Cascade cryptographic key-distribution protocol over a public network based on snapshot compressive ghost imaging

Wen-Kai Yu,* Shuo-Fei Wang, and Ke-Qian Shang

Center for Quantum Technology Research, and Key Laboratory of Advanced Optoelectronic Quantum Architecture and Measurement of Ministry of Education, School of Physics, Beijing Institute of Technology, Beijing 100081, China

(Received 3 February 2024; revised 24 April 2024; accepted 17 June 2024; published 12 July 2024)

Computational ghost imaging is widely used in classical cryptographic key distribution (CKD) because its measurement and reconstruction are equivalent to the encoding and decoding of signals. Although the measurement efficiency and key-generation rate can be improved by using snapshot compressive ghost-imaging technique, it is still hard to deal with the risk of private key leakage. In this paper, we propose a cascade snapshot CKD protocol over a public network. In each round of communication, multiframe binary block diagrams embedded with cryptographic keys, Arnold transform parameters (derived from the checksum data of recovered images in the previous communication), and users' authentication information are superimposed into one image frame that is measured within a single exposure of an array detector, here the Arnold transformation is applied to randomly disrupt the previously used modulation patterns according to the above parameters, which greatly enhances the nonstealability of private keys and makes the temporary intrusion of a fake identity nowhere to hide. Both simulation and experimental results have demonstrated the feasibility of this protocol and its ability to detect illegal attacks. We believe that the introduction of the cascade idea will accelerate the practicalization process of snapshot CKD.

DOI: 10.1103/PhysRevApplied.22.014029

I. INTRODUCTION

In the information age, information security is particularly crucial to ensure that information is not damaged, altered, or leaked due to malicious attacks. In this field, cryptographic keys (CKs) play a key role. Classical digital CKs mainly rely on NP-hard mathematical problems. With the rapid development of optical information technologies, optical CKs, which mainly rely on the randomness of optical physical quantities [1–3], are gradually entering people's vision. For example, quantum secure communication [4–6] utilizes the uncertainty, measurement collapse and nonclonability in quantum mechanics to ensure the security of CKs. In recent years, the fiber-based quantum key distribution (QKD) has been realized with a secure distance of 833.8 km [7], and some measurement-device-independent QKD schemes [8,9] have been developed to close the detection-related security loopholes. To further address the source-side security loopholes [10] and to eliminate all modulator channels, a fully passive time-bin encoding QKD system [11] has been proposed. However, due to the high cost of laying quantum channels and the low key-generation rate, quantum CKs are not suitable for real-time ultra-long-distance multiparty communication and have poor adaptability and compatibility with traditional public networks. Borrowing from the classical Vernam's "one-time pad" idea (where distributed CKs are used only once and then discarded) and the asymmetric encryption architecture [12–14] (which employs private key and public key pairs), a computational ghost-imaging-(CGI) [15–22] based public network cryptographic key-distribution (CKD) protocol is proposed. The CGI evolved from double-arm ghost imaging (GI) [23–28], and its principle lies in the use of a programmable spatial light modulator (SLM) to remove the reference arm, and then using bucket (single-pixel) measurements to acquire the total light intensities of the modulated light field, where the object image cannot be retrieved from either the modulation patterns or the bucket values alone, but can be recovered via the correlation between the two. The modulation and reconstruction processes perform encoding and decoding of the signal [29–31]. If the modulation patterns are shared to the legitimate users in advance as part of private keys, then in each communication the server can send the encrypted bucket-value sequence to the users over a public network, and each user, after receiving the sequence, can combine it with the modulation patterns on hand to reconstruct the original image and extract the distributed CKs [32,33]. If the private keys are secure enough, then the whole protocol is secure enough. Since the CKD is done directly over the public channels, it allows multiparty communication and eliminates the risk of photon loss.

*Contact author: yuwenkai@bit.edu.cn

Recently, many optimized schemes have been proposed for CGI-based CKD, among which Zhao *et al.* [34] used fractional Fourier transform to further encrypt the encoded signal for the second time, enhancing the information concealment; Zhang *et al.* [35] applied the Radon transform to reduce the amount of data transmission and improve the image-reconstruction quality; Yu *et al.* [36] and Zhang *et al.* [37] exploited fragment pattern splicing and cumulative visual cryptography to realize interactive authentication, respectively; Zhao *et al.* [38] and Yu *et al.* [39] proposed to utilize quick-response code (QR) to increase the key generation rate. However, these CGI-based CKD protocols are based on single-pixel measurements and can only encode a single image frame at a time, which severely limits the key-generation rate. Additionally, the private keys used in all communications are always the same, and the encoding processes involved in two adjacent communications are independent of each other, which limits the protocol security to a certain extent. Once an illegal user intervenes halfway by forging the identity, he or she can silently eavesdrop CKs in the subsequent CKDs without the need to know the results of previous communications, which is a serious security risk for the protocol itself that should not be neglected. The above problems seriously hinder the practicalization process of optical CKD. In recent years, the development of snapshot compressive-imaging (SCI) technology has gradually matured [40], which provides a new idea to solve the above problems. This is because it allows an overlapping encoded diagram of multiframe images to be recorded with a single exposure of a low-speed array detector. In our previous work [41], we have proposed a snapshot cryptographic key-distribution (SCKD) protocol over a public network with joint authentication, where the CKs and authentication information are embedded into multiframe color block diagrams with fixed modulation patterns (private keys) during each round of the CKD, and then the signal is encrypted using multipixel single-exposure measurements, which effectively improves the CK generation rate and reduces the sampling time. However, private key leakage can lead to catastrophic security risks in the event of identity impersonation. Imagine that if a transform is performed on the modulation patterns, it allows the private keys to be updated before each round of encoding, and can build a bridge to link with previous CKD accordingly.

In this work, we propose a cascade CKD protocol over a public network under the SCI framework, in which the server embeds the CKs, Arnold transform parameters and users' authentication information into multiframe binary block diagrams, and then takes a single-exposure photo of the superimposed diagram. Each legitimate user recovers multiframe block diagrams from the received superimposed image [i.e., encrypted pattern (EP)] according to the updated modulation patterns and extracts the CKs. In this way, it also achieves a fast encoding and a significant increase in the key-generation rate, inheriting the advantages of the previous approach [41]. The color block diagrams are replaced by binary ones to streamline the encoding process. Here, we specifically design a cascade mechanism based on the Arnold transform and embed its parameters into the binary block diagrams to update the modulation patterns for each round of the CKD. Here, the parameters are taken from the checksum data of the reconstructed results of the previous communication. The protocol implements a cascade coding of the CKD, which not only increases the nonstealability of the private keys, but also leaves no room for the temporary intervention of fake identities.

II. METHOD: CASCADE SNAPSHOT PUBLIC NETWORK CRYPTOGRAPHIC KEY-DISTRIBUTION PROTOCOL

A. Principle of snapshot compressive imaging

The SCI uses a low-speed camera to capture an integral image of video frames modulated by high-frequency switching patterns, and then applies an algorithm to decode these frames. In terms of the mathematical model, the video frames do not require continuous changes.

In this paper, we build the cascade snapshot CKD protocol based on this measurement model. As shown in Fig. 1, we use multiframe black and white block diagrams as the image frames to be sampled, which contain multiple CK extraction patterns (CKEPs) and a self-authentication pattern (SP). Let each frame of the black and white block diagram X_i $(i = 1, 2, \ldots, \tau)$ consist of $\zeta \times \eta$ pixel units, each of which contains $u \times v$ pixels (all of the same pixel value). Then, the actual pixel size of the block diagram is

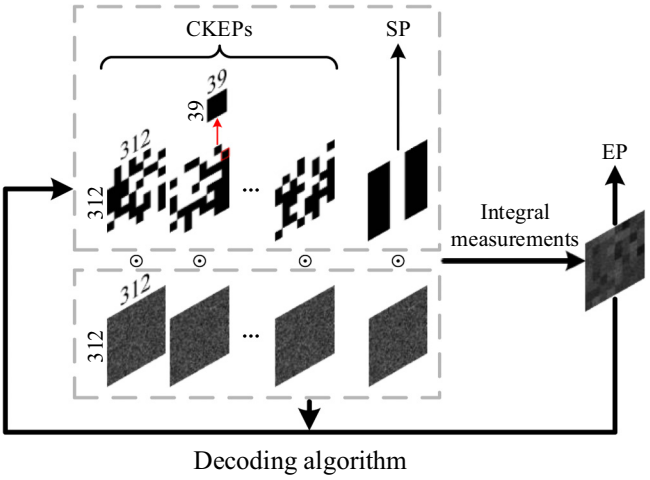

FIG. 1. Snapshot measurement process involved in our cascade snapshot public network cryptographic key-distribution protocol. CKEP, cryptographic key-extraction pattern; SP, self-authentication pattern; EP, encrypted pattern; ⊙, matrix dot-product operator.

$p \times q$, where $p = \zeta \times u$ and $q = \eta \times v$. Without loss of generality, here we set $\zeta = \eta = 8$, $u = v = 39$, and $p = q = 312$. Each frame of the block diagram must be optically modulated individually, which means that τ-frame block diagrams require τ-frame different modulation patterns φ_i. The camera records the overlapped image (i.e., EP) Y of the τ-frame modulated diagrams within a single-exposure interval, i.e., each pixel on the superimposed image is actually the pixel-value integral result of the τ-frame modulated diagrams at that pixel. The pixel sizes of X_i, φ_i, and Y are all the same. The above measurement process can be expressed by the mathematical formula:

$$Y = \sum_{i=1}^{\tau} \varphi_i \odot X_i, \tag{1}$$

where $\odot$ stands for the matrix dot-product operator. The expression in Eq. (1) can be rewritten by the following vectorized linear equation:

$$y = \Phi x, \tag{2}$$

where $y = \mathrm{Vec}(Y) \in \mathbb{R}^{pq \times 1}$ denotes a one-dimensional (1D) measurement column vector reshaped from Y, Vec is an operator for the vectorized representation of the matrix (i.e., connecting the rows or columns of a matrix end to end to form a column vector), and $x = \mathrm{Vec}(X) \in \mathbb{R}^{pq\tau \times 1}$ denotes the 1D column vector obtained by stretching the multiframe block diagrams and stitching them together. The measurement matrix $\Phi \in \mathbb{R}^{pq \times pq\tau}$ follows a diagonal array structure:

$$\Phi = [\mathrm{diag}\,(\mathrm{Vec}\,(\varphi_1)), \ldots, \mathrm{diag}\,(\mathrm{Vec}\,(\varphi_\tau))], \tag{3}$$

where diag denotes the diagonalization operator, which places the vector elements on the diagonal of a matrix,

$$\mathrm{diag}(c) = \begin{bmatrix} c_1 & 0 & \cdots & 0 \\ 0 & c_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & c_L \end{bmatrix}, \tag{4}$$

L is the length of the vector c. We can see that according to the known quantities y and Φ (with nonfull rank), solving x is an ill-posed problem, and can be modeled as

$$\hat{x} = \arg\max_x \frac{1}{2}\|y - \Phi x\|_2^2 + \lambda g(x), \tag{5}$$

where λ is a parameter to balance the fidelity term and the implicit prior $g(x)$, and $\|\cdot\|_2^2$ denotes the square of the l_2 norm defined as $\|\kappa\|_2^2 = \sum_{j=1}^{N} |\kappa_j|^2$, N is the length of the vector κ. Noting that the block diagrams are usually sparse or compressive, we can use compressed sensing algorithm to solve the problem. The algorithm used here is total variation minimization by augmented Lagrangian and alternating direction algorithm (TVAL3) [42].

B. Arnold transformation

The Arnold transformation has the advantages of low computational cost, large linear mapping space, and easy implementation, and has been successfully used for image encryption [43,44]. Performing the Arnold transform on a matrix is equivalent to rearranging the element positions in this matrix. Assuming $p = q$, the Arnold transform can be written as

$$\begin{bmatrix} x_{m+1} \\ y_{m+1} \end{bmatrix} = \begin{bmatrix} 1 & b \\ a & ab+1 \end{bmatrix} \begin{bmatrix} x_m \\ y_m \end{bmatrix} \bmod (S), \tag{6}$$

where x_m, y_m, and x_{m+1}, y_{m+1} denote the pixel positions before and after performing the Arnold transform, respectively; a and b are the transform parameters; m is the number of times the Arnold transformation is performed, taken as 1 in this work; $S = p = q$; and mod represents the modular operation.

C. Cascade mechanism

The cascade process of this protocol is illustrated in Fig. 2. Since each block diagram corresponds to a modulation pattern, in the first round of communication we embed the parameters a and b of the Arnold transform corresponding to each modulation pattern into each-frame CKEP. The parameter-embedded block diagrams are then encoded by the initial keys (modulation patterns, also regarded as part of private keys) according to the SCI measurement model, and the encrypted superimposed image is transmitted over the public channels to legitimate users.

In the second round of communication, both the server and legitimate users first perform the Arnold transform on the modulation patterns using the transformation parameters extracted from the decoded multiframe block diagrams

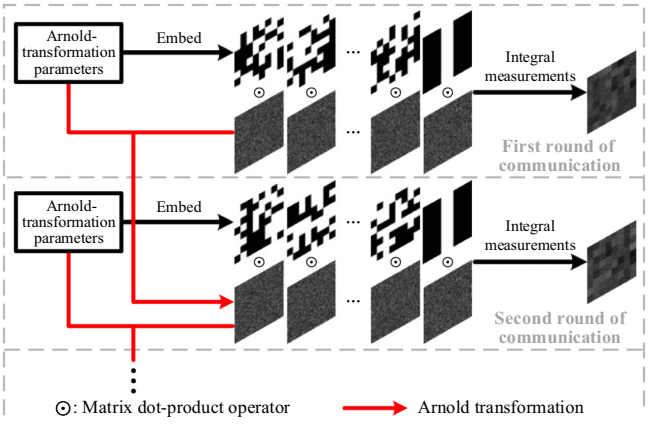

FIG. 2. Schematic diagram of cascade mechanism. The Arnold transformation parameters embedded in the subframes will be extracted and fed into the Arnold transformation to update the modulation patterns to be used in the next round of communication.

of the first round of communication, and the new modulation patterns after the transformation will update the private keys. The server likewise needs to embed the next set of the Arnold transform parameters into new CKEPs, then encode all the block diagrams with the updated modulation patterns, and retransmit the EP to the users. The above operations need to be repeated for each subsequent communication.

The modulation patterns used in each communication need to be updated accordingly based on the transformation parameters obtained from the previous communication, establishing the connection between two back-and-forth communications, which is called cascading. In this way, the memory can be used as the decoding basis for the current CKD. In addition, this mechanism allows the server and legitimate users to share the initial keys only once but ensures that the modulation patterns used in subsequent communications are different from each other. When an illegal attacker intervenes by impersonating the identity, even if he or she steals the private keys, he or she will still be unable to decipher the subsequently distributed CKs without knowing the decoding results of the previous communication, which greatly enhances the security of the protocol.

D. Cascade snapshot public network cryptographic key-distribution protocol

As shown in Fig. 3, our protocol consists of three parts: (1) preparation of initial keys; (2) cascade encoding of block diagrams and distribution of encrypted signals over the public network; and (3) users' authentication with CK and next Arnold-transformation parameter extraction.

Part I: preparation of the initial keys

(1) Initial private key sharing. The server needs to generates t sets of initial private keys and distributes them to t legitimate users via absolutely secure private media (e.g., flash card or U-shield). The initial private keys for each user contain τ modulation patterns each of $p \times q$ pixels, random seeds, etc. For simplicity and without loss of generality, we set $t = 4$ here.

(2) Initial Arnold-transformation parameter embedding. In the first round of communication, the server assigns different Arnold-transformation parameters a and b to each frame of CKEP and embeds them into it.

Part II: encoding of block diagrams and distribution of encrypted signals over the public network

(1) Private key update for current communication. In the first round of communication, the user decodes multiframe block diagrams based on the initial private keys on hand, using a fixed region of pixel units in each CKEP frame to extract the parameters a and b needed for the next round of the Arnold transform. In the second round of communication, both the server and legitimate users need to perform the Arnold transform on each frame of the modulation

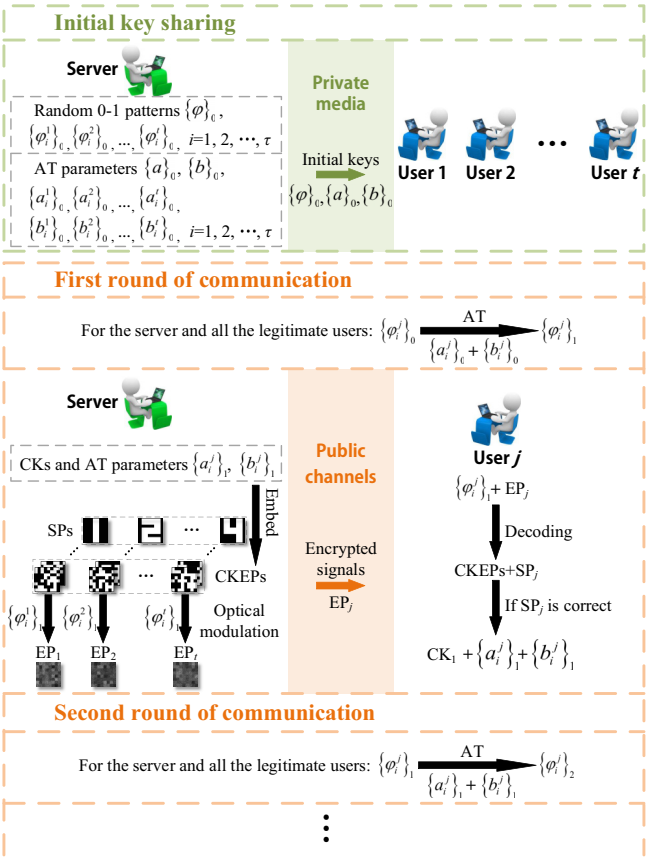

FIG. 3. Schematic diagram of snapshot cryptographic key-distribution protocol with cascade mechanism over a public network. CK, cryptographic key; AT, Arnold transformation.

pattern using the extracted parameters to update the private keys.

(2) Block diagram preparation and embedding of the Arnold-transformation parameters. In each round of communication, the server needs to prepare $\tau - 1$ frames of CKEPs and one frame of SP (containing the user's identity information) to be sampled, and there are τ frames of block diagrams in total. Then, the server embeds the Arnold-transformation parameters a and b extracted for the current transform into a specific pixel region of each frame of CKEP, respectively.

(3) Encoding and public network distribution. The server performs high-frequency modulation on the prepared block diagrams using the updated modulation patterns, and single-exposure records the overlapped image (i.e., EP), which is then converted into a one-dimensional binary bitstream and sent to the corresponding legitimate user over the public network.

Part III: users' authentication with CK and next Arnold-transformation parameter extraction

(1) Decoding of multiframe block diagrams. After receiving the encrypted one-dimensional binary bitstream, each user needs to convert it into EPs. Except for the first

round of communication where the initial private keys are used for decoding, in all other communications each user utilizes the updated private keys for decoding, to obtain all the CKEPs and SP. Here, to improve the quality of the decoding, the transcoded EP needs to be smoothed.

(2) Users' self-authentication and extraction of CKs and transformation parameters. Each user can verify that his or her respective identity information is correct by referencing the decoded SP. If it is correct, the user can proceed to extract the CKs and Arnold-transformation parameters from the decoded CKEPs. If not, a new round of the CKD will be performed.

E. Details for extraction of the CKs and Arnold-transformation parameters

The guidelines for extracting the CKs and Arnold-transform parameters are illustrated in Fig. 4. To exemplify, Figs. 4(a)–4(b) and 4(c) present the CKEPs and SP prepared for user 1, respectively, while Fig. 4(d) shows the EP obtained by integral sampling. The corresponding decoded CKEPs and SP are provided in Figs. 4(e)–4(f) and 4(g), respectively. These results should be converted into binary block diagrams by smoothing operations, as shown in Figs. 4(h)–4(j). When verifying the accuracy of the self-authentication block diagram [see Fig. 4(j)], the user can extract the CKs and Arnold-transformation parameters from the smoothed CKEPs [see Figs. 4(h)–4(i)].

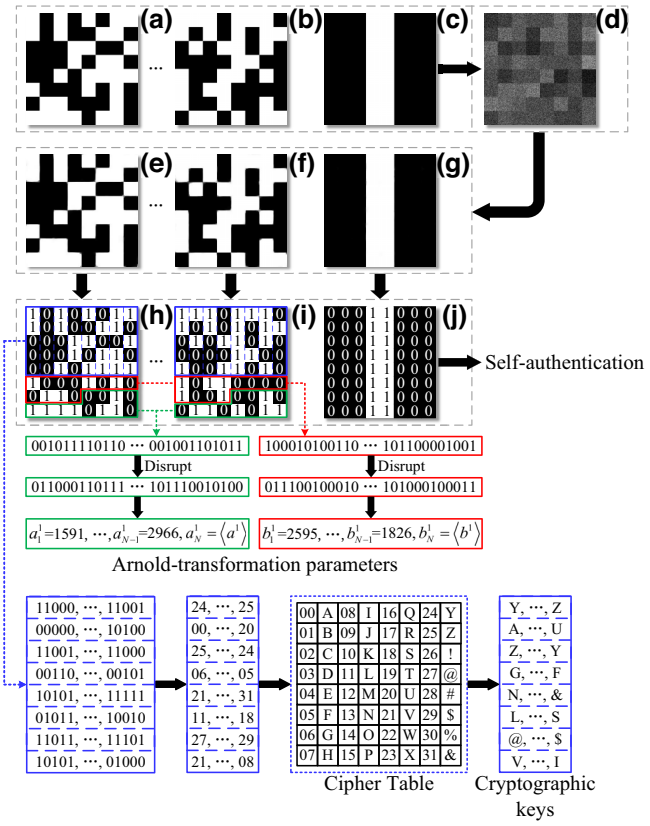

FIG. 4. Extraction rules of the CKs and Arnold-transformation parameters. (a)–(c) are the original CKEPs and SP, respectively, and (d) is the EP. (e)–(g) are the smoothed images of the CKEPs and SP after decoding. (h)–(j) are the corresponding binary matrices of (e)–(g), in which (h)–(i) are used for the Arnold-transformation parameter and CK extraction while (j) is used for self-authentication.

The method for extracting the transformation parameters is clearly described as follows. Each CKEP frame contains 12 binary digits in its green box and there are $12(\tau-1)$ binary digits in $\tau-1$ frames of CKEPs. The 12 bits in each CKEP are arranged in consecutive order and form a spatially contiguous segment within the CKEP frame. To enhance security and prevent the possibility of machine-learning image matrix completion guesswork, these bits are randomly scrambled according to the preshared random seed in the initial private keys. The binary bits are in groups of 12 bits and there are $\tau-1$ groups in total. After converting the 12 bits in each group to decimals, each user gets $\tau-1$ decimals. These decimals are then averaged and rounded up to produce the τth decimal number. In this way, τ decimal numbers correspond to the total number of modulation patterns, and each decimal number is regarded as the parameter a of the Arnold transformation performed on each modulation pattern. Similarly, each user can also extract the parameters b from the red boxes of the CKEPs.

The CK extraction rules are given as follows. The binary numbers in the blue rectangle of each CKEP frame are used to extract the CKs distributed in this round of communication. Every 5 bits are grouped together, no need to disrupt, will correspond to $0 \sim 31$ decimal numbers. We have designed a 32-cipher table consisting of 26 uppercase letters and six special characters, as shown in Fig. 4. Here, each frame of CKEP consists of 8×8 pixel units, the first five rows of which can be used for key extraction. In the same 5-bit region of the CKEPs, there are a total of $\tau-1$ 5-bit groups, and these groups can be concatenated into a row. The $\tau-1$ 5-bit groups in the next region of CKEPs are concatenated to form a second row, and so on, until a complete array of bits is formed, which is then translated into a character matrix of $8 \times (\tau-1)$. This matrix is stretched into a distributed CK string in row-major order.

Since the CKEP is a block diagram, there will be some connected blocks. Therefore, there is a certain possibility that the 12 bits in the green box and the 12 bits in the red box of each frame of the CKEP can be guessed according to the correct adjacent blocks or rows by using machine-learning-based matrix completion. As mentioned earlier, we use the strategy of randomly disrupting these bits according to the preshared random seed in the initial private keys to eliminate this effect. Of course, the illegal attacker can also use machine learning in the same way

to guess the remaining bit region of each CKEP frame. But, under our cascade strategy, the attacker is not able to fully access the updated modulation patterns of the current CKD, which will be proved later in Sec. IV A. It means that the attacker cannot decode exactly the correct CKEPs. And machine-learning-based matrix completion must rely on the correct bits, which is the critical point. The situation where the adjacent blocks or rows are correctly guessed is only possible when the number of bits to be guessed is small. However, there are a large number of bits used for CK extraction, the attacker cannot know which bits are correct, even if he or she decodes a result close to the correct CKEP. Then, the attacker can only guess $\prod_i C_{40}^i$ times to extract the CKs from 40 bits in each frame of the CKEP, C is a combinatorial notation, $i = 1, 2, \ldots, 39$. This is an extremely large number of exhaustions, and each of which will consume the computation time of machine learning once. In addition, checking the authenticity of the results of machine-learning-based matrix completion can be done in nonpolynomial time. Given this, the bits (even without disrupting) used to extract the CKs cannot be guessed by using machine-learning methods, which ensures the security of the extracted CKs.

III. EXPERIMENTAL RESULTS

The optical configuration of the snapshot acquisition used by the server is shown in Fig. 5. The thermal light emitted from a tungsten halogen lamp is expanded and collimated onto reflective object frames, which are then imaged onto a digital micromirror device (DMD), and finally imaged onto the detection surface through an imaging lens. The server uses a camera to sample the modulated light field integrally within a single exposure. Without loss of generality, we set $\tau = 12$ here. Since there is a need to quickly switch between 11 frames of CKEPs and 1 frame of SP, an easy way is to use another DMD to act as a display of object frames. During synchronization, it is necessary to display 12 object image frames in sync with the modulation patterns and to display them all within a single-exposure time interval. As the bottom edge of the DMD is at a 45° angle to the horizontal plane, the image captured by the camera is a rhomboid pattern that must undergo a series of image preprocessing operations such as rotation, transposition, and stretch distortion correction before being converted to a binary bitstream and sent to legitimate users over public channels.

It is worth mentioning that the snapshot compressive GI system can be expressed by Eq. (1), and using Eq. (1) to directly generate the EP seems much easier and more

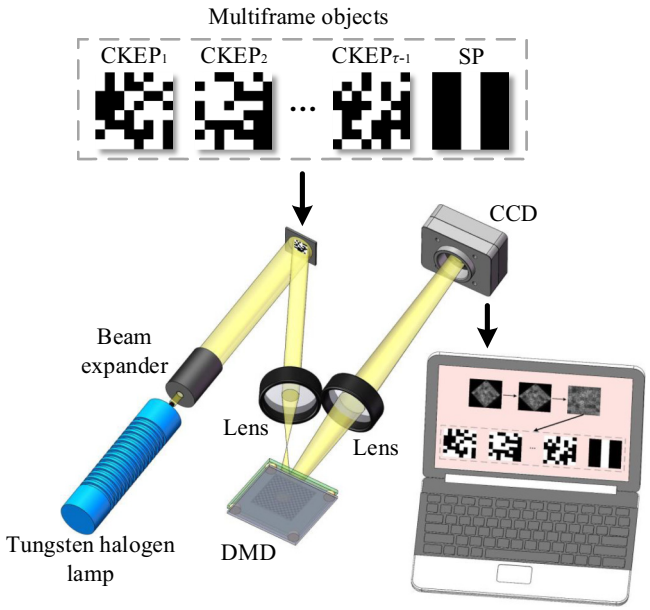

FIG. 5. Experimental setup of our protocol. The thermal light illuminates the reflective object image frames, which are imaged onto a digital micromirror device (DMD) and finally captured by a charge-coupled device (CCD) within a single exposure.

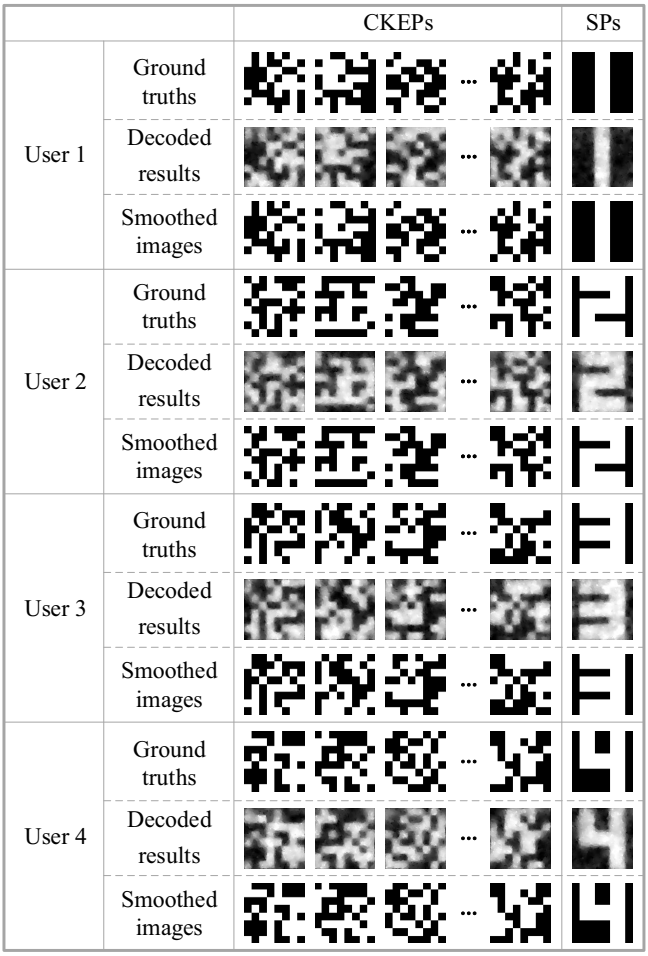

FIG. 6. Experimental results of CKEPs and SPs reconstructed by four legitimate users. Here, we present the ground truths, decoded results, and smoothed images of four users.

stable, but the measurement model is relatively fixed and there is a potential possibility that the modulation patterns are partially guessed. In contrast, with the snapshot compressive GI system, the actual integral measurement process inevitably involves the impact of a number of factors, such as optical convolution, point-spread function, optical defocus, lens aberration, turbulence, temperature drift jitter of the light source, stray light, camera bit depth, pixel grayscale value discretization, and so on. These are very complex optical and electrical interactions, and many of them have no fixed exact physical models and cannot be imitated by computer simulation. With these complex interactions, it becomes more difficult for the illegal attacker to guess the private keys including the modulation patterns from the EP, which significantly increases the security of the whole cascade CKD protocol. This is the reason why we use the snapshot compressive GI system here instead of directly using Eq. (1) to generate the EP.

For demonstration purposes, we collected data for four users. The original, decoded, and smoothed images of 11-frame CKEPs and one-frame SP in each group are shown in Fig. 6. It can be seen that the smoothed SPs all match with the original SPs perfectly, indicating that the self-authentication is successful, and at this point the users can perform the subsequent extraction operations of the CKs and Arnold-transform parameters.

IV. SECURITY ANALYSIS AND ATTACK DETECTION

A. Security analysis of private keys

The next step is to prove the security of this protocol. We assume that Eve, the eavesdropper, can listen to the bitstream transmitted over the public network and is able to employ all advanced techniques to steal the distributed keys. Since the private keys are inaccessible, Eve can only guess them by brute-force exhaustion. Taking $\tau = 12$ and $p = q = 312$ used in this paper as an example, if Eve wants to correctly decode 11 frames of CKEPs and one frame of SP distributed for a single user, he or she must accurately guess 12 frames of 0–1 312×312 modulation patterns, which requires $2^{312\times312\times12} \approx 10^{351642}$ exhaustive attempts. Each attempt must be verified by decoding the image frames, and all the attempts will take much more time than a single communication time. Moreover, this protocol is in one-time pad and cascaded, in each round of communication, the CKs to be distributed are different and the private keys are updated in real time, so even if Eve is lucky enough to guess the private keys correctly once, he or she will have to reguess the private keys again in the next round of communication, so this protocol is sufficiently secure.

Now, let us imagine that there exists a very dangerous situation: Eve knocks out one of the legitimate users (e.g., user 2) and directly loots the private keys held in his or her hand. Since the random seed used in our protocol can generate only the current random sequence in one direction, and cannot inversely reacquire the random sequence already generated in the previous round of communication, Eve cannot know about the previous disrupted order of the bits generated by the random seed. Given that the private keys are updated in cascade in each round of communication, Eve must guess all the Arnold-transformation parameters a and b to be used in the current round of communication. There are 12 sets of the Arnold-transformation parameters in total. Since the last set of parameters is obtained by averaging, it is only necessary to accurately guess 11 sets of parameters a and b. In the demonstration example given above, each parameter corresponds to a 12-bit binary number with 2^{12} possibilities. So to guess all the parameters correctly, Eve will have to search exhaustively at most $(2^{12})^{22} \approx 10^{79}$ times. Here again, Eve has to decode the image frames so as to judge whether the guess is successful or not according to the corresponding decoding results. However, the recovery process of single-exposure compressed imaging is an ill-posed problem, thus applying any reconstruction algorithms is faced with an extremely high computational complexity. At present, the most powerful computing units, such as supercomputers and quantum computers, are effective for special algorithms, but are ineffective in speeding up the solving of this kind of ill-posed problem, and completing the decoding of several image frames within 1 ms is already at their upper speed limit. By this extrapolation, the time for Eve to validate all the parameters is at least $10^{79} \times 10^{-3}$ s $= 10^{76}$ s $\approx 3 \times 10^{68}$ years, and this time spent is still very impressive. This makes it impossible for Eve to continue eavesdropping on the distributed keys by pretending to be someone else.

Certainly, Eve can also use other advanced techniques to minimize the number of exhaustive guesses. For example, Eve can complement the pixel units used to extract transformation parameters by matrix filling or machine learning based on the partially known correct pixel units. Since the filling process is algorithm dependent, there is no guarantee that the values of the filled pixel units are correct, but the number of exhaustive guesses can be reduced to some extent. It should be noted that this is a relatively extreme case, and in fact Eve is not sure which pixel-unit values are correct in the image frames he or she reconstructs. It makes no sense to fill in or extrapolate neighboring pixel-unit values by assuming the correct ones. In addition to this, Eve could certainly start with the probability distribution of the Arnold-transform parameters. For example, 11 sets of transformation parameters a and b extracted from the CKEPs take values ranging from 1 to $2^{12} - 1$, following a random distribution, thus the mean values of the parameters a and b are predicted to be around 2048. With the help of this prior knowledge, Eve can reduce the number of exhaustive guesses. It should be noted, however, that

the above assumptions are only valid in the large sample space, when the number of image frames is small, the parameter averages will also exhibit a large random fluctuation, making it difficult for Eve to further narrow the range of exhaustive parameter values. Moreover, the server can also make the parameter mean deviate from the theoretical statistical mean estimate by artificially designing each subframe. The transformation parameters obtained through exhaustive guesses require image reconstruction for verification, which is still much more time consuming than each communication time.

During Eve's guessing of the Arnold-transformation parameters, it is quite possible to have a set of guessed values that are very close to the true values. We show in Fig. 7 the reconstructed CKEPs and SPs (corresponding to user 2) using the transformation parameters with different accuracies. The numbers in the green and red boxes are the binary sequences of the parameters a and b, respectively, with the erroneous binary bits marked in blue. In the first major row, Eve reconstructs CKEPs and SP by using the parameters that are significantly different from the true values, but fails. In the second and third major rows, Eve guesses the vast majority of the binary bits in a and b, with the incorrect bits occurring in the high and low bits, respectively; the decimal numbers converted from the latter are closer to the true values, but any inconsistency in the parameter values leads to significant deviations in the reconstruction results. The fourth major row gives the reconstructed results using the correct parameters a and b as a reference for comparison. It can be seen that Eve

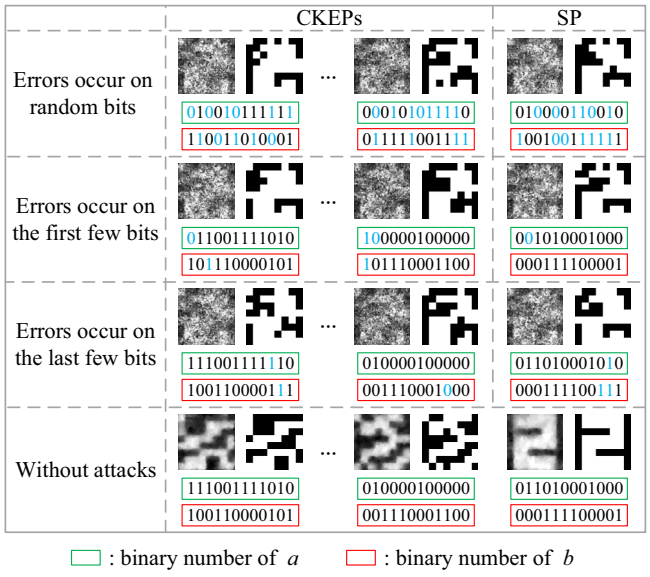

FIG. 7. Experimental results of CKEPs and SPs reconstructed by different Arnold-transformation parameters. We show the decoded and smoothed results in cases where errors occur on random bits, on the first few bits and on the last few bits, as well as with no attacks.

has to guess all the Arnold-transformation parameters correctly to reconstruct the subframes accurately. Meanwhile, Eve is unable to determine the closeness of the guessed parameters to the true values based on the reconstruction results, and has no way of knowing exactly which bits are guessed correctly or incorrectly, so he or she can only perform brute-force exhaustion. Consequently, this protocol excludes the possibility of adjusting the bit values according to the reconstruction results using the guessed parameters, to reduce the number of exhaustions.

In summary, since the modulation patterns and Arnold-transformation parameters are regarded as private keys, they are absolutely secure and cannot be stolen by Eve. This mechanism guarantees the security of the CKD.

B. Anti-interference analysis

Next, we will analyze the attack detection capabilities of this protocol. Since only the encrypted signals are transmitted over the public network, we only need to discuss the various attacks imposed on the bitstream. The physical attacks, such as channel cutting or hacking, will not be discussed here, as no protocol can withstand such kinds of attacks. Without loss of generality, we take the bitstream of the experimentally measured data of user 2, EP_2, as the target of the attack.

With respect to the EP_2 bitstream, Eve can perform a global attack on the entire sequence or a local attack on a small segment of the sequence. The attacked bitstream is rearranged into a 312×312 image, from which a SP can be decoded, as shown in Fig. 8. In the green dotted box of Fig. 8, we show the results without any attacks [see Figs. 8(a1)–8(a3)] as a reference comparison. We present the results under four different global attacks in the brown dotted box of Fig. 8, including completely disrupting the order of the entire sequence [disordering, see Figs. 8(b1)–8(b3)]; forging a new sequence to replace the original one [forging, see Figs. 8(c1)–8(c3)]; setting a threshold by which the converted decimal values greater than the threshold are set to 1 and the rest to 0 [binarization, see Figs. 8(d1)–8(d3)]; and expanding the length of the entire sequence to 101% of the original one by interpolation, and then cutting out a new bitstream sequence of the same length as the original one [resizing, see Figs. 8(e1)–8(e3)]. We can see from Figs. 8(b1)–8(e1) that the reshaped 312×312 images, after being subjected to the global attacks, have deviated significantly from the original EP_2 image. This is because every bit value in the bitstream sequence has been changed, and the image-reconstruction results will inevitably be severely affected, allowing legitimate users to easily detect the presence of an attack based on the smoothed SPs.

Compared to the global attack, the local attacks are harder to detect because they are only imposed on some small segments of the EP_2 bitstream sequence.

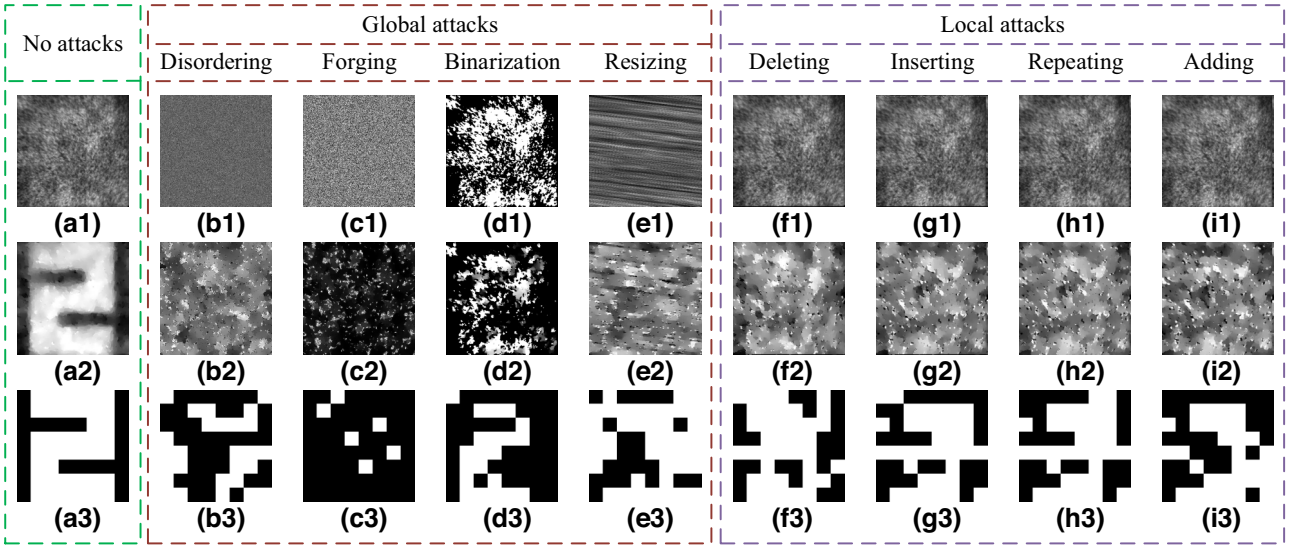

FIG. 8. Results under different attacks. (a1)–(a3) are EP_2, corresponding decoded image and smoothed image without attacks. (b1)–(b3), (c1)–(c3), (d1)–(d3), (e1)–(e3) are EP_2, decoded image and smoothed image under global attacks, including disordering, forging, binarization, resizing attacks. (f1)–(f3), (g1)–(g3), (h1)–(h3), (i1)–(i3) are EP_2, decoded image and smoothed image under local attacks, including deleting, inserting, repeating, adding attacks.

We show the results under the local attacks in the purple dotted box of Fig. 8. The local attacks employed here include deleting a fragment of the bit sequence [deleting, see Figs. 8(f1)–8(f3)]; inserting a sequence of consecutive random bits at a given position [inserting, see Figs. 8(g1)–8(g3)]; repeating a sequence of bits once and shifting the subsequent sequence of bits backwards [repeating, see Figs. 8(h1)–8(h3)]; and adding a single value at several random positions of the bitstream sequence and shifting the subsequent bits backwards accordingly [adding, see Figs. 8(i1)–8(i3)]. Since only some small segments of the EP_2 bitstream sequence are attacked, the reshaped 312×312 images [see Figs. 8(f1)–8(i1)] appear very close to the recovered SP without attacks [as shown in Fig. 8(a1)]. However, based on the smoothed SP results, users can still easily detect the presence of an attack. Once when the legitimate user detects the attack from Eve, he or she will immediately discard the distributed CKs in this round of communication, notify the server and the intermediary of the existence of the attack, and request to initiate a new round of the CKD.

It can be seen that this protocol can easily detect the existence of illegal attacks. The nonstealability of the private keys and the robustness of the authentication mechanism provide double insurance for the CKD, which greatly improves the security of the protocol.

C. Antinoise capability analysis

In the actual encrypted optical signal acquisition process, measurement noise, and transmission channel noise are unavoidable. The presence of such noise will affect the decoding quality of image frames, which in turn will affect the accuracy of authentication, CKs' and Arnold-transformation parameters' extraction. It is therefore necessary to analyze the performance of the protocol against noise. Here, we add white Gaussian noise with a standard deviation σ changing from 50 to 200 in intervals of 50 to

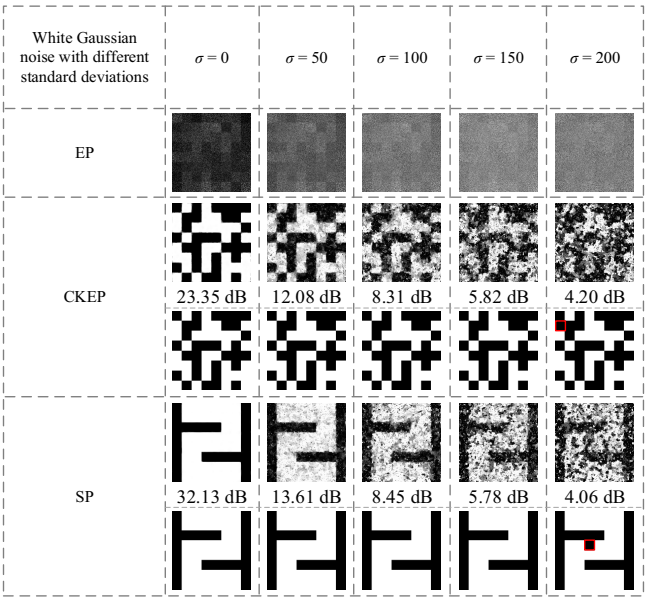

FIG. 9. Antinoise performance test by adding white Gaussian noise with a standard deviation ranging from 50 to 200 in intervals of 50 to simulated acquired EP_2. The case of $\sigma = 0$ is treated as a reference.

the simulated acquired EP_2, to evaluate the noise resistance of the protocol.

In Fig. 9, we take the first CKEP and the SP of user 2 as observing subjects. We can see from the results that both of them can be accurately recovered when $\sigma < 200$. This is particularly evident when $\sigma = 150$: the reconstructed CKEP and SP are still free of error points after smoothing, although the effective signal of the EP is submerged in noise. However, when $\sigma = 200$, the noise fluctuation is more significant, further degrading the reconstructed results and eventually leading to an error point in the smoothed CKEP and SP. Here, we use the peak signal-to-noise ratio (PSNR) [26] as a quantitative metric to evaluate the image quality. The decoding results demonstrate that our protocol is able to function even in extremely harsh and noisy environments, and has an ultrahigh noise immunity. This advantage makes the protocol highly applicable to practical optical CKD tasks.

V. CONCLUSION

In conclusion, we propose here a cascade SCI-based CKD protocol over a public network. In this protocol, multiple image frames are encrypted by using updated modulation patterns and superimposed into a single image frame to be recorded within a single exposure, which not only improves the encoding efficiency and CK generation rate, but also reduces the storage capacity of private keys and the computational burden of decoding. In the key-distribution process, we use a cascading mechanism, where parameters are extracted from the checksum data of the decoding results in the previous round of communication, and fed into the Arnold transformation to randomly scramble the modulation patterns previously used. This allows the modulation patterns employed in each CKD to be updated in real time, improving the nonstealability of private keys and making the temporary intervention of fake identity nowhere to hide. Both simulation and experimental results have demonstrated the feasibility of this protocol. Through self-authentication, legitimate users can immediately and effortlessly detect the presence of an illegal attack (this paper provides examples of four global attacks and four local attacks). We also provide an in-depth analysis and justification of the protocol's security and noise resistance. In a nutshell, the proposed protocol provides a new idea of cascading for optical CKD, and offers technical support and guarantee for it to move towards practical applications.

ACKNOWLEDGMENTS

This work was supported by the Beijing Natural Science Foundation (Grant No. 4222016) and the National Defense Technology Innovation Special Zone Project, Commission of Science, Technology and Industry for National Defense of the CPLA (Grant No. 23-TQ09-41-TS-01-011).

[1] B. Javidi, Securing information with optical technologies, Phys. Today **50**, 27 (1997).
[2] G. Unnikrishnan and K. Singh, Double random fractional Fourier domain encoding for optical security, Opt. Eng. **39**, 2853 (2000).
[3] O. Matoba, T. Nomura, E. Perez-Cabre, M. S. Millan, and B. Javidi, Optical techniques for information security, Proc. IEEE **97**, 1128 (2009).
[4] J. S. Sidhu, T. Brougham, D. McArthur, R. G. Pousa, and D. K. L. Oi, Finite key performance of satellite quantum key distribution under practical constraints, Commun. Phys. **6**, 210 (2023).
[5] R. Mandil, S. DiAdamo, B. Qi, and A. Shabani, Quantum key distribution in a packet-switched network, npj Quantum Inf. **9**, 1 (2023).
[6] S.-F. Shao, X.-Y. Cao, Y.-M. Xie, J. Gu, W.-B. Liu, Y. Fu, H.-L. Yin, and Z.-B. Chen, Phase-matching quantum key distribution without intensity modulation, Phys. Rev. Appl. **20**, 024046 (2023).
[7] S. Wang, Z.-Q. Yin, D.-Y. He, W. Chen, R.-Q. Wang, P. Ye, Y. Zhou, G.-J. Fan-Yuan, F.-X. Wang, W. Chen, Y.-G. Zhu, P. V. Morozov, A. V. Divochiy, Z. Zhou, G.-C. Guo, and Z.-F. Han, Twin-field quantum key distribution over 830-km fibre, Nat. Photonics **16**, 154 (2022).
[8] P. Zeng, H. Zhou, W. Wu, and X. Ma, Mode-pairing quantum key distribution, Nat. Commun. **13**, 3903 (2022).
[9] F.-Y. Lu, Z.-H. Wang, Z.-Q. Yin, S. Wang, R. Wang, G.-J. Fan-Yuan, X.-J. Huang, D.-Y. He, W. Chen, Z. Zhou, G.-C. Guo, and Z.-F. Han, Unbalanced-basis-misalignment-tolerant measurement-device-independent quantum key distribution, Optica **9**, 886 (2022).
[10] F.-Y. Lu, P. Ye, Z.-H. Wang, S. Wang, Z.-Q. Yin, R. Wang, X.-J. Huang, W. Chen, D.-Y. He, G.-J. Fan-Yuan, G.-C. Guo, and Z.-F. Han, Hacking measurement-device-independent quantum key distribution, Optica **10**, 520 (2023).
[11] F.-Y. Lu, Z.-H. Wang, V. Zapatero, J.-L. Chen, S. Wang, Z.-Q. Yin, M. Curty, D.-Y. He, R. Wang, W. Chen, G.-J. Fan-Yuan, G.-C. Guo, and Z.-F. Han, Experimental demonstration of fully passive quantum key distribution, Phys. Rev. Lett. **31**, 110802 (2023).
[12] R. L. Rivest, A. Shamir, and L. Adleman, A method for obtaining digital signatures and public-key cryptosystems, Commun. ACM **21**, 120 (1978).
[13] R. Chen, T. Shang, and J. Liu, IND-secure quantum symmetric encryption based on point obfuscation, Quantum Inf. Process. **18**, 161 (2019).
[14] F. Lalem, A. Laouid, M. Kara, M. Al-Khalidi, and E. Amna, A novel digital signature scheme for advanced asymmetric encryption techniques, Appl. Sci. **13**, 5172 (2023).
[15] J. H. Shapiro, Computational ghost imaging, Phys. Rev. A **78**, 061802(R) (2008).
[16] Y. Bromberg, O. Katz, and Y. Silberberg, Ghost imaging with a single detector, Phys. Rev. A **79**, 053840 (2009).
[17] W.-K. Yu, M.-F. Li, X.-R. Yao, X.-F. Liu, L.-A. Wu, and G.-J. Zhai, Adaptive compressive ghost imaging based on

wavelet trees and sparse representation, Opt. Express **22**, 7133 (2014).
[18] H.-D. Ren, L. Wang, and S.-M. Zhao, Efficient edge detection based on ghost imaging, OSA Contin. **1**, 64 (2019).
[19] S. Jiao, M. Sun, Y. Gao, T. Lei, Z. Xie, and X. Yuan, Motion estimation and quality enhancement for a single image in dynamic single-pixel imaging, Opt. Express **9**, 12841 (2019).
[20] H. Cui, J. Cao, Q. Hao, D. Zhou, H. Zhang, L. Lin, and Y. Zhang, Improving the quality of panoramic ghost imaging via rotation and scaling invariances, Opt. Laser Technol. **60**, 109102 (2023).
[21] L. Zhou, Y. Xiao, and W. Chen, High-resolution self-corrected single-pixel imaging through dynamic and complex scattering media, Opt. Express **31**, 23027 (2023).
[22] L.-K. Du, S. Sun, L. Jiang, C. Chang, H.-Z. Lin, and W.-T. Liu, Information segregating towards simultaneous tracking and imaging based on ghost imaging, Phys. Rev. Appl. **19**, 054014 (2023).
[23] T. B. Pittman, Y. H. Shih, D. V. Strekalov, and A. V. Sergienko, Optical imaging by means of two-photon quantum entanglement, Phys. Rev. A **52**, R3429(R) (1995).
[24] M.-F. Li, Y.-R. Zhang, K.-H. Luo, L.-A. Wu, and H. Fan, Time-correspondence differential ghost imaging, Phys. Rev. A **87**, 033813 (2013).
[25] Z. Liu, S. Tan, J. Wu, E. Li, X. Shen, and S. Han, Spectral camera based on ghost imaging via sparsity constraints, Sci. Rep. **6**, 25718 (2016).
[26] Y.-X. Li, W.-K. Yu, J. Leng, and S.-F. Wang, Pseudo-thermal imaging by using sequential-deviations for real-time image reconstruction, Opt. Express **27**, 35166 (2019).
[27] W. Gong, Sub-Nyquist ghost imaging by optimizing point spread function, Opt. Express **29**, 17591 (2021).
[28] L.-Y. Chen, C. Wang, X.-Y. Xiao, C. Ren, D.-J. Zhang, Z. Li, and D.-Z. Cao, Denoising in SVD-based ghost imaging, Opt. Express **30**, 6248 (2022).
[29] Y. Kang, L. Zhang, H. Ye, and D. Zhang, Hybrid encryption scheme based on temporal ghost imaging, Appl. Phys. B **27**, 124 (2021).
[30] J. Xiong, P. Zheng, Z. Gao, and H.-C. Liu, Algorithm-dependent computational ghost encryption and imaging, Phys. Rev. Appl. **18**, 034023 (2022).
[31] C. Xu, D. Li, K. Guo, Z. Yin, and Z. Guo, Computational ghost imaging with key-patterns for image encryption, Opt. Commun. **73**, 129190 (2023).
[32] S. Li, X.-R. Yao, W.-K. Yu, L.-A. Wu, and G.-J. Zhai, High-speed secure key distribution over an optical network based on computational correlation imaging, Opt. Lett. **38**, 2144 (2013).
[33] Q. Guan, H. Deng, W. Liang, X. Zhong, and M. Ma, Multi-images encryption and watermarking with small number of keys via computational ghost imaging, Opt. Laser Technol. **68**, 109957 (2023).
[34] S. Zhao, X. Yu, L. Wang, W. Li, and B. Zheng, Secure optical encryption based on ghost imaging with fractional Fourier transform, Opt. Commun. **74**, 126086 (2020).
[35] L. Zhang, Y. Wang, and D. Zhang, Research on multiple-image encryption mechanism based on Radon transform and ghost imaging, Opt. Commun. **04**, 127494 (2022).
[36] W.-K. Yu, N. Wei, Y.-X. Li, Y. Yang, and S.-F. Wang, Multi-party interactive cryptographic key distribution protocol over a public network based on computational ghost imaging, Opt. Lasers Eng. **55**, 107067 (2022).
[37] Y. Kang, S. Kanwal, B. Liu, and D. Zhang, Ghost key distribution under mutual authentication mechanism, Inf. Sci. **40**, 119025 (2023).
[38] S. Zhao, L. Wang, W. Liang, W. Cheng, and L. Gong, High performance optical encryption based on computational ghost imaging with QR code and compressive sensing technique, Opt. Commun. **53**, 90 (2015).
[39] W.-K. Yu, Y. Yang, Y.-X. Li, N. Wei, and S.-F. Wang, Multi-party cryptographic key distribution protocol over a public network based on a quick-response code, Sensors **22**, 3994 (2022).
[40] Z. Zhang, C. Deng, Y. Liu, X. Yuan, J. Suo, and Q. Dai, Ten-mega-pixel snapshot compressive imaging with a hybrid coded aperture, Photon. Res. **9**, 2277 (2021).
[41] W.-K. Yu, S.-F. Wang, and K.-Q. Shang, Joint authentication public network cryptographic key distribution protocol based on single exposure compressive ghost imaging, Chin. Phys. Lett. **41**, 024201 (2024).
[42] C. Li, W. Yin, H. Jiang, and Y. Zhang, An efficient augmented Lagrangian method with applications to total variation minimization, Comput. Optim. Appl. **56**, 507 (2013).
[43] G. Qu, X. Meng, Y. Yin, H. Wu, X. Yang, X. Peng, and W. He, Optical color image encryption based on Hadamard single-pixel imaging and Arnold transformation, Opt. Lasers Eng. **37**, 106392 (2021).
[44] Y. Zhou, Y. Sun, M. Yang, B. Zhou, J. Hou, T. Zeng, Z. Xiao, and L. Sui, Optical multiple-image authentication based on computational ghost imaging and hybrid non-convex second-order total variation, Opt. Express **31**, 20887 (2023).