4. Conclusion

We have proposed a method for fast numerical generation and encryption of a Fresnel hologram. In the generation of the latter, a 3D object scene is partitioned into a vertical stack of evenly spaced horizontal scan planes. For each of the scan planes, a 1D hologram subline is generated. A Fresnel hologram is then derived by convolving the subline with a 1D conversion signal along the vertical direction. To encrypt a hologram, the conversion signal is first convolved with a member of the M-sequence prior to its application on the sublines. This effectively merges the generation and encryption process into a single entity. Our scheme has four major advantages. First, the amount of complexity involved in the generation of the hologram is significantly smaller than that employing the direct table lookup technique. Second, as explained previously, there is no additional computation involved in the encryption process. Third the M-sequence, which serves as a key, is spectrally flat and does not impose uneven attenuation on the frequency spectrum of the hologram. Fourth, the hologram can be decrypted simply by correlating it with the same M-sequence that is selected in the encryption process. Experimental results reveal that the visual quality of a reconstructed image from a hologram that is generated and encrypted by our method is extremely poor, if not beyond recognition, if it is obtained without decrypting the hologram with the correct key.

The works described in this paper are by no means definitive, but rather we anticipate that they can be taken as a foundation for further research in the realm of holographic encryption. For example, the latter can be integrated with existing image encryption schemes, such as [2–8] to foster stronger resistance of optical images towards illegal access and attacks. It is also possible to complement our proposed method with optical encryption [26], to prevent brute force decryption with numerical means. This is essential as the emergence of powerful hardware, such as the graphic processing unit (GPU), has enabled a massive amount of computation to be conducted in a parallel fashion.

References

1. T.-C. Poon, ed., *Digital Holography and Three-Dimensional Display: Principles and Application* (Springer, 2006).
2. P. P. Dang and P. M. Chau, "Image encryption for secure internet multimedia applications," IEEE Trans. Consum. Electron. **46**, 395–403 (2000).
3. S. J. Li, C. Q. Li, G. R. Chen, and K. T. Lo, "Cryptanalysis of the RCES/RSES image encryption scheme," J. Syst. Softw. **81**, 1130–1143 (2008).
4. I. Ozturk and I. Sogukpinar, "Analysis and comparison of image encryption," Int. J. Inf. Technol., **1**, 108–111 (2004).
5. M. Ali, B. Younes, and A. Jantan, "Image encryption using block-based transformation algorithm," IAENG Int. J. Comput. Sci. **35**, 15–23 (2008).
6. D. Arroyo, C. Q. Li, S. J. Li, G. Alvarez, and W. A. Halang, "Cryptanalysis of an image encryption scheme based on a new total shuffling algorithm," Chaos Solitons Fractals **41**, 2613–2616 (2009).
7. X. Y. Liang and X. L. Min, "An image encryption approach using a shuffling map," Commun. Theor. Phys. **52**, 876–880 (2009).
8. M. Sharma and M. K. Kowar, "Image encryption techniques using chaotic schemes: a review," Int. J. Eng. Sci. Tech **2**, 2359–2362 (2010).
9. E. Tajahuerce and B. Javidi, "Encrypting three-dimensional information with digital holography," Appl. Opt. **39**, 6595–6601 (2000).
10. B. Javidi and T. Nomura, "Securing information by means of digital holography," Opt. Lett. **25**, 28–30 (2000).
11. T. Nomura and B. Javidi, "Optical encryption system with a binary key code," Appl. Opt. **39**, 4783–4787 (2000).

12. K. B. Doh, K. Kim, and T.-C. Poon, "Computer generated holographic image processing for information security," Lect. Notes Comput. Sci. **3320**, 313–322 (2005).
13. Y.-H. Seo, H. J. Choi, and D. W. Kim, "Digital hologram encryption using discrete wavelet packet transform," Opt. Commun. **282**, 367–377 (2009).
14. T. Yamaguchi, G. Okabe, and H. Yoshikawa, "Real-time image plane full-color and full-parallax holographic video display system," Opt. Eng. (Bellingham, Wash.) **46**, 125801 (2007).
15. H. Yoshikawa, "Fast computation of fresnel holograms employing difference," Opt. Rev. **8**, 331–335 (2001).
16. S. C. Kim and E. S. Kim, "Effective generation of digital holograms of three-dimensional objects using a novel look-up table method," Appl. Opt. **47**, D55–D62 (2008).
17. S. C. Kim and E. S. Kim, "Fast computation of hologram patterns of a 3D object using run-length encoding and novel look-up table methods," Appl. Opt. **48**, 1030–1041 (2009).
18. Y. Sakamoto and T. Nagao, "A fast computational method for computer-generated Fourier hologram using patch model," Electron. Commun. Jpn., Part 2, Electron. **85**, 16–24 (2002).
19. K. Matsushima, H. Schimmel, and F. Wyrowski, "Fast calculation method for optical diffraction on tilted planes by use of the angular spectrum of plane waves," J. Opt. Soc. Am. A **20**, 1755–1762 (2003).
20. K. Matsushima and S. Nakahara, "Computer generated holograms for three dimensional surface objects with shade and texture," Appl. Opt. **44**, 4607–4614 (2005).
21. H. Sakata and Y. Sakamoto, "A fast computation method for Fresnel hologram using three-dimensional affine transformations in real space," Appl. Opt. **48**, H212–H221 (2009).
22. H. Yoshikawa, "Computer-generated holograms for white light reconstruction," in *Digital Holography and Three-Dimensional Display: Principles and Applications*, T.-C. Poon, ed. (Springer, 2006).
23. T.-C. Poon, T. Akin, G. Indebetouw, and T. Kim, "Horizontal-parallax-only electronic holography," Opt. Express **13**, 2427–2432 (2005).
24. S. W. Golomb, *Shift Register Sequences*, revised ed. (Aegean Park Press, 1981).
25. P. W. M. Tsang, J. P. Liu, K. W. K. Cheung, and T.-C. Poon, "Fast generation of Fresnel holograms based on multirate filtering," Appl. Opt. **48**, H23–H30 (2009).
26. K. B. Doh, K. Dobson, T.-C. Poon, and P. S. Chung, "Optical image coding with a circular Dammann grating," Appl. Opt. **48**, 134–139 (2009).