A hybrid heuristic algorithm to improve known-plaintext attack on Fourier plane encryption

Wensi Liu,[1] Guanglin Yang,[1,*] and Haiyan Xie[2]

[1]*State Key Laboratory on Advanced Optical Communication Systems and Networks, Peking University, Beijing, 100871, China*

[2]*China Science Patent Trademark Agents Ltd., Beijing, 100083, China*

[*]*ygl@pku.edu.cn*

Abstract: A hybrid heuristic attack scheme that combines the hill climbing algorithm and the simulated annealing algorithm is proposed to speed up the search procedure and to obtain a more accurate solution to the original key in the Fourier plane encryption algorithm. And a unit cycle is adopted to analyze the value space of the random phase. The experimental result shows that our scheme can obtain more accurate solution to the key that can achieve better decryption result both for the selected encrypted image and another unseen ciphertext image. The searching time is significantly reduced while without any exceptional case in searching procedure. For an image of 64×64 pixels, our algorithm costs a comparatively short computing time, about 1 minute, can retrieve the approximated key with the normalized root mean squared error 0.1, therefore, our scheme makes the known-plaintext attack on the Fourier plane image encryption more practical, stable, and effective.

OCIS codes: (070.4560) Data processing by optical means; (200.3050) Information processing; (200.4560) Optical data processing; (060.4785) Optical security and encryption.

References and links

1. Ph. Réfrégier and B. Javidi, "Optical image encryption based on input plane and Fourier plane random encoding," Opt. Lett. **20**, 767-769 (1995)
2. L. G. Neto and Y. Sheng, "Optical implementation of image encryption using random phase encoding," Opt. Eng. **35**, 2459-2463 (1996).
3. O. Matoba and B. Javidi, "Encrypted optical memory system using three-dimensional keys in the Fresnel domain," Opt. Lett. **24**, 762-764 (1999).
4. O. Matoba and B. Javidi, "Encrypted optical storage with wavelength-key and random phase codes," Appl. Opt. **38**, 6785-6790 (1999).
5. E. Tajahuerce and B. Javidi, "Encrypting three-dimensional information with digital holography," Appl. Opt. **39**, 6595-6601 (2000).
6. G. Unnikrishnan, J. Joseph, and K. Singh, "Optical encryption by double-random phase encoding in the fractional Fourier domain," Opt. Lett. **25**, 887-889 (2000).
7. S. Liu, L. Yu, and B. Zhu, "Optical image encryption by cascaded fractional Fourier transforms with random phase filtering," Opt. Commun. **187**, 57-63 (2001).
8. J. Ohtsubo and A. Fujimoto, "Practical image encryption and decryption by phase-coding technique for optical security systems," Appl. Opt. **41**, 4848-4855 (2002).
9. G. Situ and J. Zhang, "Double random-phase encoding in the Fresnel domain," Opt. Lett. **29**, 1584-1586 (2004).
10. H. Suzuki, M. Yamaguchi, M. Yachida, N. Ohyama, H. Tashima, and T. Obi, "Experimental evaluation of fingerprint verification system based on double random phase encoding," Opt. Express **14**, 1755-1766 (2006). http://www.opticsexpress.org/abstract.cfm?URI=OPEX-14-5-1755.
11. R. Tao, Y. Xin, and Y. Wang, "Double image encryption based on random phase encoding in the fractional Fourier domain," Opt. Express **15**, 16067-16079 (2007). http://www.opticsexpress.org/abstract.cfm?URI=OPEX-15-24-16067.
12. X. C. Cheng, L. Z. Cai, Y. R. Wang, X. F. Meng, H. Zhang, X. F. Xu, X. X. Shen, and G. Y. Dong, "Security enhancement of double-random phase encryption by amplitude modulation," Opt. Lett. **33**, 1575-1577 (2008).

#110582 - $15.00 USD Received 24 Apr 2009; revised 25 Jun 2009; accepted 8 Jul 2009; published 27 Jul 2009

13. P. Kumar, A. Kumar, J. Joseph, and K. Singh, "Impulse attack free double-random-phase encryption scheme with randomized lens-phase functions," Opt. Lett. **34**, 331-333 (2009)
14. W. Stallings, *Cryptography and Network Security: Principles and Practice, 3rd ed.* (Prentice Hall, 2004).
15. A. Carnicer, M. Montes-Usategui, S. Arcos, and I. Juvells, "Vulnerability to chosen-ciphertext attacks of optical encryption schemes based on double random phase keys," Opt. Lett. **30**, 1644-1646 (2005).
16. U. Gopinathan, D. S. Monaghan, T. J. Naughton, and J.T. Sheridan, "A known-plaintext heuristic attack on the Fourier plane encryption algorithm," Opt. Express **14**, 3181-3186 (2006). http://www.opticsexpress.org/abstract.cfm?URI=OPEX-14-8-3181.
17. X.Peng, P.Zhang, H.Wei, and B.Yu, "Known-plaintext attack on optical encryption based on double random phase keys," Opt. Lett. **31**, 1044-1046, (2006).
18. X. Peng, H. Wei, and P. Zhang, "Chosen-plaintext attack on lensless double-random phase encoding in the Fresnel domain," Opt. Lett. **31**, 3261-3263 (2006).
19. Y. Frauel, A. Castro, T. J. Naughton, and B. Javidi, "Resistance of the double random phase encryption against various attacks," Opt. Express **15**, 10253-10265 (2007). http://www.opticsexpress.org/abstract.cfm?URI=OPEX-15-16-10253.
20. S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by Simulated Annealing," Science **220**, 671-680 (1983).
21. S. Nahar, S. Sahni, and E. Shragowitz, "Simulated annealing and combinatorial optimization," in *Proceedings of ACM/IEEE Conference on Design Automation* (Institute of Electrical and Electronics Engineers, New York, 1986), pp. 293-299.
22. M. Nieto-Vesperinas, R. Navarro, and F. J. Fuentes, "Performance of a simulated annealing algorithm for phase retrieval," J. Opt. Soc. Am. A **5**, 30-38 (1988).
23. F. J. O. Martinez, J. S. Gonzalez, and I. Stojmenovic, "A parallel hill climbing algorithm for pushing dependent data in clients-providers-servers systems," in *Proceedings of IEEE International Symposium on Computers and Communications* (Institute of Electrical and Electronics Engineers, New York, 2002), pp. 611-616.

1. Introduction

In recent years, many encryption algorithms based on double random phase (DRP) encoding have been proposed [1-13]. Several simulations and testing on DRP encryption system reveal that it is vulnerable to various attacks [15-19], such as the known-plaintext attack (KPA) and chosen-plaintext attack (CPA).

The CPA requires the attacker to obtain more resources and more control over the encryption system compared to the KPA [14,17], so the latter is more typical and thus studied more widely. However, obtaining the accurate key by exhaustive searching is almost impossible due to the huge key space. In practice, many heuristic algorithms are employed to obtain the approximated key, for example, the simulated annealing (SA) algorithm used in the KPA to retrieve the approximated key has achieved better experimental result [16]. However, the time cost of the SA remains considerably high when the size of plaintext is greater than 32x32 pixels [16,19], and there exists a considerable error when employed the key obtained through the SA algorithm to decrypt another unseen ciphered image.

In this paper, we focused on improving the accuracy of approximated key and speeding up the key searching procedure. We adopted a unit cycle to analyze the value space of the random phase. From the unit cycle point of view, the operation that perturbs one pixel of the random phase can be regarded as the rotation of corresponding point on the unit cycle.

We proposed a hybrid heuristic algorithm that combines the hill climbing algorithm [21,23] with the SA algorithm [16]. This hybrid algorithm remains the advantages of two algorithms, such as the simplicity of the hill climbing algorithm and the excellent performance in searching global optima of the SA algorithm, while overcomes their inherent weaknesses, such as being easy to fall into the local optimal solution of the hill climbing algorithm and the very long searching process of the SA algorithm. In our algorithm, we adopted the fixed perturbation value instead of the variable perturbation values in the SA algorithm, and some interesting properties are analyzed, such as the symmetry of perturbation error, and some heuristic rules are also adopted to speed up the searching procedure.

The rest of this paper is organized as follows. In Section 2 we recall the encryption algorithm with double random phase mask. In Section 3 the value space of random phase is

#110582 - $15.00 USD Received 24 Apr 2009; revised 25 Jun 2009; accepted 8 Jul 2009; published 27 Jul 2009

described. In Section 4 the hybrid heuristic algorithm is presented. In Section 5 experiments and discussion are given. In Section 6 conclusions are presented.

2. Encryption algorithm with double random phase mask

The double random plane encryption algorithm encodes an input image to a stationary white noise by employing two statistically independent random phase masks in input plane and Fourier plane. This process can be expressed mathematically as

$$\Psi(x) = \{f(x)R_1\} * \hat{R}_2 \tag{1}$$

where $\hat{R}$ denotes the Fourier transform of R, and $*$ denotes a convolution. $R_1 = \exp[i2\pi n(x)]$ and $R_2 = \exp[i2\pi b(v)]$ denote the random phase masks in input plane and Fourier plane, respectively. We assume that the input image is real-valued amplitude encoded [1], so we can decrypt a ciphertext image without knowing the key in the input plane [16,19]. Therefore, in the following discussion, the key need to be cracked only means the random phase R_2.

3. Value space of random phase

The random phase can be denoted by $\exp(i2\pi b)$, which b stands for independent white sequence uniformly distributed in (0, 1) [1]. The random phase has a constant modulus, so a unit circle in complex plane with radius 1 contains the value space of the random phase. Each pixel in random phase has the counterpart, which locates on the circumferences of the unit circle as shown in Fig. 1. The changes in the value of a pixel of random phase correspond to rotation of a point on the circumferences of the unit circle, for example, the value of a pixel, such as x_1+y_1i, is changed to x_2+y_2i corresponding to the point P1 sliding from the first quadrant to P2 of the second quadrant anti-clockwise as shown in Fig. 1. The fixed perturbation value in our hybrid algorithm can be regarded as a rotation angle in the unit circle, for example, perturbing of a pixel of the random phase by the value 0.01 corresponds to a 3.6-degree rotation of a counterpart on the unit circle anti-clockwise while -0.01 corresponds to a 3.6-degree rotation clockwise. We assume that P1, as shown in Fig. 1, is a pixel of initial guess random phase, and P2 is the final solution of P1. In order to find P2 we can perturb P1 by a perturbation value, i.e., a particular rotation angle, to obtain its neighbor, P1', then basing on P1' to find P1'', through such a series of iteration, we can asymptotically approach the final solution point P2.

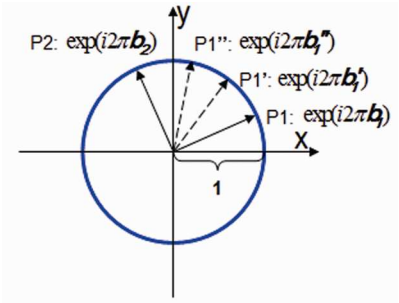

Fig. 1. Unit circle of the random phase.

4. Hybrid heuristic algorithm

4.1 Hill climbing algorithm

The hill climbing algorithm is a simple heuristic method for searching optimal solution to the problems that have huge solution spaces [21,23]. In this algorithm, the search starts from an initial guess solution generated randomly, and then bases on the current solution to find its neighbors. The current solution is replaced when a better one is found, and the search procedure continues from the better one in the same fashion, otherwise, generates randomly

#110582 - $15.00 USD Received 24 Apr 2009; revised 25 Jun 2009; accepted 8 Jul 2009; published 27 Jul 2009

another solution and begins to search. The search procedure will stop when the maximum iteration steps is exceeded.

4.2 Simulated annealing algorithm

The SA algorithm is a Monte Carlo method that is widely used to search global optima in various fields [16,20,22]. The procedure of the SA is performed as follows: An initial guess solution is generated and its corresponding cost function E^0 will be calculated, and then perturbs a pixel by a number chosen from a sequence of computer-generated random numbers uniformly distributed in the interval $(-\alpha,\alpha)$, which is called the scale of perturbation. The cost function E^{new} of the new tentative solution is calculated. The difference $\Delta E = E^{new} - E^0$ between E^{new} and E^0 is evaluated. The new tentative solution is accepted if $\Delta E < 0$, and begins a new round of iteration by randomly select a pixel to perturb. Otherwise, the change is accepted with a probability $\exp(-\Delta E / T)$ where T is the temperature parameter that is chosen sufficiently high at the beginning of algorithm. All pixels of the solution are perturbed in this way as many cycles as necessary until the algorithm converges for a given temperature T, for example, when three consecutive times at the end of a cycle the number of accepted perturbations raising E differ from the number of perturbations lowering E to within 5% [22], or $|\Delta E|$ is less than 5% of the initial value for each iteration [16]. The temperature T is decreased, and the procedure starts again. The search procedure is repeated until the error of the guess solution is reduced to a certain threshold value.

4.3 Impact of random phase perturbation on the decryption results

We studied the impact of perturbation of the single pixel and multi-pixels of the random phase on the final decryption results, and adopted the normalized root mean squared (NRMS) error as the criteria to evaluate the result [16]. The NRMS error is calculated as [16]

$$NRMS = \sqrt{\frac{\sum_{i=1}^{M}\sum_{j=1}^{N} | I_d(i,j) - I(i,j) |^2}{\sum_{i=1}^{M}\sum_{j=1}^{N} | I(i,j) |^2}} \tag{2}$$

where $I_d = |\tilde{f}|^2$ and $I = |f|^2$, $\tilde{f}$ denotes the estimated plaintext, and f denotes the original plaintext with the size of $M \times N$ pixels.

For single pixel perturbation, each pixel in the random phase is chosen sequentially, and perturbs it by a fixed perturbation value, the NRMS errors of two resulted images decrypted by the random mask before and after perturbation are recorded, and their difference is calculated. A series of perturbation values from 0.01 to 0.1 by interval of 0.01 are chosen for testing the errors of single pixel perturbation. The maximum and minimum differences of NRMS errors are shown in Fig. 2. In all cases, the minimum perturbation errors of all pixels of the random phase are almost close to zero, and the maximum perturbation errors increase linearly with the perturbation values changed from 0.01 to 0.1. From Fig. 2 we can find that the maximum perturbation errors only approach a considerably small value of 0.025, even though the perturbation value is equal to 0.1. The experimental result shows that the impact of single pixel perturbation on the final decrypted result is trivial in all these cases.

For multi-pixels perturbation, the pixels involved are perturbed by the value randomly chosen between the fixed positive perturbation value and the negative one. In experiment, three perturbation values, 0.01, 0.02, and 0.03 are chosen. As shown in Fig. 3, as the number of involved pixels increases the error also increases, it reaches the peak when all pixels are involved in perturbation, the maximum errors of three cases are 0.05, 0.10, and 0.16, respectively.

According to above analysis, we chose the fixed perturbation value, namely, only a positive perturbation value and a negative perturbation value instead of a scale of perturbation in the SA algorithm to speed up searching procedure, and to avoid oscillating highly. So it is crucial to choose a proper perturbation value to asymptotically approach the final solution to the key.

#110582 - $15.00 USD Received 24 Apr 2009; revised 25 Jun 2009; accepted 8 Jul 2009; published 27 Jul 2009

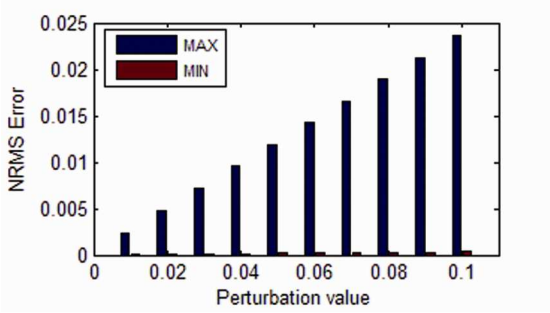

Fig. 2. Errors of single pixel perturbation.

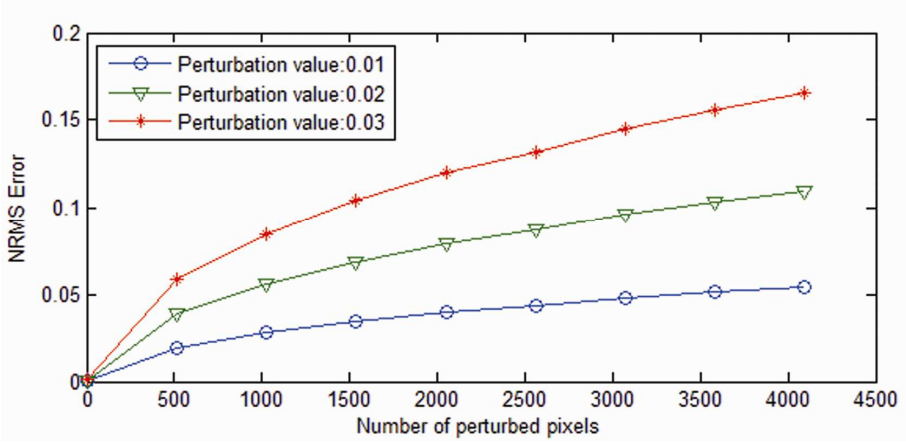

Fig. 3. Errors of multi-pixels perturbed by the value chosen randomly.

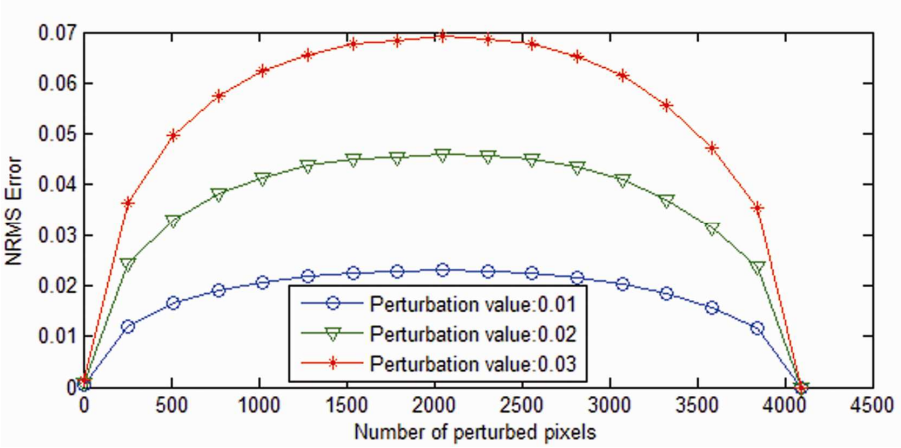

Fig. 4. Errors of multi-pixels perturbed by the same perturbation value.

4.4 Symmetry of perturbation error

The symmetry of perturbation error will occur when multi-pixels are simultaneously perturbed by the same perturbation value. As shown in Fig. 4, we can see that as the number of involved pixels increases, the error also increases, when nearly half of pixels of the random phase are perturbed, the maximum error will be produced, then as the number of involved pixels continue to increase, the error will be decreased until reaches zero.

#110582 - $15.00 USD Received 24 Apr 2009; revised 25 Jun 2009; accepted 8 Jul 2009; published 27 Jul 2009

All pixels of the random phase are performed by the same numerical operation, which can be regarded as a holistic rotation of all corresponding points on the circumferences of the unit circle, so it has no impact on NRMS error, and the decrypted image is exactly equal to the original plaintext image.

The symmetry of perturbation error suggests that there exist infinitely many equivalents of the original key, and the only difference among them is the rotation angle. So we only need to find any one of them, by which we can obtain the accurate decryption result.

4.5 Reduce the duration of equilibrium status

To analyze the curve changing in the cost function of SA algorithm, we have found that at the early stage of algorithm, the curve changing is sharp, as the iteration process continues, the change of cost function decreases, the scale of perturbation becomes more and more smaller, the curve changing is considerably gentle which is nearly a straight line. And this is the most time consuming process. In the SA algorithm, only one pixel is perturbed each iteration. According to the symmetry property of perturbation error as shown in Fig. 4, we added an operation into our algorithm that perturbs the multi-pixels simultaneously so as to avoid such stagnation problem and to speed up the convergence process. When the change of NRMS error is relatively small during a certain successive number of iteration, it means that the algorithm has already entered the equilibrium status. In order to reduce the duration and make rapid progress toward a solution, the multi-pixels perturbation will be performed. We adopted the mean standard deviation (MSTD) to measure the change in NRMS error. The NRMS errors of all iteration steps are recorded in processing, and a route that is used to judge whether the equilibrium status occurs will be started when the new NRMS error is bigger than the currently minimum error after each perturbation of the single pixel. The MSTD is defined as follow

$$MSTD = \frac{\left\{\frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^2)\right\}^{\frac{1}{2}}}{\left|\overline{x}\right|}, \qquad \overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{3}$$

where n is the number of elements in the sample, and we defined its value as half of pixels of the original plaintext image, x_i is the NRMS error at the i^{th} step. From Fig. 4 we can see, the error increases as the number of pixels involved grows, the maximum error is about 0.02 when the half of pixels of random phase are perturbed by 0.01, namely, the error changes from 0 to 0.02 in the interval (0, 2000). We set n to 2000 for our test images with the size of 64×64 pixels, the MSTD value calculated according to Eq. (3) is about 0.51, so the threshold value of MSTD can be defined as 0.5 when the perturbation value is 0.01, namely, if the MSTD of the latest n-element of the NRMS error vector is lower than 0.5, the equilibrium status is expected to occur. In such equilibrium status, we used the multi-pixels perturbation instead of the single pixel perturbation to search for the key, so as to avoid the local optima and time-consuming inefficient searching. A quarter of whole pixels are perturbed simultaneously and the new NRMS error is calculated to compare with the currently minimum error, the current solution will be replaced with the new solution when the new NRMS error is lower than the current error, otherwise, the new one will be accepted with a probability.

4.6 Known-plaintext attack using hybrid heuristic algorithm

For the hill climbing algorithm, the greatest advantage is its simplicity while suffers from premature convergence to local optima. For the SA algorithm, the main advantage due to the excellent performance in searching global optima while its computational cost is considerably high. In our hybrid algorithm, the advantages of two algorithms are still remained, such as the simplicity of the hill climbing algorithm and the excellent performance in searching global optima of the SA algorithm, while overcomes their inherent weaknesses, such as being easy to

#110582 - $15.00 USD Received 24 Apr 2009; revised 25 Jun 2009; accepted 8 Jul 2009; published 27 Jul 2009

fall into the local optimal solution of the hill climbing algorithm and the very long searching process of the SA algorithm.

In known-plaintext cryptanalysis, the attackers have a priori knowledge about the encryption algorithm and a plaintext-ciphertext pair is available [14,16-17]. We adopted the normalized root mean squared (NRMS) error as the criteria to evaluate the result [16]. This algorithm will be stopped when the NRMS error between the original plaintext image and the NRMS error of decrypted ciphertext image reaches the threshold value previously chosen. In experiments, the threshold value is set to 0.1.

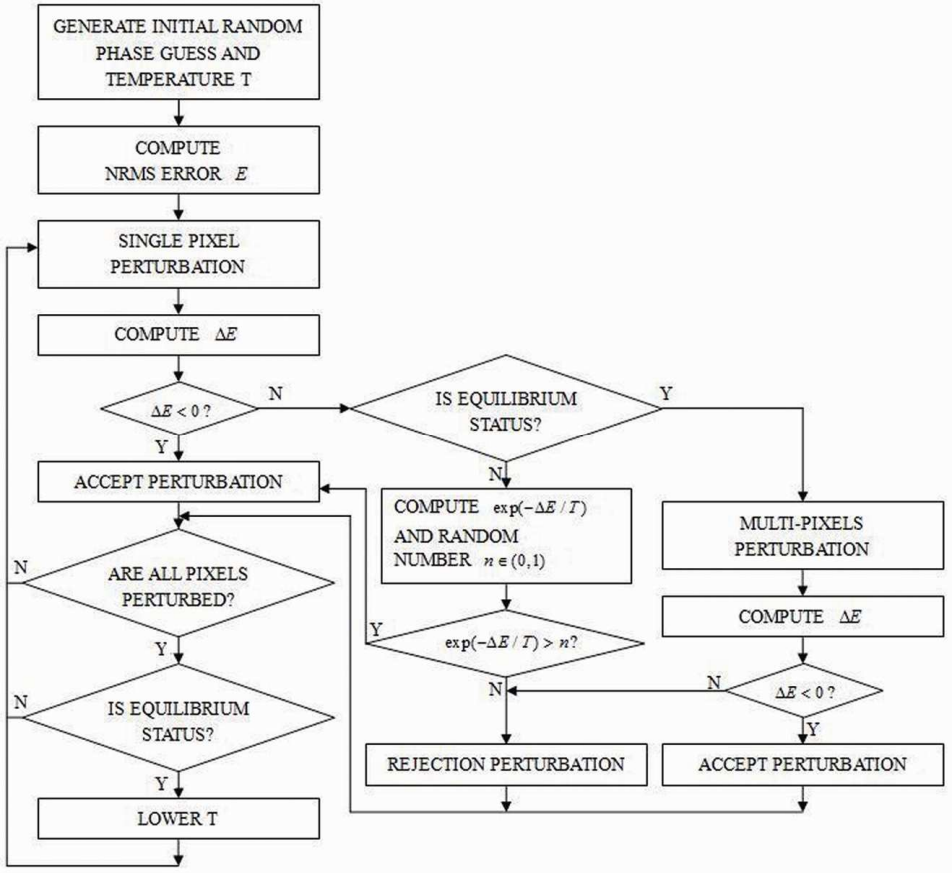

Fig. 5. Block diagram of the hybrid heuristic algorithm.

The block diagram of the algorithm is shown in Fig. 5, and the detailed procedures of steps are performed as follows:

Step 1: The initial guess random phase mask R_g is generated by assigning the phase of the Fourier transform of the encrypted image $\Psi(.)$ to every other pixel in both dimensions [22], the other half is chosen randomly from a sequence of computer-generated random numbers uniformly distributed in the interval $(0, 2\pi)$. The iteration step counter m is set to zero, and the threshold value of the NRMS error is set to 0.1. The initial temperature T_0 is initialized to a sufficiently high value [16, 22].

Step 2: The NRMS error E between the decrypted image and the original plaintext image is calculated.

Step 3: Perturbs single pixel by a fixed perturbation value, such as some values mentioned in Section 4.3. The new NRMS error E^{new} is calculated, and the difference between two errors

#110582 - $15.00 USD Received 24 Apr 2009; revised 25 Jun 2009; accepted 8 Jul 2009; published 27 Jul 2009

is evaluated as $\Delta E = E^{new} - E$. The perturbation is accepted if $\Delta E < 0$. Otherwise, the route mentioned in Section 4.5 that is used to judge whether the equilibrium status occurs will be started. The multi-pixels perturbation procedure will be performed if there exists equilibrium status in the process. Otherwise, the perturbation is accepted with a probability $\exp(-\Delta E / T)$.

Step 4: Steps 2 and 3 are repeated until all pixels of the random phase are perturbed and the equilibrium status occurs.

Step 5: The temperature is reduced according to the annealing schedule $T = T_0 \times 0.97$.

Steps 2 to 5 are repeated until the NRMS error between the decrypted ciphertext image and the original plaintext image is reduced to the threshold value 0.1.

5. Experiments and Discussion

In known-plaintext cryptanalysis, it is always assumed that the attackers are familiar with the encryption algorithm and a plaintext-ciphertext pair is available [14,16-17] as shown in Figs. 6(a), 6(b) and 6(c). In experiment, the initial guess random phase is generated by choosing randomly from an independent white sequence uniformly distributed in (0, 1). And we used a Dell Precision T5400 Intel Xeon CPU 2.5 GHz workstation with 4 GB RAM memory and MATLAB version 7 for our trials.

Firstly, we tested the time cost in the hybrid heuristic algorithm, the threshold value of the NRMS error is set to 0.1, and in Section 4.3 three numbers 0.01, 0.02, and 0.03 are chosen as the perturbation values to search the key. The average running time of 20 trials is 62 seconds when the perturbation value is 0.01, and the average running times are 75 and 237 seconds when the perturbation values are 0.02 and 0.03 as shown in Fig. 7, and the minimum time and maximum time in three cases are 57 and 68 seconds, 66 and 84 seconds, 206 and 251 seconds, respectively. As the perturbation value increases, the time taken to search the key also increases. The average searching time in the case of 0.01 is close to that of 0.02. However, the average searching time increases dramatically to three times when the perturbation value is 0.03. The main reason is that a bigger perturbation value will produce more fluctuation in cost function as shown in Figs. 2 and 3, so as to take much longer time to converge.

In these three cases and all of the 60 trials, our algorithm can converge to a solution without any exceptional trial that its running time greatly exceeds the average time. And the small difference in running time of each trial shows the robustness and stability of our algorithm. However, compared to our algorithm, on our specific computing platform, the average time of SA algorithm [16,22] taken to decrypt this image with NRMS error 0.1 is 4510 seconds, which is equivalent to 76 minutes, and the maximum time of the worst case is 131 minutes.

On the other hand, the NRMS error that decrypted another unseen image Ψ_B that encrypted using the same key as that of Ψ_A for 60 trials in the different perturbation values is shown in Fig. 8 and the image B is shown in Fig. 6(e). In these three cases, the average errors in decrypted the image B are 0.1137, 0.1306, and 0.1736, respectively. With the perturbation value changed from 0.01 to 0.03, the error in decrypted the image B also increases. And in each case, the decryption error is close to the average error, and there does not exist any exceptional error that is much bigger than the average error.

The plaintext images A and B are shown in Figs. 6(a) and 6(e), respectively. The real and imaginary parts of the complex-valued ciphertext images of A and B are shown in Figs. 6(b), 6(c), and Figs. 6(f), 6(g), respectively. The decrypted image of ciphertext images of A with an error of 0.1 is shown in Fig. 6(d) and that of B with an error 0.1193 is shown in Fig. 6(h).

We also employed other two images C and D which have more complicated texture and rich grayscale to test the decrypted results. The images C and its decrypted image with error 0.0916 are shown in Figs. 6(i) and 6(j), and the images D and its decrypted image with error 0.0952 are shown in Figs. 6(k) and 6(l). The decrypted images of Ψ_C and Ψ_D are both decrypted using the same key as that of Ψ_A. Although the errors of decrypted images of Ψ_C and Ψ_D are both lower than that of Ψ_A and Ψ_B, the decrypted images of Ψ_C and Ψ_D are

#110582 - $15.00 USD Received 24 Apr 2009; revised 25 Jun 2009; accepted 8 Jul 2009; published 27 Jul 2009

more blurry comparing to their corresponding original plaintext images. In order to improve the decrypted result of this type images, such as C and D, lower error is required.

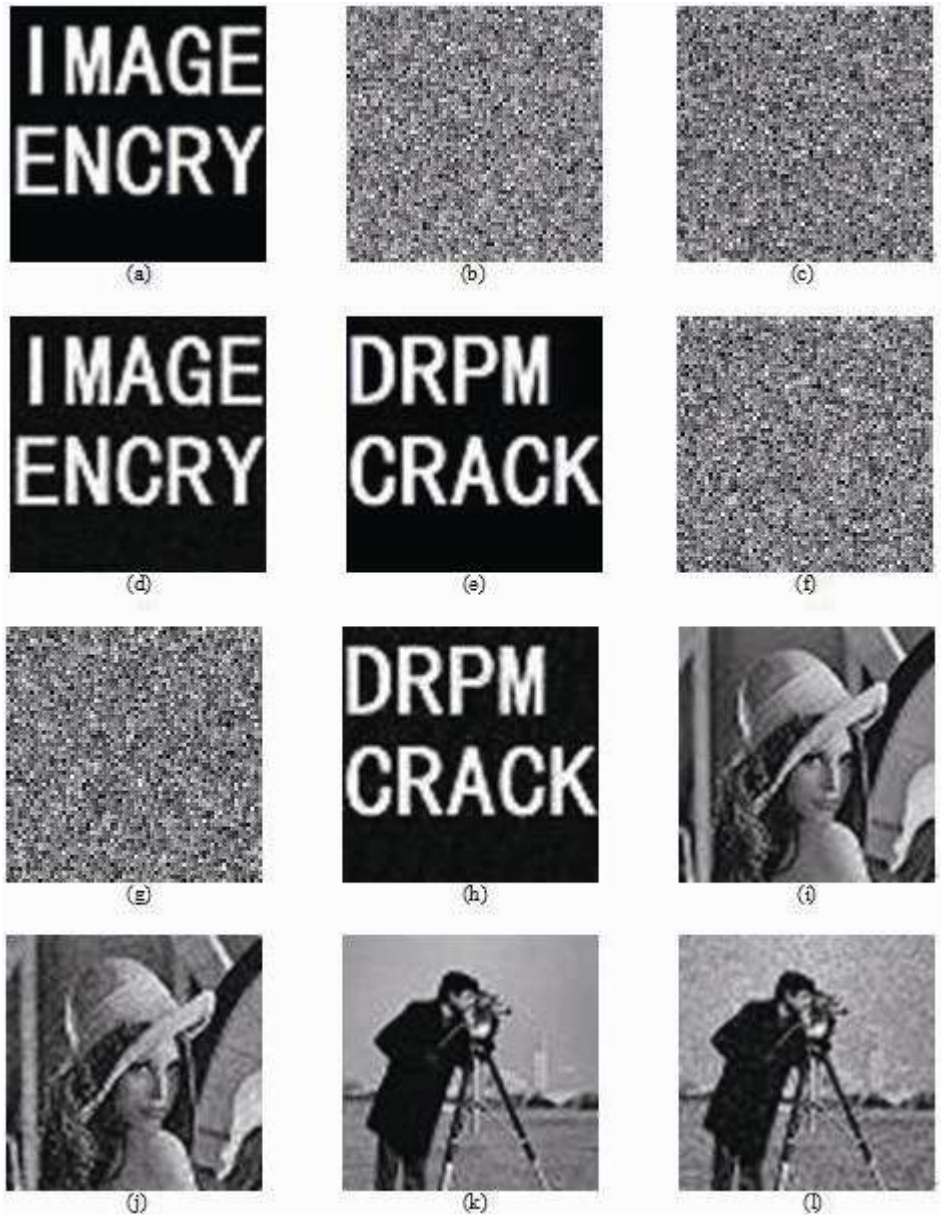

Fig. 6. (a) Original plaintext image A with 64×64 pixels, (b) the real part and (c) the imaginary part of the complex-valued encrypted image of A, (d) the decrypted image with an NRMS error of 0.1, (e) the plaintext B, (f) the real part and (g) the imaginary part of the complex-valued encrypted image of B, (h) the decrypted image B with an NRMS error of 0.1193, (i) Original plaintext image C with 64×64 pixels, (j) the decrypted image with an NRMS error of 0.0916, (k) Original plaintext image D, (l) the decrypted image with an NRMS error of 0.0952.

#110582 - $15.00 USD Received 24 Apr 2009; revised 25 Jun 2009; accepted 8 Jul 2009; published 27 Jul 2009

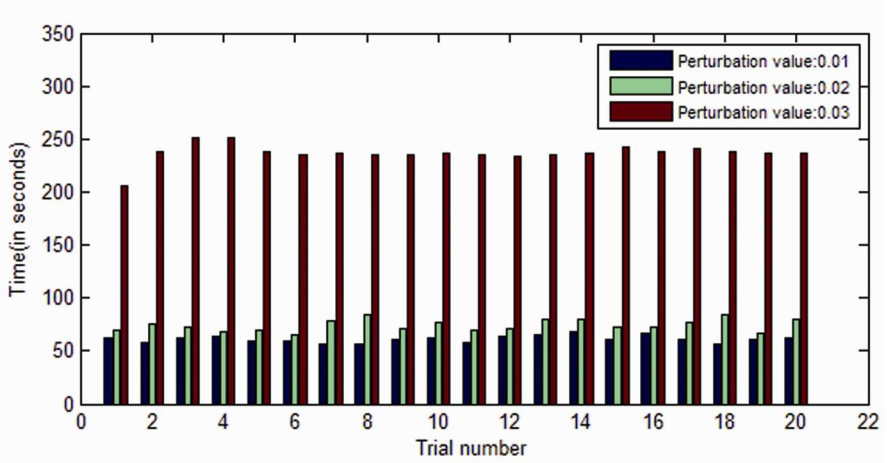

Fig. 7. Time cost of estimating the key used to encrypt image of A and with an NRMS error of 0.1 in the cases of three different perturbation values.

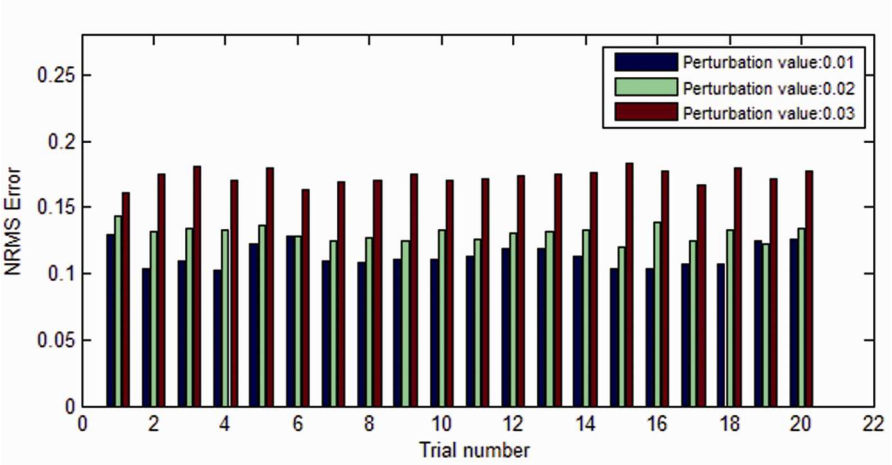

Fig. 8. NRMS error to decrypt the encrypted image of B using the same key that decrypts the encrypted image of A with NRMS error of 0.1.

The experimental results clearly show that 0.01 is the best among three perturbation values in both the searching time and the decryption error for other unseen ciphertext images. We also employed some other values smaller than 0.01, such as 0.005 and 0.001, to test the decrypted results. We found that the searching time taken to decrypt the Ψ_A for 0.001 is longer than that of 0.005, and both of them are longer than that of 0.01. However, when we adopt the decrypted image of Ψ_A with the NRMS error of 0.1 and corresponding approximated key to continue this search procedure until the algorithm cannot obtain any significant improvement in decrypted result, the NRMS error of decrypted image of Ψ_A for 0.001 is lower than that of 0.005, and both of them are lower than that of 0.01, which means the decrypted result using perturbation value 0.001 is better than that of 0.005 and 0.01. The reason is mainly due to a smaller perturbation value produce less fluctuation in cost function so as to take a longer time to converge while can obtain a better decrypted result. It is a tradeoff between the searching time and decrypted results for the choice of different perturbation values in our algorithm.

#110582 - $15.00 USD Received 24 Apr 2009; revised 25 Jun 2009; accepted 8 Jul 2009; published 27 Jul 2009

For other images with the larger size than 64×64 pixels, such as 128×128, the average time of our proposed algorithm taken to decrypt the image with the perturbation value 0.01 and NRMS error 0.1 is about 32 minutes. The time cost is a little high even though on our high-performance computing platform, which indicates the hybrid heuristic algorithm need to be optimized further to adapt to this situation.

6. Conclusion

We proposed a hybrid heuristic algorithm that combines the hill climbing algorithm and the simulated annealing algorithm to search the key, which can decrypt a ciphertext with a predetermined arbitrary low error. And a unit circle is introduced to analyze the intrinsic properties of the random phase and the impact of pixel perturbation on the final decryption results, and to employ the heuristic rules to speed up the searching procedure and to improve the decrypted results. The hybrid heuristic algorithm has shown the significant improvement that reduced both the searching time and the decryption error for other unseen binary and grayscale ciphertext images. All trials can obtain the approximated key using almost the same computational cost, which shows the stability and effectiveness of this algorithm.

Acknowledgments
This work was supported by the Natural Science Foundation of China under Grant No. 40672087.

#110582 - $15.00 USD Received 24 Apr 2009; revised 25 Jun 2009; accepted 8 Jul 2009; published 27 Jul 2009