Computer Physics Communications ■ (■■■■) ■■■–■■■

Contents lists available at ScienceDirect

Computer Physics Communications

journal homepage: www.elsevier.com/locate/cpc

Restoration of digital off-axis Fresnel hologram by exemplar and search based image inpainting with enhanced computing speed

A.S.M. Jiao [a,*], P.W.M. Tsang [a], T.-C. Poon [a,b]

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

[b] Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA 24061, USA

ARTICLE INFO

Article history:
Received 8 July 2014
Received in revised form
16 March 2015
Accepted 29 March 2015
Available online xxxx

Keywords:
Restoration
Hologram
Damage
Image inpainting
ABC algorithm

ABSTRACT

In this paper, we proposed a method for restoration of heavily damaged digital off-axis Fresnel holograms based on image inpainting. Being different from existing image inpainting algorithms, we have employed Artificial Bee Colony search technique to increase the computation efficiency of the inpainting process, resulting in a reduction of over one order of magnitude in the computation time. Briefly, an exemplar and search based image inpainting technique is applied to fill in the damaged regions in the hologram, with the most similar fringe patterns in the intact areas obtained from Artificial Bee Colony search algorithm. Experimental results reveal that with our proposed method, the pictorial information in heavily damaged holograms can be recovered with good fidelity, as compared with those obtained with the original intact holograms. This is the first time that image inpainting technique is applied on hologram restoration.

1. Introduction

The advancement of computing technologies in the past 2 decades has imposed significant impact in the research on Digital Holography. At present, it is possible to generate and process digital holograms at video rates with high speed hardware devices [1], or fast software algorithms [2–4]. Similar to digital data of signals in general, possible damage and contamination can occur in the capture, transmission and storage process of digital holograms, leading to loss of fringe patterns in certain part(s) of the hologram. Although a hologram is inherently resistant to small amount of damage [5], the pictorial content can still be severely jeopardized if the defective area is too large. In digital communication and data processing, error correction code is a common method for error detection and correction. Additional bits (commonly referred to as the error correction codes) are inserted into each sample of data so that if some of the data bits are erroneous, they could be detected and corrected by the added bits. A comprehensive description on classical error correction methods can be found in [6]. However, the insertion of the error correction codes will increase the data size significantly, especially when the amount of errors to be corrected is large. In this paper, we propose a method for restoration of heavily damaged digital holograms based on the concept of image inpainting, a classical technique that is employed for repairing damaged photographs and images. We assume that some error detection measure, such as parity bits, has been added to the source hologram, so that a crude location of the defective area can be found. To date, there are many kinds of inpainting methods, and an overview of the topic can be found in [7]. The existing algorithms are mainly based on interpolation [8], PDE (Partial Differential Equation) [9], texture synthesis [10,11], Kalman filtering [12], tensor voting [13], exemplar and search [14], or combinations of a plurality of methods. The effectiveness of each algorithm is largely governed by the target image properties. For example, [8,9] are more effective in handling continuous natural images, [10,11] are most suitable for images with pure texture patterns, [12,13] are effective in preserving edge information, and [14] is employed to patch damaged area with similar patterns in other parts of the image.

Compared to natural images which are generally smooth in intensity distribution, a hologram is composing of high frequency fringe patterns. Hence, inpainting methods based on the assumption of smooth image intensities, or edge preservation, are not feasible for holograms. On the other hand, texture synthesis methods, that are often employed to reconstruct the textural content of optical images to give a visually pleasing effect, are not suitable for inpainting hologram. The reason is that the missing fringe patterns in holograms represent meaningful information about the

* Corresponding author.
E-mail address: albertjiaoee@gmail.com (A.S.M. Jiao).

http://dx.doi.org/10.1016/j.cpc.2015.03.018

recorded 3D object scene, and needs to be filled in with better accuracy. With all the factors above taken into account, we propose a hologram inpainting method that is based on the exemplar and search technique. Briefly, the missing pixels are scanned sequentially, starting from the ones that are nearer to the intact part of the hologram. For each visited pixel, a square patch surrounding it (the target patch) is defined. A search process is conducted on the intact part of the hologram to find a donor patch that contains the most similar fringe patterns as target patch. All the missing pixels in the target patch are then replaced by the corresponding ones in the donor patch. Our proposition is in fact supported by the past research attempts on the application of Vector Quantization (VQ) [15,16] in hologram compression [17–21]. In these works, a hologram is partitioned into non-overlapping square regions, and the fringe patterns in each region is replaced by the closest member (in terms of Euclidean distance) from a small library (generally referred to as the codebook) of pre-defined patterns. Significant compression can be achieved by representing each square region with the label of its closest member in the codebook, rather than the original fringe patterns. Experimental evaluation has demonstrated that after applying the Vector Quantization (VQ) compression, the fidelity of the pictorial content contained in the hologram can be preserved favorably even with a small codebook of 256–1024 fringe patterns. As such, our proposed hologram inpainting method is equivalent to replacing local fringe patterns of the damaged regions with a codebook that is derived from the hologram itself. In this paper, we shall be focusing on the restoration of digital off-axis Fresnel hologram that can be generated numerically [2–4,22,23], or acquired with optical means [24]. Organization of the paper is outlined as follows. After the introduction, the principles of our proposed hologram inpainting method are presented in Section 2. We shall point out that while the inpainting method is effective in restoring a damaged hologram, the computation time is too lengthy. In Section 3, the adoption of the Artificial Bee Colony (ABC) algorithm [25] for speeding the inpainting process is described. Experimental evaluation is provided in Section 4, followed by a conclusion on the essential findings.

2. The proposed hologram inpainting method

Our proposed method is targeted on the restoration of digital off-axis Fresnel hologram. The hologram formation process of digital off-axis Fresnel hologram is illustrated in the following. An object scene is assumed to be comprising of a collection of self-illuminated object points. The object wave emitted by these points on a hologram plane that is located at some distance away from the object space is given by

$$O(u,v) = \sum_{k=0}^{N-1} \frac{a_k}{r_k(u,v)} \exp\left(\frac{j2\pi r_k(u,v)}{\lambda}\right), \tag{1}$$

where u and v are the horizontal and vertical co-ordinates of the hologram plane. a_k and $r_k(u,v)$ are the intensity of the kth object point, and its distance to a point at (u,v) on the hologram. λ is the wavelength of the optical beam. Subsequently, the object wave can be converted into an off-axis hologram $H(u,v)$ by multiplying it with a reference plane wave $R(v)$ that is inclined at an angle θ along the vertical direction, and retaining the real part of the product as

$$H(u,v) = RE\left[O(u,v)R(v)\right]. \tag{2}$$

Our proposed hologram inpainting method is based on the exemplar and search technique that was developed by Criminisi in [14]. Here we present a modified version of the algorithm for the restoration of damaged hologram, and the steps are described as follows. The target hologram is derived from a computer-generated off-axis digital Fresnel hologram generated based on Eqs. (1) and (2), and which has been damaged in a known manner. The damaged pixels are considered as "missing" and their erroneous values will not be involved in the inpainting process. $p_{u;v}$ denotes a pixel that is located at the uth row, and vth column of $H(u,v)$. Initially, the hologram is partitioned into a donor region Φ that includes all the existing pixels, and a target region Ω that includes all the missing pixels. Each pixel is assigned a status value $s(u,v)$, which is initially set to 1 and 0 for intact and missing pixels, respectively. A $x \times x$ square window, known as a patch $P\left(p_{u;v}\right)$, is defined around each pixel $p_{u;v}$, and a priority value always between 0 and 1 is determined as

$$V(u;v) = \frac{1}{x^2}\sum_{m=-k}^{k}\sum_{n=-k}^{k} s(u+m,n+v), \tag{3}$$

where $k=(x-1)/2$. The x value needs to be set appropriately to achieve good inpainting quality and reasonable computing speed. It is determined by trial and error and a typical size of 35 x 35 is adopted in this paper. The difference between a pair of patches in the hologram is reflected by the Sum of Square Differences (SSD) given by

$$D\left(P(p_{u_1;v_1}),P(p_{u_2;v_2})\right) = \sqrt{\frac{1}{x^2}\sum_{m=-k}^{k}\sum_{n=-k}^{k}\left[H(u_1+m,v_1+n)-H(u_2+m,v_2+n)\right]^2}. \tag{4}$$

Note that if $p_{u_1+m;v_1+n}$ or $p_{u_2+m;v_2+n}$ is a missing pixel, we set its value to zero (i.e., $[H(u_1+m,v_1+n)-H(u_2+m,v_2+n)]=0$).

Our proposed hologram inpainting method for recovering the intensity values of the damaged pixels is outlined with the steps shown in Table 1.

Briefly, the process iteratively fills in the missing members in the patch (hereafter referred to as the target patch) associated with each missing pixel, from the most similar patch (hereafter referred to as the donor patch). As depicted in Step 4 of Table 1, the donor patch is found via a full search process by evaluating the Sum of Square Differences (SSD) between the target patch and all the patches in the donor region of the hologram, and selecting the one with the minimum Sum of Square Differences (SSD). It should be pointed out that a patch in the donor region referring to that all the pixels in this patch are in the donor region. The order of the patch to be filled is conducted progressively in descending order of priority values, to ensure that the filling is proceed along an inward direction, commencing from the boundary of the damaged region. In the hologram inpainting algorithm described above, the most computational intensive part is the full search process in Step 4, where the Sum of Square Differences (SSD) between each target patch and all the patches in the donor region of the hologram have to be evaluated with Eq. (4). For a small 1024×1024 hologram with about 10% of missing pixels, the inpainting time is over 1.5×10^4 s, which is too lengthy in practical applications. The comparison is conducted in Matlab R2011(a) programming environment. The processor is Intel (R) Core (TM) i7-2600 CPU @ 3.40 GHz. The RAM memory is 4.0 GB. It is run on 32-bit Win 7 operating system.

For the sake of clarity, we shall demonstrate the restoration of a small 8×8 hologram in Fig. 1(a) with the inpainting method. The three black squares represent the defective pixels in the hologram. The inpainting process, based on a patch size of 3×3, is outlined as follows with reference to the steps listed in Table 1.

(a) Step 1: the donor region Φ and target region Ω are identified with the corresponding pixels labeled with the status value '1' and '0', respectively as shown in Fig. 1(b).
(b) Step 2: The priority value (PV) of each missing pixel is computed with Eq. (3) and shown in Fig. 1(c). The maximum PV

Table 1
Proposed inpainting algorithm for hologram restoration.

Steps	Operation
1	Calculate the range of donor region Φ and target region Ω. If $\Omega = 0$, go to Step 6.
2	Compute a priority value for each missing pixel in the hologram with Eq. (3) based on the status values for each pixel. Find the maximum of the priority value (MVP).
3	Visit each pixel in the hologram along a top to bottom, and left to right sequence.
4	Suppose the current visited candidate is a missing pixel with priority value equal to the maximum of the priority value (MVP). The patch associated with the pixel is replaced by the most similar patch in the donor region of the hologram. Suppose the current pixel is $p_{u_c;v_c}$, the most similar patch $P(p_{u_{nc};v_{nc}})$ to $P(p_{u_c;v_c})$ will be $(u_{nc}; v_{nc}) = \arg\left\{\min\left[Dist\left(P(p_{u_c;v_c}), P(p_{u,v})\right)\right]\right\}\big|_{P(p_{u;v})\in\Phi}$. Subsequently, each missing pixel in $P(p_{u_c;v_c})$ will be replaced by the corresponding pixel in $P(p_{u_{nc};v_{nc}})$. The status value of each missing pixel in $P(p_{u_c;v_c})$ will be assigned to be the maximum of the priority value (MVP) instead of 0.
5	After all pixels are scanned, go to Step 1
6	End of process

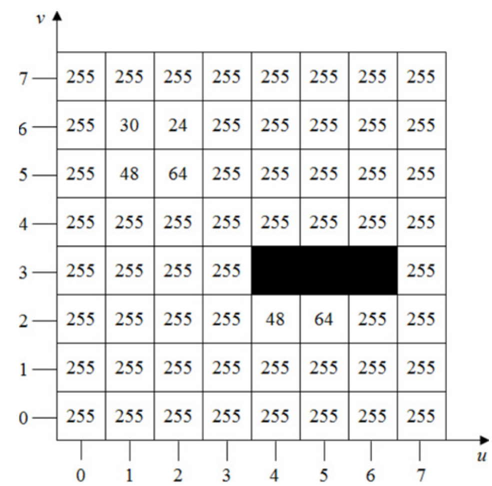

Fig. 1(a). A small 8 × 8 hologram with 3 erroneous pixels highlighted in black.

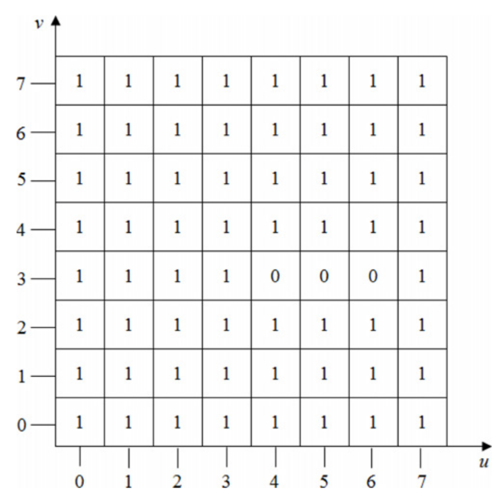

Fig. 1(b). Identify the donor region Φ and the target region Ω, with the corresponding pixels labeled with the status value '1' and '0', respectively.

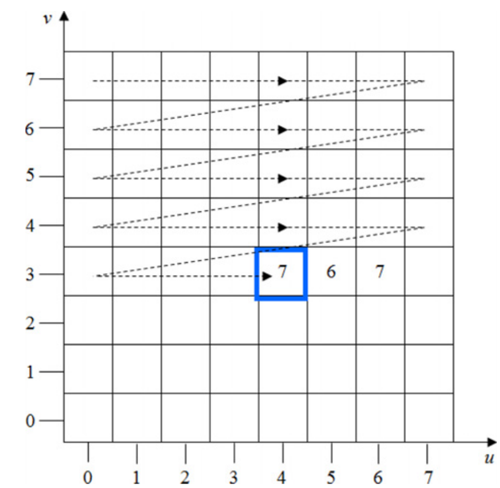

Fig. 1(c). The priority values of the missing pixels are computed based on Eq. (3). The MPV is 7, and the first missing pixel with the MPV that is located along the scan path is outlined in Blue. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

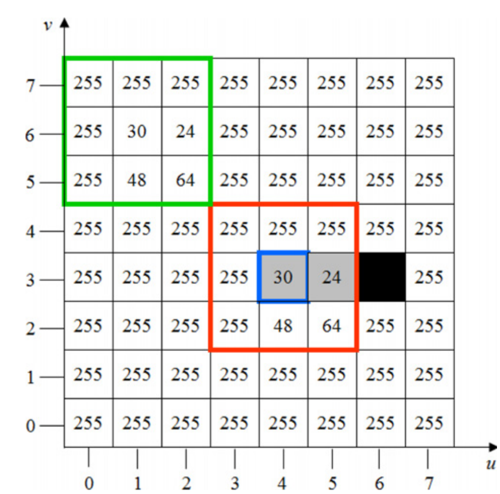

Fig. 1(d). The target patch is outlined in Red, and the closest donor patch (outlined in Green) is located based on Eq. (4). The 2 missing pixels in the target patch is replaced with the corresponding pixels in the donor patch. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

(MPV) is 7, and the first pixel with the MPV that is encountered in the scan path is located at $(u, v) = (4.3)$ and outlined in Blue. The patch centered at this pixel (it is assumed that the patch size $x = 3$), outlined in Red, is taken as the current target patch.

(c) Steps 3–4: The hologram is scanned along a top-to-bottom, and left-to-right manner. At each visited pixel, the Sum of Square Differences (SSD) of the patch centered on it is determined. After completing the scanning, the donor patch that is most similar to the target patch (i.e. minimum SSD) is determined and outlined in Green. The 2 missing pixels in the target patch is replaced by the corresponding pixels of the donor patch as shown in Fig. 1(d).

(d) Steps 5–6: The above process is repeated for the revised hologram in Fig. 1(d), which has only one remaining missing pixel. The target and the most similar donor patches are outlined in Red and Green, respectively as shown in Fig. 1(e). The missing pixel at $(u, v) = (6.3)$ replaced by the corresponding pixel at $(u, v) = (3.6)$ of the donor patch. After all the 3 pixels have been recovered, the process is terminated.

Please cite this article in press as: A.S.M. Jiao, et al., Restoration of digital off-axis Fresnel hologram by exemplar and search based image inpainting with enhanced computing speed, Computer Physics Communications (2015), http://dx.doi.org/10.1016/j.cpc.2015.03.018

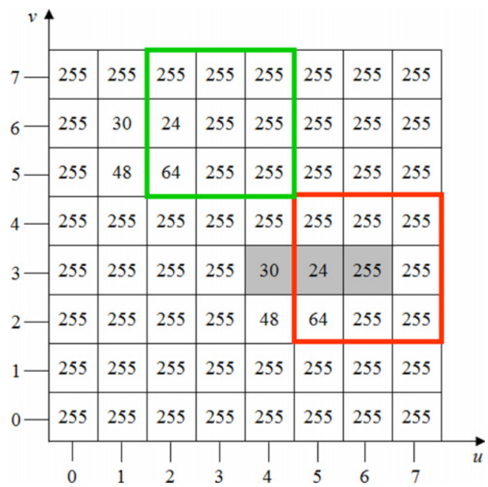

Fig. 1(e). The donor patch that is closest to the target patch centered at the remaining missing pixel at $(u, v) = (6.3)$ is identified based on Eq. (4). Subsequently, the missing pixel at $(u, v) = (6.3)$ is replaced by the corresponding pixel at $(u, v) = (3.6)$ of the donor patch. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

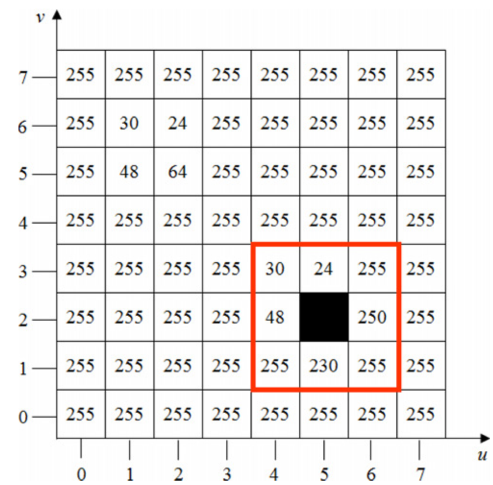

Fig. 2(a). A small 8×8 hologram with 1 erroneous pixel highlighted in black. The target patch corresponding to the missing pixel is outlined in Red. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

3. Enhancing the hologram inpainting process with the Artificial Bee Colony (ABC) algorithm

To shorten the lengthy computation time in the inpainting process described in Section 2, we further propose to adopt the Artificial Bee Colony algorithm to conduct the search of the donor patch. The Artificial Bee Colony (ABC) algorithm, proposed in 2006 [23] is an efficient and effective optimization method that has been widely adopted in many difficult engineering and scientific problems. The concept of Artificial Bee Colony (ABC) algorithm is based on the food searching behavior and the functional organization of honey bees. Briefly, three solution seeking mechanisms are involved in the algorithm, mimicking the tasks carried by the "employed bees", "onlooker bees", and the "scout bees". Each employed bee is a front line member to explore the richness of food source in its neighborhood. The collective information on the food sources gathered by the employed bees will be evaluated by the onlooker bees, after which they will determine the potential food sources. If an employed bee ceased to locate a better food source, the scout bee will be dispatched to pioneer new areas. When translated into the realm of optimization, the bees and food sources represent the solutions in the problem space, and their fitness on the solution, respectively. The Artificial Bee Colony optimization process is outlined by the following steps.

Step 1: Initialize the population of NP (for example, NP $= 10$) individuals (employed bees), each being an ordered pair pointing to a randomly potential donor patch location in the whole donor region of the hologram. The identity of an individual is indexed by the variable 'i'.

Step 2: Compute the Sum of Square Differences (SSD) between the potential donor patch indexed by each individual of the population, and the target patch. The fitness value of an individual, representing a donor patch at location $p_{u_i;v_i}$ is given by

$$fit_i = \frac{1}{1 + D\left(P(p_{u_c;v_c}), P(p_{u_i;v_i})\right)}, \tag{5}$$

which is basically the reciprocal of the Sum of Square Differences (SSD) between the target and the donor patches.

Step 3: A randomly selected mutant location nearby the potential donor is generated for each individual in the population PP, and its fitness is evaluated. If the fitness is higher than that corresponding to the existing candidate, the latter will be replaced by the mutant location.

Step 4: Calculate a probability value for each individual in the population. The probability value of the 'i' individual, denoted by '$prob_i$', is given by Eq. (6). It is proportional to the fitness values.

$$prob_i = \frac{fit_i}{\sum_{k=1}^{N-1} fit_k}. \tag{6}$$

Step 5: Onlooker bee stage. The process is similar to Step 3, but instead of producing one mutant donor image patch location for each current patch in the population, the number of mutant image patches is generated around each individual at a rate according to its probability value $prob_i$. As such, an individual with higher fitness value (which also leads to higher probability value), will have a higher chance of generating mutant patches around it. For example, maybe among the totally 10 mutant image patches, 5 are assigned to patch No. 1 in the population, 3 are assigned to patch No. 5, 2 are assigned to No.7. For the other patches, no mutant donor image patch is produced at this step. During this step, when one mutant image patch is generated, the fitness value is calculated. If it is better than the fitness value of the corresponding current patch in the population, it will replace the current one.

Step 6: Scout bee stage. For each donor patch in the population, count the number of iterations that it has not been updated in both Step 3 and Step 6. If the number of iterations exceeds the pre-defined limit (say limit $= 30$ iterations), this donor image patch location will be discarded from the population and a new image patch location randomly generated from any position in the hologram will replace this discarded one.

Step 7: find the donor image patch location with the best fitness value in the population. If it is better than the stored optimal matched image patch, the stored optimal matched image patch will be updated.

Step 8: go to Step 3 and start the next iteration unless the number of target iterations (say 2500 iterations) has been reached or the current best solution is higher than pre-defined satisfied Sum of Square Differences (SSD) threshold σ, which means the similarity between the visited target patch and the donor patch is already sufficiently good.

When the whole procedures are finished, one optimal donor image patch is found for the patch to be filled in.

For the sake of clarity, we shall illustrate the concept of applying the ABC algorithm in recovering a missing pixel of a small 8×8 hologram in Fig. 2(a). Similar to the example in Section 2, a small

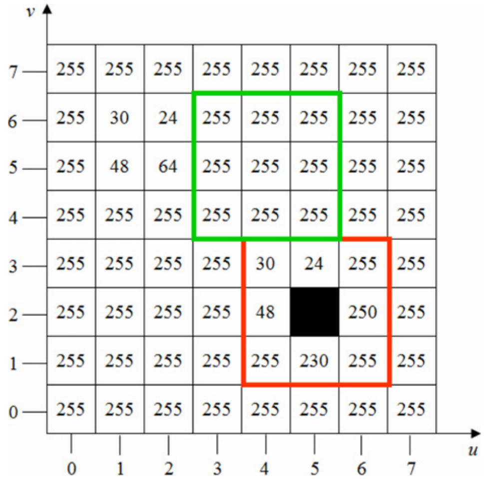

Fig. 2(b). Assuming a single bee colony, a randomly selected donor patch is generated and outlined in Green. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

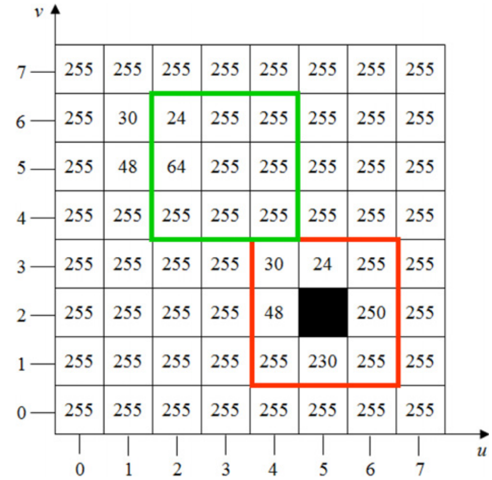

Fig. 2(d). The mutant patch, which is higher in fitness than the donor patch, is assigned to be the new donor patch (outlined in Green). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

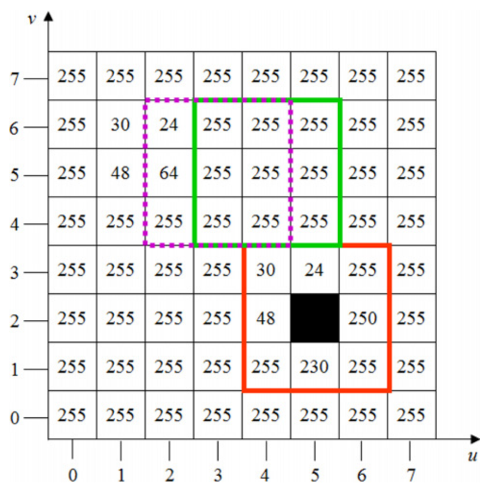

Fig. 2(c). A mutant patch (outlined in Purple) is randomly generated nearby the donor patch. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

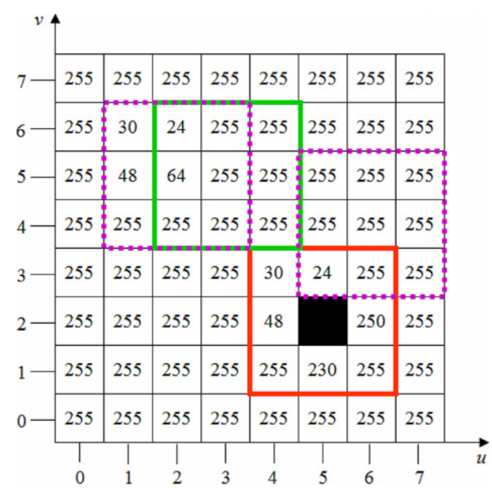

Fig. 2(e). 2 mutant patches (outlined in Purple) are generated from the donor patch in the bee colony. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

patch size of 3×3 is adopted, and the target patch corresponding to the missing pixel is outlined in Red. The σ value is assumed to be 0.1. To keep the description simple and easy to understand, we have assumed a colony with a single bee (i.e. NP = 1). The concept can be easily extended to a colony with multiple bees as each individual is operating in a more or less independent manner. The steps are outline as follows.

(a) Step 1: To begin with a population of NP bees each representing a randomly selected donor patch is generated (NP = 1 in this example). Suppose the selected donor patch is outlined in Green as shown in Fig. 2(b).
(b) Step 2: Eq. (4) is applied to compute the SSD between the donor and the target patch, resulting in a value of 136 and a fitness of 0.007.
(c) Steps 3–4: A mutant patch, outlined in dotted Purple, is randomly generated nearby the target patch as shown in Fig. 2(c). The fitness value is 0.012 (SSD = 82.4) which is higher than the target patch. As a result, the mutant patch is taken to be the target patch, as illustrated in Fig. 2(d).
(d) Step 5: The onlooker bee evaluates the fitness of the individual that has been assigned in the previous step. Suppose with such fitness value, 2 mutants (outlined in dotted Purple) are generated as shown in Fig. 2(e). The fitness of the left and the right mutants are 0.1 (SSD = 9) and 0.005 (SSD = 158), respectively.
(e) Steps 6–8: Amongst these 2 mutants, the one at the left has a fitness of 0.1 which is higher than that of the donor patch and the other mutant patches. The SSD is smaller than σ, which indicates that the mutant patch is quite matched with the target patch. As a result, the missing pixel is replaced by the corresponding pixel in the mutant patch at the right as shown in Fig. 2(f). As the correct donor has been located and taken to recover the missing pixel in the target patch, the process is terminated. Different from the full search in Section 2, only a small number of image patches are visited with ABC algorithm.

4. Experiment results

In the experiment, three test images shown in Fig. 3(a)–(c) are placed at an axial distance of 0.3 m from the hologram.

Eqs. (1) and (2) are applied to generate an off-axis hologram for each of the images. The wavelength λ of the optical beam is 650 nm and the reference plane wave is inclined at an angle of $\theta = 1.2°$ degree along the vertical direction. The hologram is composed of 1024×1024 pixels, each being a square of length 7 µm. Without loss of generality, the dimension and pixel size of the test images are assumed to be the same as the hologram. Each

Please cite this article in press as: A.S.M. Jiao, et al., Restoration of digital off-axis Fresnel hologram by exemplar and search based image inpainting with enhanced computing speed, Computer Physics Communications (2015), http://dx.doi.org/10.1016/j.cpc.2015.03.018

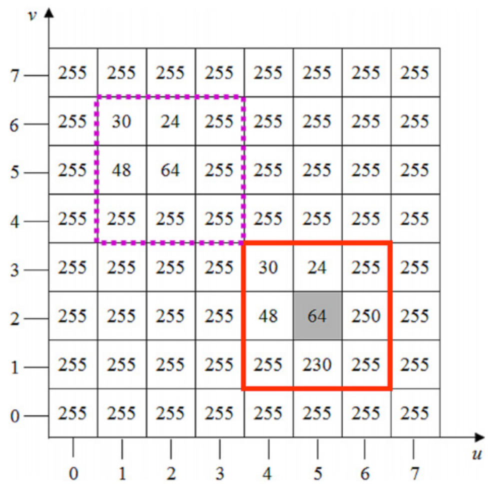

Fig. 2(f). The mutant patch at the left has highest fitness value, and selected to recover the missing pixel in the target patch.

hologram is damaged by removing some of its pixels. The damaged hologram is then restored with the full search inpainting and our proposed Artificial Bee Colony (ABC) based inpainting method. As an example, the hologram of the sample image in Fig. 3(b) is shown in Fig. 4(a), and the damaged parts are highlighted in Green as shown in Fig. 4(b). The holograms recovered with the full search inpainting method, and our proposed method are shown in Fig. 4(c) and (d), respectively. The images reconstructed by inverse Fresnel transform from original, damaged and restored holograms are illustrated in Fig. 5 with Table 2 documenting the RMS errors for the different methods. It can be observed that the reconstructed image quality after image inpainting is performed on the hologram is significantly improved compared with the damaged holograms. For the two different search methods in image inpainting, from Table 2, the reconstructed image RMS error of Artificial Bee Colony search method is very close to full search method when the stopping criterion of Artificial Bee Colony search in Step 8 is set to be $\theta^2 = 50$. But the computation time of hologram inpainting in this example with Artificial Bee Colony search method is 1695 s and the one with full search is 18 153 s. The computation conditions are the same as Section 2.

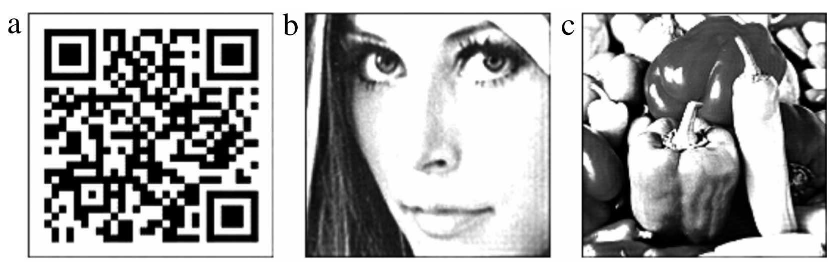

Fig. 3. (a)–(c) Test images.

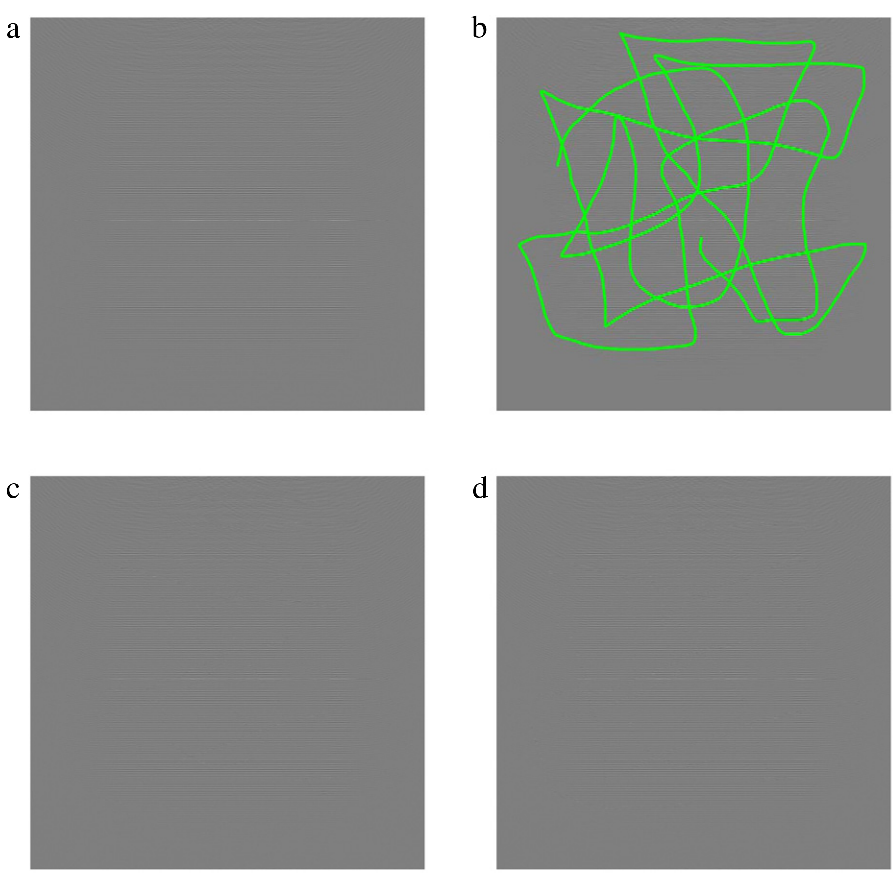

Fig. 4. (a) Original hologram; (b) damaged hologram; (c) restored hologram by full search; (d) restored hologram by ABC algorithm.

Fig. 5(a). Reconstructed image from original hologram.

Fig. 5(b). Reconstructed image from damaged hologram.

Fig. 5(c). Reconstructed image from restored hologram by full search.

Fig. 5(d). Reconstructed image from restored hologram by Artificial Bee Colony algorithm search when $\sigma^2 = 50$.

Table 2
Comparison of Root Mean Square (RMS) error in the reconstructed images from damaged and restored holograms with the reconstructed images from the original holograms.

Hologram	RMS error for reconstructed image from damaged hologram	RMS error for reconstructed image from restored hologram with full search	RMS error for reconstructed image from restored hologram with ABC algorithm search when $\theta^2 = 50$
QR Code	45.51	6.98	10.59
Lenna	59.22	4.49	8.63
Pepper	60.71	6.60	9.2

In the experiment, the trade-off between computation speed and reconstructed image quality for Artificial Bee Colony search method is investigated and the results for different σ values are illustrated in Table 3. It can be observed when the threshold σ is small, the computation time is longer but the reconstructed image quality is more close to the full search result (for pepper image, full search RMS error is 6.60). When the threshold σ increases, the computation speed can be fastened but the reconstructed image quality is reduced. The Artificial Bee Colony search algorithm can be tuned for different computation speed and reconstructed image quality requirement criteria.

5. Conclusion

This paper introduces an exemplar and search based image inpainting technique as a novel method for hologram restoration. The Artificial Colony Bee algorithm is used to speed up the searching process to reduce the computation to around only 10% or less as compared with the full search method. Experiment results have demonstrated that it can greatly improve the reconstructed image quality of damaged holograms with computation time significantly reduced. This work can also serve as foundations for further research works based on hologram inpainting such as hologram

Table 3
Comparison of computation time and Root Mean Square (RMS) error in the reconstructed images from damaged and restored holograms with the reconstructed images from the original holograms with Artificial Bee Colony search for different SSD threshold θ (pepper image).

σ^2	Computation time (s)	RMS error
25	3267	8.58
50	1695	9.2
75	1086	11.02
100	920	11.31
125	827	12.08
150	807	13.2

compression and speckle removal. Our proposed method can also be applied to the recovery of computer generated phase-only hologram (POH) [26–31], a topic of increasing interest in recent years on account of the higher diffraction efficiency of POH as compared with amplitude-only hologram. A potential application is to apply inpainting to recover hologram pixels in a POH that have been modified to accommodate encrypted data [32,33], so that the quality of the reconstructed images can be enhanced.

References

[1] Koki Murano, Tomoyoshi Shimobaba, Atsushi Sugiyama, Naoki Takada, Takashi Kakue, Minoru Oikawa, Tomoyoshi Ito, Fast computation of computer-generated hologram using Xeon Phi coprocessor, Comput. Phys. Comm. (2014) 2742–2757.
[2] P.W.M. Tsang, T.-C. Poon, Fast generation of digital holograms based on warping of the wavefront recording plane, Opt. Express 23 (2015) 7667–7673.
[3] Tomoyoshi Shimobaba, Jian Tong Weng, Takahiro Sakurai, Naohisa Okada, Takashi Nishitsuji, Naoki Takada, Atsushi Shirakic, Nobuyuki Masuda, Tomoyoshi Ito, Computational wave optics library for C++: CWO++ library, Comput. Phys. Comm. 183 (2012) 1124–1138.
[4] Jiantong Weng, Tomoyoshi Shimobaba, Minoru Oikawa, Nobuyuki Masuda, Tomoyoshi Ito, Fast recurrence relation for computer-generated-hologram, Comput. Phys. Comm. 183 (2012) 46–49.
[5] S.-M. Jiao, P.W.M. Tsang, A damage resistant holographic QR code, in: Digital Holography and Three-Dimensional Imaging Topical Meeting, Optical Society of America, 2013.
[6] W.W. Peterson, E.J. Weldon Jr., Error-Correcting Codes, second ed., MIT Press, Cambridge, MA, 1972.
[7] P. Patel, A. Prajapati, S. Mishra, Review of different inpainting algorithms, Int. J. Comput. Appl. 59 (18) (2012) 107–117.
[8] C. Lin, C. Yu, New interpolation algorithm for image inpainting, Physics Procedia 22 (2011) 107–111.
[9] M. Bertalmio, G. Sapiro, V. Caselles, C. Ballester, Image inpainting, in: Proc. ACM Conf. Comp. Graphics SIGGRAPH'00, 2000, pp. 417–424.
[10] A.A. Efros, W.T. Freeman, Image quilting for texture synthesis and transfer, in: Proc. ACM Conf. Comp. Graphics, SIGGRAPH'01, 2001, pp. 341–346.
[11] L.-Y. Wei, M. Levoy, Fast texture synthesis using tree-structured vector quantization, in: Proc. 27th Annual Conf. Comp. Graph. Inter. Tech., SIGGRAPH'00, 2000, pp. 479–488.
[12] G. Subrahmanyam, A. Rajagopalan, R. Aravind, Recursive framework for joint inpainting and de-noising of photographic films, J. Opt. Soc. Amer. A 27 (2010) 1091–1099.
[13] M. Kulkarni, A. Rajagopalan, Depth inpainting by tensor voting, J. Opt. Soc. Amer. A 30 (2013) 1155–1165.
[14] A. Criminisi, P. Peerez, K. Toyama, Region filling and object removal by exemplar-based image inpainting, IEEE Trans. Image Process. 13 (9) (2004).
[15] R.M. Gray, Vector quantization, IEEE ASSP Mag. 1 (2) (1984) 4–29.
[16] Y. Linde, A. Buzo, R.M. Gray, An algorithm for vector quantizer design, IEEE Trans. Commun. (1980) 702–710.
[17] Y. Lam, W. Situ, P. Tsang, Fast compression of computer-generated holographic images based on a GPU-accelerated skip-dimension vector quantization method,, Chin. Opt. Lett. 11 (2013) 050901.
[18] P.W.M. Tsang, K.W.K. Cheung, T.-C. Poon, C. Zhou, Demonstration of compression ratio of over 4000 times for each digital hologram in a sequence of 25 frames in a holographic video, J. Opt. 14 (12) (2012) 125403.
[19] P.W.M. Tsang, W.K. Cheung, T.-C. Poon, Low bit-rate compression of computer-generated Fresnel holograms based on vector quantization, Dig. Holo. and 3D Imag., OSA Tech. Dig. (CD) (OSA, 2011), paper DTuD1.
[20] P.W.M. Tsang, K.W.K. Cheung, T.-C. Poon, Low-bit-rate computer-generated color Fresnel holography with compression ratio of over 1600 times using vector quantization, Appl. Opt. 50 (2011) H42–H49.
[21] A.E. Shortt, T.J. Naughton, B. Javidi, Vector quantisation compression of digital holograms of three-dimensional objects, Proc. SPIE 5827 (2005) 265–273.
[22] T.-C. Poon, J.-P. Liu, Introduction to Modern Digital Holography with MATLAB, Cambridge University Press, 2014.
[23] P.W.M. Tsang, J.-P. Liu, W. Cheung, T.-C. Poon, Fast generation of Fresnel holograms based on multirate filtering, Appl. Opt. 48 (2009) H23–H30.
[24] T.-C. Poon, Optical scanning holography—A review of recent progress, J. Opt. Soc. Korea 13 (2009) 406–415.
[25] D. Karaboga, B. Basturk, A powerful and efficient algorithm for numerical function optimization: Artificial Bee Colony (ABC) algorithm, J. Global Optim. 39 (3) (2007) 459–471.
[26] P.W.M. Tsang, Y.T. Chow, T.-C. Poon, Generation of phase-only Fresnel hologram based on down-sampling, Opt. Express 22 (2014) 25208–25214.
[27] P.W.M. Tsang, A. Jiao, T.-C. Poon, Fast conversion of digital Fresnel hologram to phase-only hologram based on localized error diffusion and redistribution, Opt. Express 22 (2014) 5060–5066.
[28] X. Li, J. Liu, J. Jia, Y. Pan, Y. Wang, 3D dynamic holographic display by modulating complex amplitude experimentally, Opt. Express 21 (2013) 20577–20587.
[29] P.W.M. Tsang, T.-C. Poon, Novel method for converting digital Fresnel hologram to phase-only hologram based on bidirectional error diffusion, Opt. Express 21 (2013) 23680–23686.
[30] H. Song, G. Sung, S. Choi, K. Won, H. Lee, H. Kim, Optimal synthesis of double-phase computer generated holograms using a phase-only spatial light modulator with grating filter, Opt. Express 20 (2012) 29844–29853.
[31] J. Yeom, J. Hong, J.-H. Jung, K. Hong, J.-H. Park, B. Lee, Phase-only hologram generation based on integral imaging and its enhancement in depth resolution, Chin. Opt. Lett. 9 (12) (2011) 12009.
[32] P.W.M. Tsang, Y.T. Chow, T.-C. Poon, Embedding image into a phase-only hologram, Opt. Commun. 341 (2015) 188–193.
[33] P. Tsang, T.-C. Poon, Data-embedded-error-diffusion hologram, Chin. Opt. Lett. 12 (2014) 060017.

Please cite this article in press as: A.S.M. Jiao, et al., Restoration of digital off-axis Fresnel hologram by exemplar and search based image inpainting with enhanced computing speed, Computer Physics Communications (2015), http://dx.doi.org/10.1016/j.cpc.2015.03.018