



International Journal of Communication Networks and Distributed Systems

ISSN online: 1754-3924 - ISSN print: 1754-3916

<https://www.inderscience.com/ijcnds>

Addressing escalating threats: a secret image sharing scheme with adjustable threshold resilience against external adversaries and colluding participants

Krishnaraj Bhat, Devesh Jinwala, Yamuna Prasad, Mukesh A. Zaveri

DOI: [10.1504/IJCND5.2025.10063445](https://doi.org/10.1504/IJCND5.2025.10063445)

Article History:

Received:	24 August 2023
Last revised:	12 March 2024
Accepted:	16 March 2024
Published online:	02 December 2024

Addressing escalating threats: a secret image sharing scheme with adjustable threshold resilience against external adversaries and colluding participants

Krishnaraj Bhat* and Devesh Jinwala

Department of Computer Science and Engineering,
Sardar Vallabhbhai National Institute of Technology,
Surat, Gujarat, 395007, India
Email: krrishvaradha08@gmail.com
Email: dcj@svnit.ac.in
*Corresponding author

Yamuna Prasad

Department of Computer Science and Engineering,
Indian Institute of Technology Jammu,
Jammu & Kashmir, 181221, India
Email: yamuna.prasad@iitjammu.ac.in

Mukesh A. Zaveri

Department of Computer Science and Engineering,
Sardar Vallabhbhai National Institute of Technology,
Surat, Gujarat, 395007, India
Email: mazaveri@gmail.com

Abstract: We propose a novel threshold changeable secret image sharing (TCSIS) scheme based on univariate and asymmetric bivariate polynomials for securing greyscale and colour secret images in the semi-honest model. The proposed scheme supports increasing the threshold number of shares needed to reveal the secret image using the share update mechanism. Unlike the existing TCSIS schemes, the proposed scheme can prevent the revealing of a secret image before the end of its lifespan. This revelation is possible due to the increasing capabilities of an external adversary and the increasing number of colluding participants. Furthermore, in contrast to some existing TCSIS schemes, each share generated using the proposed scheme does not leak visual information about the secret image. We prove the properties of the proposed scheme using theoretical analysis. We also provide experimental results supporting the implications of theoretical analysis corresponding to the execution time and the randomness of shares.

Keywords: image security; secret image; TCSIS; threshold changeable secret image sharing; external adversary; colluding participants; information leakage; univariate polynomial; asymmetric bivariate polynomial.

Reference to this paper should be made as follows: Bhat, K., Jinwala, D., Prasad, Y. and Zaveri, M.A. (2025) 'Addressing escalating threats: a secret image sharing scheme with adjustable threshold resilience against external adversaries and colluding participants', *Int. J. Communication Networks and Distributed Systems*, Vol. 31, No. 1, pp.89–122.

Biographical notes: Krishnaraj Bhat received his BE in Information Science and Engineering from M. S. Ramaiah Institute of Technology, Bangalore, India, in 2012, the MTech in Information Systems Security Engineering from National Institute of Technology Jamshedpur, Jharkhand, India, in 2017. He is currently pursuing PhD in the Department of Computer Science and Engineering, Sardar Vallabhbhai National Institute of Technology, Surat, Gujarat, India. His current research interests include the areas of cryptography, and information security.

Devesh Jinwala received his BE in Electronics Engineering, ME in Electrical Engineering, and PhD in Computer Engineering from Sardar Vallabhbhai National Institute of Technology, Surat, Gujarat, India, in 2009. He is currently a Professor in the Department of Computer Science and Engineering, Sardar Vallabhbhai National Institute of Technology. His current research interests include the areas of cryptography, information and communications security, privacy preserving, security in resource constrained devices, software requirements specifications, load distribution and failure tolerance in distributed systems, algorithms and computational complexity.

Yamuna Prasad received his BTech in Computer Science and Engineering, in 2004, MTech in Computer Science and Engineering, in 2008, and PhD in Computer Science and Engineering from Indian Institute of Technology Delhi, India in 2016. He is currently an Assistant Professor in the Department of Computer Science and Engineering, Indian Institute of Technology Jammu, Jammu and Kashmir, India. His current research interests include the areas of machine learning, deep learning, big data, soft computing, cyber physical system security, natural language processing, computer vision, biometrics, and bioinformatics.

Mukesh A. Zaveri received his BE in Electronics Engineering from Sardar Vallabhbhai Regional College of Engineering and Technology, Surat, India, in 1990, ME in Electrical Engineering from Maharaja Sayajirao University, Baroda, India, in 1993, and the PhD in Electrical Engineering from Indian Institute of Technology Bombay, Mumbai, in 2005. He is currently a Professor in the Department of Computer Science and Engineering, Sardar Vallabhbhai National Institute of Technology. His current research interests include the areas of signal and image processing, multimedia, computer networks, sensor networks, wireless communications, internet of things, and machine learning.

1 Introduction

Military, intelligence, police, business, and medical organisations generate and store greyscale and colour secret images containing sensitive information (Wang et al., 2023; Ye and Chen, 2023; Wu et al., 2022)). The Secret Image Sharing (SIS) technique provides

security against the disclosure, corruption, and deletion of these secret images stored at a location (Saha et al., 2023; Sardar and Adhikari, 2020). In (k, n) SIS, where $2 \leq k \leq n$, the dealer divides a secret image I into n shares and distributes among n participants (Bhat et al., 2023; Yu et al., 2020; Wu et al., 2020). Then, the dealer deletes I and corresponding n shares to avoid a single point of failure. To reconstruct I , a combiner (dealer or participant) requires at least k (threshold) shares (Li et al., 2019; Gong et al., 2019). Furthermore, a combiner can reconstruct I even if $n - k$ shares are corrupted or deleted.

The (k, n) SIS schemes (Prashanti and Bhat, 2024; Zhang et al., 2024; Luo et al., 2024; Yang et al., 2023a; Wu and Yao, 2023; Mashhadi and Saeedi, 2023; Xie et al., 2023; Hilmi et al., 2023; Wu et al., 2023; Yan et al., 2023; Xiong et al., 2023a; Pande et al., 2023; Xiong et al., 2023b; Li et al., 2023; Hua et al., 2022; Yan et al., 2021; Meng et al., 2021; Luo et al., 2023; Yang et al., 2023b; Ren et al., 2022; Yang et al., 2022; Paul et al., 2022; Hu et al., 2021; Patil and B.R., 2021) have the following two assumptions considered within the lifespan of a secret image:

- An external adversary cannot capture more than $k - 1$ shares.
- The number of colluding participants cannot increase more than $k - 1$.

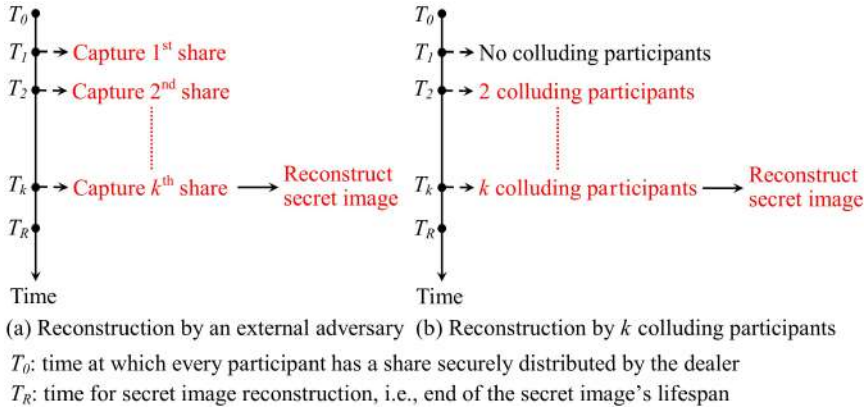
However, over time, the capabilities of an external adversary and/or the number of colluding participants can increase. With these behaviours, an external adversary can capture k shares, and/or the number of colluding participants can increase to k , before the end of the secret image's lifespan. Thus, the reconstruction of the secret image before the end of its lifespan is possible.

For instance, consider that an agent A_g from an intelligence agency works undercover in a criminal organisation O_C . To protect the identity of A_g , a high-rank officer (honest dealer) in that intelligence agency securely distributes n shares of a photo (greyscale or colour image) of A_g to n low-rank officers (semi-honest participants). As shown in Figure 1, consider that at time T_0 , each participant has a share securely distributed by the dealer. Furthermore, consider that at time T_R , A_g will accomplish the mission, and a combiner can reveal A_g 's identity. However, by time $T_k < T_R$ as shown in Figure 1, O_C can capture k shares, or the number of colluding participants can increase to k . This behaviour discloses A_g 's identity before time T_R . To overcome this problem, the solution is to increase the threshold from k to t assuming that neither an external adversary captures t shares, nor the number of colluding participants increases to t by time T_R (Sun et al., 2022; Wu et al., 2021; Liu et al., 2019; Islam et al., 2016). Moreover, reconstruction of the secret image with less than t shares should not be possible.

In the literature, threshold changeable secret sharing (TCSS) schemes (Harn et al., 2023; Ding et al., 2023, 2022; Harn et al., 2022; Meng et al., 2020; Jia et al., 2019; Yuan et al., 2016; Harn and Hsu, 2015; Zhang et al., 2012; Steinfeld et al., 2007, 2006) for sharing a cryptographic key in the semi-honest model exist with support for increase in threshold (Section 2). However, directly applying these TCSS schemes on greyscale and colour secret images leads to the following two demerits (Liu et al., 2019; Yan et al., 2020):

- 1 addition of noise in the reconstructed secret image due to the inability to handle pixel value range-related issues
- 2 the size of each generated share is greater than that of the secret image.

Figure 1 Reconstruction of the secret image before the end of its lifespan (see online version for colours)



The first demerit causes degradation in the visual quality of the reconstructed secret image. With the second demerit, each share consumes more storage memory.

The threshold changeable secret image sharing (TCSIS) schemes (Sun et al., 2022; Wu et al., 2021; Liu et al., 2019; Islam et al., 2016) designed for sharing greyscale and colour secret images in the semi-honest model overcome the first demerit of TCSS schemes. We classify these TCSIS schemes into two categories, viz. Category-1 and Category-2, based on the mechanism they use to support the increase in the threshold.

Category-1 TCSIS schemes (Islam et al., 2016; Wu et al., 2021) support the increase in threshold by maintaining the confidentiality of respective key(s) needed to compute fake identifiers corresponding to different supported thresholds. However, using the brute-force attack corresponding to the possible number of fake identifiers, an external adversary and colluding participants can reveal the secret image by collecting the minimum supported threshold number of shares.

Category-2 TCSIS schemes (Liu et al., 2019; Sun et al., 2022) eliminate this brute-force attack possible in Category-1 TCSIS schemes by not depending on the confidentiality of identifiers of shares to increase the threshold. Instead, in Category-2 TCSIS schemes, the identifier of each share is public. They support the increase in threshold by updating each share corresponding to the new threshold. Furthermore, the size of each share generated using Category-2 TCSIS schemes is lesser than that of the secret image. This feature overcomes the second demerit of using TCSS schemes on greyscale and colour images. However, an external adversary and colluding participants can reconstruct the secret image using collected initial and updated shares whose total count is less than the final threshold. This reconstruction is possible by subtraction operations and solving linearly independent equations. Furthermore, there is visual information leakage of the secret image via each share generated using Category-2 TCSIS schemes. With this demerit, a participant can reveal the secret image using the share distributed by the dealer.

In summary, the two demerits common to Category-1 and Category-2 TCSIS schemes include

- 1 insecurity against the increasing capabilities of an external adversary
- 2 insecurity against the increasing number of colluding participants.

Furthermore, one demerit specific to Category-2 TCSIS schemes is visual information leakage of the secret image via each share. We describe how these demerits exist in the Category-1 and Category-2 TCSIS schemes in Section 2. We aim to propose a TCSIS scheme that overcomes these three demerits in the Category-1 and Category-2 TCSIS schemes.

We know that Category-2 TCSIS schemes eliminate the brute-force attack possible in Category-1 TCSIS schemes corresponding to identifiers of shares. In the existing TCSIS schemes, Sun et al. (2022) scheme belonging to Category-2 is the state-of-the-art. We provide the reasons for Sun et al. (2022) scheme being state-of-the-art in Section 2.2. To achieve our aim, we propose a novel Category-2 TCSIS scheme based on univariate and asymmetric bivariate polynomials by removing the vulnerabilities in Sun et al. (2022) scheme (Section 4). The proposed scheme supports $h - k + 1$ consecutive thresholds $k, k + 1, \dots, h$, where k is the initial threshold, h is the maximum possible threshold, and $2 \leq k < h \leq n$. Based on the need, the threshold can be increased from k to t via updating each share, where $k < t \leq h$. We prove each initial and updated share is random (Section 5). We prove the number of colluding participants less than the current threshold cannot reconstruct the secret image. We prove an external adversary cannot reconstruct the secret image using captured initial and updated shares whose total count is less than the current threshold. We theoretically show that the asymptotic time complexity of updating a share corresponding to threshold $t \in [k + 1, h - 1]$ and the size of each initial share are lesser than those corresponding to Sun et al. (2022) scheme. We also show examples of visual information leakage of the secret image via individual share generated using the existing Category-2 TCSIS schemes (Liu et al., 2019; Sun et al., 2022) (Section 6). Furthermore, we provide the experimental results validating the implications of theoretical analysis.

In summary, the main contributions of this paper are as follows:

- We propose a novel TCSIS scheme based on univariate and asymmetric bivariate polynomials for sharing greyscale and colour secret images with support for $h - k + 1$ consecutive thresholds $k, k + 1, \dots, h$.
- The proposed scheme can resist the increasing capabilities of an external adversary and the increasing number of colluding participants by supporting the increase in threshold using the share update mechanism. Furthermore, each initial and updated share reveals no information about the secret image.
- The size of an initial share and the execution time for updating a share corresponding to the new threshold $t \in [k + 1, h - 1]$ using the proposed scheme are lesser than those using the state-of-the-art Sun et al. (2022) scheme.

2 Related work

In the literature, TCSS schemes (Harn et al., 2023; Ding et al., 2023, 2022; Harn et al., 2022; Meng et al., 2020; Jia et al., 2019; Yuan et al., 2016; Harn and Hsu, 2015; Zhang et al., 2012; Steinfeld et al., 2007, 2006) for sharing a cryptographic key in the semi-honest model exist to support the increase in threshold. However, directly applying these TCSS schemes on greyscale and colour secret images results in the following two demerits (Liu et al., 2019; Yan et al., 2020):

- Addition of noise in the reconstructed secret image because of the inability to address pixel value range-related issues.

- The size of each generated share is more than that of the secret image.

The first demerit causes degradation in the visual quality of the reconstructed secret image. The second demerit causes more storage memory consumption by each share. Hence, to overcome these two demerits while sharing greyscale and colour secret images, researchers propose TCSIS schemes (Islam et al., 2016; Wu et al., 2021; Liu et al., 2019; Sun et al., 2022).

We classify these TCSIS schemes (Islam et al., 2016; Wu et al., 2021; Liu et al., 2019; Sun et al., 2022) into two categories, viz. Category-1 and Category-2, based on the mechanism they use to support the increase in the threshold. Category-1 TCSIS schemes (Islam et al., 2016; Wu et al., 2021) support the increase in threshold via maintaining the confidentiality of respective key(s) needed to compute fake identifiers corresponding to different supported thresholds. Category-2 TCSIS schemes (Liu et al., 2019; Sun et al., 2022) support the increase in threshold via updating each share corresponding to the new threshold. Note that Category-1 TCSIS schemes only overcome the first demerit of TCSS schemes. However, Category-2 TCSIS schemes overcome the two demerits of TCSS schemes.

The two demerits common to Category-1 and Category-2 TCSIS schemes include:

- insecurity against the increasing capabilities of an external adversary
- insecurity against the increasing number of colluding participants.

These two demerits lead to the disclosure of the secret image before the end of its lifespan by an external adversary and by colluding participants respectively. We now category-wise briefly describe how these two demerits exist in each TCSIS scheme.

2.1 Category-1 TCSIS schemes

Category-1 TCSIS schemes (Islam et al., 2016; Wu et al., 2021) support z potential thresholds $k_1, k_2, \dots, k_z$, where $2 \leq k_1 < k_2 < \dots < k_z \leq n$ (in Islam et al. (2016) scheme, for $1 \leq j < z, k_1 > k_{j+1} - k_j$). If an external adversary captures k_j shares, or the number of colluding participants increases to k_j , then the dealer chooses k_{j+1} as the threshold for the reconstruction of the secret image. The assumption is neither an external adversary captures k_z shares, nor the number of colluding participants increases to k_z , before the end of the lifespan of the respective secret image. In these schemes (Islam et al., 2016; Wu et al., 2021), during the shares generation phase, the dealer uses z random keys $K_1, K_2, \dots, K_z$, and the real identifiers of n shares $r_1, r_2, \dots, r_n$ to generate zn fake identifiers $f(K_j, r_i)$, where $1 \leq j \leq z$ and $1 \leq i \leq n$. Here, f is a two-variable one-way function. The dealer only knows the z random keys K_1 to K_z . Only the dealer and i^{th} ($1 \leq i \leq n$) participant knows r_i . During the secret image reconstruction phase, if the dealer chooses k_j as the threshold, then dealer broadcasts keys $K_j, K_{j+1}, \dots, K_z$ and threshold k_j in case of Islam et al. (2016) scheme. In the case of Wu et al. (2021) scheme, the dealer broadcasts key K_j and threshold k_j . Using the respective key(s) and real identifiers corresponding to k_j shares, a combiner computes respective fake identifiers and reconstructs the secret image using k_j shares. Islam et al. (2016) and Wu et al. (2021) claim that without knowing the respective key(s) for computing fake identifiers, a combiner cannot reconstruct the secret image using any number of shares.

Note that in these schemes (Islam et al., 2016; Wu et al., 2021), all operations are performed in $GF(p)$, where $p \in [0, 255]$ is a prime. Furthermore, each generated share is

meaningful. To generate each meaningful share with Peak Signal to Noise Ratio greater than 35 decibel, the preferred value of p is 7. Hence, the value of each fake identifier lies within the interval $[1, p - 1]$ (zero is not used as a real or fake identifier). In Islam et al. (2016) scheme, with k_1 shares, the total number of possible fake identifiers is $(p^{-1}P_{k_1})^z$ (permutations). Similarly, in Wu et al. (2021) scheme, with k_1 shares, the total number of possible fake identifiers is $p^{-1}P_{k_1}$. Because k_1 and z are less than p , and the preferred value of p is 7, individually attempting $(p^{-1}P_{k_1})^z$ and $p^{-1}P_{k_1}$ numbers of fake identifiers is computationally feasible. Hence, an external adversary using k_1 captured shares, or k_1 colluding participants using their shares, can reconstruct the secret image using the brute-force attack even though the chosen threshold for reconstruction is greater than k_1 .

2.2 Category-2 TCSIS schemes

Category-2 TCSIS schemes (Liu et al., 2019; Sun et al., 2022) eliminate the brute-force attack corresponding to identifiers of shares possible in the case of Category-1 TCSIS schemes. Category-2 TCSIS schemes support the increase in threshold via updating each share, and the identifier of each share is public. In Category-2 TCSIS schemes, the dealer generates initial shares corresponding to threshold k , where $2 \leq k < n$. Before an external adversary captures k shares, or the number of colluding participants increases to k , the dealer broadcasts a new threshold t to all participants, where $k < t \leq n$. Then, each participant updates the initial share corresponding to t . The assumption is neither an external adversary captures t shares, nor the number of colluding participants increases to t , before the end of the lifespan of the corresponding secret image. If no need to increase the threshold, then the dealer broadcasts k as the final threshold for the reconstruction of the secret image. In Category-2 schemes, there is visual information leakage of the secret image via individual share. The reason is not hiding the correlation among adjacent pixels in the secret image. With this demerit, a participant can reveal the secret image via the initial share. We now separately provide a brief on how an external adversary and colluding participants can reveal the secret image in Liu et al. (2019) and Sun et al. (2022) schemes.

Liu et al. (2019) scheme supports three thresholds k_1, k, k_2 , where k is the initial threshold and $2 \leq k_1 < k < k_2 \leq n$. Furthermore, this scheme has all operations in $GF(251)$. The dealer generates initial shares using the following steps. First, divide the secret image into non-overlapping blocks with k_1 greyscale values in each block. For each block B_m , truncate every greyscale value greater than 250 to 250. Then, generate a polynomial $v_{k_1}(x)$ of degree $k_1 - 1$ with k_1 greyscale values in B_m as k_1 coefficients in $v_{k_1}(x)$. For each B_m , generate i^{th} ($1 \leq i \leq n$) share value $s_{m,i}$ as $s_{m,i} = v_{k_1}(i) + v_k(i)$, where $v_k(x) = c_{k_1}x^{k_1} + c_{k_1+1}x^{k_1+1} + \dots + c_{k-1}x^{k-1}$ is a $k - 1$ degree polynomial with $k - k_1$ random coefficients in the interval $[0, 250]$. Use the same $v_k(x)$ while generating initial share values corresponding to every block in the secret image. Here, value i is the identifier of i^{th} share. To increase the threshold to k_2 , the dealer broadcasts k_2 and $v_{k_2}(i) \forall i \in [1, n]$ to all participants. Here, $v_{k_2}(x) = c_kx^k + c_{k+1}x^{k+1} + \dots + c_{k_2-1}x^{k_2-1}$ is a $k_2 - 1$ degree polynomial with $k_2 - k$ random coefficients in the interval $[0, 250]$. Then, each i^{th} ($1 \leq i \leq n$) participant updates every share value $s_{m,i}$ in the initial share to $s_{m,i}^{k_2}$ as $s_{m,i}^{k_2} = s_{m,i} + v_{k_2}(i)$.

Consider that an external adversary E_A captures $k - 1$ initial shares $S_1, S_2, \dots, S_{k-1}$ corresponding to threshold k . Furthermore, E_A captures an updated share $S_k^{k_2}$ corresponding to threshold k_2 . Now, E_A can convert $S_k^{k_2}$ back to the corresponding initial share S_k using operation $s_{m,k} = s_{m,k}^{k_2} - v_{k_2}(k)$ on every greyscale value $s_{m,k}^{k_2}$ in

$S_k^{k_2}$. We know that every greyscale value in an initial share S_i is an evaluated value of the corresponding $k - 1$ degree univariate polynomial. In this univariate polynomial, k_1 coefficients are the corresponding k_1 secret greyscale values. Hence, using k shares $S_1, S_2, \dots, S_k, E_A$ can reconstruct the secret image using Lagrange interpolation on every k corresponding share values. Similarly, each i^{th} participant in k colluding participants can convert updated share $S_i^{k_2}$ to initial share S_i , and reconstruct the secret image using k regenerated initial shares. Hence, even after increasing the threshold from k to k_2 , an external adversary and k colluding participants can reconstruct the secret image using k shares.

Sun et al. (2022) scheme overcomes the demerit of support for only three possible thresholds in Liu et al. (2019) scheme. Unlike in Category-1 and Liu et al. (2019) schemes, the dealer need not broadcast extra values other than the chosen threshold for the reconstruction of the secret image. Sun et al. (2022) scheme supports $h - k + 1$ consecutive thresholds $k, k + 1, \dots, h$, where k is the initial threshold, h is the maximum possible threshold, and $2 \leq k < h \leq n$. Hence, Sun et al. (2022) scheme is state-of-the-art. Furthermore, this scheme supports operations in both $GF(251)$ and $GF(2^8)$. However, operations in $GF(251)$ lead to the lossy reconstruction of the secret image as in the Liu et al. (2019) scheme. The reason is the truncation of greyscale secret values greater than 250 to 250. In Sun et al. (2022) scheme, the dealer generates initial shares using the following steps. First, divide the secret image into non-overlapping blocks with kh greyscale values in each block. For each block B_m , generate an asymmetric bivariate polynomial $Q_m(x, y)$ as $Q_m(x, y) = \sum_{i=0}^{k-1} \sum_{j=0}^{h-1} c_{i,j} x^i y^j$. Here, $c_{i,j} \forall i \in [0, k - 1], j \in [0, h - 1]$ are kh greyscale values in the respective B_m . Then, generate i^{th} ($1 \leq i \leq n$) initial share block $s_{m,i}$ corresponding to B_m as $s_{m,i} = u_{m,i}(y) \parallel v_{m,i}(x)$. Here, $u_{m,i}(y) = Q_m(i, y)$, $v_{m,i}(x) = Q_m(x, i)$, value i is the identifier of i^{th} share, and $\parallel$ denotes concatenation. During the increase in threshold to t , where $k < t < h$, each i^{th} ($1 \leq i \leq n$) participant updates every share block $s_{m,i}$ in the initial share S_i as $s_{m,i}^t = v_{m,i}(x) \parallel u_{m,i}(e_1) \parallel u_{m,i}(e_2) \parallel \dots \parallel u_{m,i}(e_{h-t})$. Here, $s_{m,i}^t$ is the corresponding block in the updated share S_i^t . Furthermore, $e_1, e_2, \dots, e_{h-t}$ are the $h - t$ distinct integers other than the identifiers of n shares. Similarly, for $t = h$, $s_{m,i}^h = v_{m,i}(x)$, and for $t = k$, $s_{m,i}^k = u_{m,i}(y)$.

Note that we can form $(k - 1)(h + k) = kh + k^2 - (h + k)$ number of linearly independent equations using $k - 1$ initial share blocks corresponding to kh greyscale values per secret block (Harn et al., 2017). However, for $k \geq (h/k) + 1$, $kh + k^2 - (h + k) \geq kh$. Hence, for $k \geq (h/k) + 1$, an external adversary using $k - 1$ captured initial shares can reconstruct the secret image by solving linearly independent equations. Similarly, $k - 1$ colluding participants before the increase in threshold can do the same. For $k < t < h$ and $t - k < k$, we can form $(k - 1)(h + k) + (t - k)k = kh + kt - (h + k)$ number of linearly independent equations using $k - 1$ initial share blocks and $t - k$ updated share blocks corresponding to threshold t . However, for $t \geq (h/k) + 1$, $kh + kt - (h + k) \geq kh$. Furthermore, for $k < t < h$ and $t - k \geq k$, we can form $(k - 1)(h + k) + (t - k)k + (h - t)k = kh + kh - (h + k)$ number of linearly independent equations using $k - 1$ initial share blocks and $t - k$ updated share blocks corresponding to threshold t . However, for $k \geq 2$, $kh + kh - (h + k) > kh$. Hence, for $t \geq (h/k) + 1$ and $t - k < k$, and for $k \geq 2$ and $t - k \geq k$, an external adversary using captured $k - 1$ initial shares and $t - k$ updated shares corresponding to threshold t (total $t - 1$ shares) can reconstruct the secret image by solving linearly independent equations. Similarly, we can form $(k - 1)(h + k) + (h - k)k = kh + kh - (h + k)$ number of linearly independent equations using $k - 1$ initial

share blocks and $h - k$ updated share blocks corresponding to threshold h . As mentioned above, for $k \geq 2$, $kh + kh - (h + k) > kh$. Hence, an external adversary using captured $k - 1$ initial shares and $h - k$ updated shares corresponding to threshold h (total $h - 1$ shares) can reconstruct the secret image by solving linearly independent equations.

In Table 1, we provide the three demerits existing in the Category-1 and Category-2 TCSIS schemes briefed so far. In these three demerits, two demerits are common, and one demerit is specific to Category-2 TCSIS schemes. Hence, we aim to propose a TCSIS scheme that overcomes these three demerits. We know that Category-2 TCSIS schemes already eliminate the brute-force attack possible in Category-1 TCSIS schemes corresponding to identifiers of shares. Hence, we propose a Category-2 TCSIS scheme.

Table 1 The three demerits and corresponding effects in the existing TCSIS schemes

Category	Demerit	Effect
Category-1 and Category-2	(1) Insecurity against the increasing capabilities of an external adversary as in Islam et al. (2016), Wu et al. (2021), Liu et al. (2019) and Sun et al. (2022) schemes.	An external adversary can reconstruct the secret image before the end of secret image's lifespan.
	(2) Insecurity against the increasing number of colluding participants as in Islam et al. (2016), Wu et al. (2021), Liu et al. (2019) and Sun et al. (2022) schemes.	The colluding participants can reconstruct the secret image before the end of secret image's lifespan.
Category-2	(3) Visual information leakage of the secret image via each share as in Liu et al. (2019) and Sun et al. (2022) schemes.	A participant can reveal the secret image with the share distributed by the dealer.

3 Preliminaries

In this section, we provide a brief on univariate and asymmetric bivariate polynomials, which are the foundations for the proposed scheme.

We can represent an univariate polynomial $q(x)$ of degree $t - 1$ as shown in equation (1). In equation (1), $c_i, \forall i \in [0, t - 1]$ are constants. Evaluating $q(x)$ at $x = d$, where d is a constant, results in a constant. To reconstruct $q(x)$ using Lagrange interpolation, we require at least t evaluated values $q(d_1), q(d_2), \dots, q(d_t)$ corresponding to t constants $d_1, d_2, \dots, d_t$ (Bhat et al., 2021).

$$q(x) = c_0 + c_1x + \dots + c_{t-1}x^{t-1} \quad (1)$$

We can represent an asymmetric bivariate polynomial $Q(x, y)$ with degree t_x in x and t_y in y as shown in equation (2). In equation (2), $c_{i,j}, \forall i \in [0, t_x - 1]$ and $j \in [0, t_y - 1]$, are constants. Evaluating $Q(x, y)$ at $x = d$, where d is a constant, results in an univariate polynomial $u_d(y) = Q(d, y) = u'_0(d) + u'_1(d)y + \dots + u'_{t_y-1}(d)y^{t_y-1}$ of

degree $t_y - 1$. Here, $u'_i(x) = c_{0,i} + c_{1,i}x + \dots + c_{t_x-1,i}x^{t_x-1}$ is an univariate polynomial of degree $t_x - 1$. Similarly, evaluating $Q(x, y)$ at $y = d$ results in an univariate polynomial $v_d(x) = Q(x, d) = v'_0(d) + v'_1(d)x + \dots + v'_{t_x-1}(d)x^{t_x-1}$ of degree $t_x - 1$. Here, $v'_i(y) = c_{i,0} + c_{i,1}y + \dots + c_{i,t_y-1}y^{t_y-1}$ is an univariate polynomial of degree $t_y - 1$. To reconstruct $Q(x, y)$ using Lagrange interpolation, we require at least t_x univariate polynomials $u_{d_1}(y), u_{d_2}(y), \dots, u_{d_{t_x}}(y)$ corresponding to t_x constants $d_1, d_2, \dots, d_{t_x}$ as shown in equation (3), or we require at least t_y univariate polynomials $v_{d_1}(x), v_{d_2}(x), \dots, v_{d_{t_y}}(x)$ corresponding to t_y constants $d_1, d_2, \dots, d_{t_y}$ as shown in equation (4). Furthermore, $u_{d_i}(d_j) = v_{d_j}(d_i)$ and $v_{d_i}(d_j) = u_{d_j}(d_i)$ (Harn et al., 2017).

$$Q(x, y) = \begin{cases} c_{0,0} + c_{0,1}y + \dots + c_{0,t_y-1}y^{t_y-1} + \\ c_{1,0}x + c_{1,1}xy + \dots + c_{1,t_y-1}xy^{t_y-1} + \\ \dots \\ c_{t_x-1,0}x^{t_x-1} + c_{t_x-1,1}x^{t_x-1}y + \dots + c_{t_x-1,t_y-1}x^{t_x-1}y^{t_y-1} \end{cases} \quad (2)$$

$$Q(x, y) = \sum_{l=1}^{t_x} u_{d_l}(y) \left(\prod_{g=1, g \neq l}^{t_x} \frac{x - d_g}{d_l - d_g} \right) \quad (3)$$

$$Q(x, y) = \sum_{l=1}^{t_y} v_{d_l}(x) \left(\prod_{g=1, g \neq l}^{t_y} \frac{y - d_g}{d_l - d_g} \right) \quad (4)$$

4 Proposed scheme

We propose a TCSIS scheme by modifying operations in the state-of-the-art (Sun et al., 2022) scheme. Similar to Sun et al. (2022) scheme, the proposed scheme also supports $h - k + 1$ consecutive thresholds $k, k + 1, \dots, h$ for sharing greyscale and colour secret images. Here, k is the initial threshold, h is the maximum possible threshold, $2 \leq k < h \leq n$ and $n + h - k \leq 255$. The proposed scheme has three phases: shares generation, share update, and secret image reconstruction. In the shares generation phase, the dealer generates n initial shares of secret image I corresponding to threshold k and securely distributes them to n participants. Then, the dealer deletes I and corresponding n initial shares. Before an external adversary captures k shares, or the number of colluding participants increases to k , the dealer broadcasts a new threshold t to all participants, where $k < t \leq h$. Then, every participant executes the share update phase by generating the updated share from the initial share and then deleting the initial share. The assumption is that an external adversary can capture a maximum of $t - 1$ shares, or the number of colluding participants can increase to a maximum value $t - 1$, before the end of the lifespan of I . If no need to increase the threshold, the dealer broadcasts nothing, and k is the final threshold. Else, $t \in [k + 1, h]$ is the final threshold. In the secret image reconstruction phase, i.e., at the end of the lifespan of I , a combiner (dealer or participant) securely collects t ($k \leq t \leq h$) shares and reconstructs the secret image.

In the shares generation phase, an asymmetric bivariate polynomial $Q_m(x, y)$ is formed from each secret block B_m in I , where B_m contains $k(h - 1)$ greyscale values. Then, n initial share blocks are generated from $Q_m(x, y)$ corresponding to threshold k as shown in

Algorithm 1. Here, each initial share block is a combination of a univariate polynomial in x and evaluated values of a univariate polynomial in y . In the share update phase, each block in an initial share is updated corresponding to a new threshold t ($k < t \leq h$) using copy operations as shown in Algorithm 2. In the secret image reconstruction phase, $Q_m(x, y)$ corresponding to each B_m is reconstructed using Lagrange interpolation with univariate polynomials obtained from the corresponding t ($k \leq t \leq h$) share blocks as shown in Algorithms 3 and 4. We know that each greyscale value in a greyscale or colour secret image lies in the interval $[0, 255]$. Hence, to reconstruct the secret image, the same as the original corresponding to every greyscale value, all the operations are performed in $GF(2^8)$.

The proposed scheme overcomes the three existing demerits in the related schemes by eliminating the following two vulnerabilities in Sun et al. (2022) scheme:

- While generating initial share blocks per secret block B_m corresponding to threshold k , two univariate polynomials $Q_m(i, y)$ and $Q_m(x, i)$ corresponding to asymmetric bivariate polynomial $Q_m(x, y)$ are stored in i^{th} ($1 \leq i \leq n$) share block (from Section 2.2). However, only $Q_m(i, y)$ from each share block is sufficient to reconstruct kh greyscale values in B_m with threshold k . Additional $Q_m(x, i)$ in each share block is extra information about B_m , which is being exploited by an external adversary and colluding participants.
- The correlation among adjacent pixels in the secret image is not hidden during the generation of initial shares. This behaviour leads to visual information leakage of the secret image via individual share.

We first provide the security model for the proposed scheme. Second, we describe the three phases in the proposed scheme by mentioning the respective changes made to operations in Sun et al. (2022) scheme. Then, we illustrate the proposed scheme. In Table 2, we provide the notations along with their description which we use throughout this paper.

Table 2 Notations and their description

Notation	Description
I	Secret image (greyscale or colour)
G	Number of greyscale values in I
k	Initial threshold
h	Maximum possible threshold
n	Number of shares
S_i	i^{th} initial share of I
S_i^j	i^{th} updated share of I corresponding to threshold j
B_m	A secret block in I
$Q_m(x, y)$	Asymmetric bivariate polynomial corresponding to B_m
$s_{m,i}$	i^{th} initial share block of B_m
$s_{m,i}^j$	i^{th} updated share block of B_m corresponding to threshold j
$\parallel$	Concatenation

4.1 Security model

We consider the following security model for the proposed scheme. The dealer is honest. A participant can be semi-honest. The transmission of a share between the dealer (or a

combiner) and a participant occurs via a secure channel. It is sufficient to increase the threshold once. Each participant deletes the initial share after generating the corresponding updated share.

4.2 Shares generation phase

In the shares generation phase, the dealer generates initial shares with the following steps. First, divide secret image I into b blocks with each block B_m containing $k(h-1)$ greyscale values, where $m \in \{1, 2, \dots, b\}$. Then, for every B_m , generate n initial share blocks $s_{m,1}, s_{m,2}, \dots, s_{m,n}$ using Algorithm 1. For every B_m with $k(h-1)$ greyscale values, every corresponding $s_{m,r}$ ($1 \leq r \leq n$) contains h greyscale values. For each $r \in \{1, 2, \dots, n\}$, generate r^{th} initial share S_r as $S_r = s_{1,r} \parallel s_{2,r} \parallel \dots \parallel s_{b,r}$.

Algorithm 1: Initial share blocks generation.

Input: $k; h; n; B_m$ of size $k(h-1)$; publicly known n shares' distinct identifying indices $d_1, d_2, \dots, d_n \in [1, 255]$; $h-k$ publicly known distinct integers $e_1, e_2, \dots, e_{h-k} \in [1, 255]$, other than $d_1, d_2, \dots, d_n$.
Output: n initial share blocks $s_{m,1}, s_{m,2}, \dots, s_{m,n}$ of B_m .

```

1 for  $i = 1$  to  $k$  do /* lines 1-3: generate  $k(h-1)$  coefficients
   from  $B_m$  */
2   for  $j = 1$  to  $h-1$  do
3      $c_{i-1,j-1} = B_m[i][j]$ 
4 for  $i = 1$  to  $k$  do /* lines 4-5: generate  $k$  random
   coefficients in the interval  $[1, 255]$  */
5    $c_{i-1,h-1} = \text{Rand}(1, 255)$ 
6 Construct the asymmetric bivariate polynomial  $Q_m(x, y)$  as
   
$$Q_m(x, y) = \begin{cases} c_{0,0} + c_{0,1}y + \dots + c_{0,h-1}y^{h-1} + \\ c_{1,0}x + c_{1,1}xy + \dots + c_{1,h-1}xy^{h-1} + \\ \dots \\ c_{k-1,0}x^{k-1} + c_{k-1,1}x^{k-1}y + \dots + c_{k-1,h-1}x^{k-1}y^{h-1} \end{cases} \quad (5)$$

7 for  $i = 1$  to  $n$  do /* lines 7-11: generate  $n$  initial share
   blocks */
8    $s_{m,i} = v_{m,d_i}(x)$  /*  $v_{m,d_i}(x) = Q_m(x, d_i)$  */
9    $u_{m,d_i}(y) = Q_m(d_i, y)$ 
10  for  $j = 1$  to  $h-k$  do
11     $s_{m,i} = s_{m,i} \parallel u_{m,d_i}(e_j)$  /*  $\parallel$  denotes concatenation */

```

To eliminate the two vulnerabilities in Sun et al. (2022) scheme, we have done the following respective changes in the initial share blocks generation:

- While generating initial share blocks per secret block B_m corresponding to threshold k , store a univariate polynomial $Q_m(x, d_i)$ and $h-k$ evaluated values of univariate polynomial $Q_m(d_i, y)$ corresponding to $Q_m(x, y)$ in i^{th} ($1 \leq i \leq n$) share block

(from lines 7–11 of Algorithm 1). All this information stored in each share block is necessary to reconstruct $k(h - 1)$ greyscale values in B_m with threshold k (refer Section 5 for proof). Each share block provides no additional information about B_m .

- Use of random integers during the initial share blocks generation to hide the correlation among adjacent pixels in the secret image (from lines 4–6 of Algorithm 1).

4.3 Share update phase

In the share update phase, every r^{th} ($1 \leq r \leq n$) participant updates the initial share S_r to S_r^t corresponding to new threshold t broadcast by the dealer, where $k < t \leq h$. The steps for updating an initial share are as follows. First, divide S_r into b blocks with each block $s_{m,r}$ containing h greyscale values, where $m \in \{1, 2, \dots, b\}$. Then, generate updated block $s_{m,r}^t$ corresponding to $s_{m,r}$ using Algorithm 2. For every $s_{m,r}$ with h greyscale values, corresponding $s_{m,r}^t$ contains $k + h - t$ greyscale values. Finally, generate updated share S_r^t as $S_r^t = s_{1,r}^t \parallel s_{2,r}^t \parallel \dots \parallel s_{b,r}^t$, and delete S_r .

Algorithm 2: Updated share block generation corresponding to new threshold t , where $k < t \leq h$.

Input: h ; new threshold t , where $k < t \leq h$; initial share block

$s_{m,r} = v_{m,d_r}(x) \parallel u_{m,d_r}(e_1) \parallel u_{m,d_r}(e_2) \parallel \dots \parallel u_{m,d_r}(e_{h-k})$, where $r \in \{1, 2, \dots, n\}$.

Output: updated share block $s_{m,r}^t$.

```

1  $s_{m,r}^t = v_{m,d_r}(x)$  /* copy  $v_{m,d_r}(x)$  from  $s_{m,r}$  to  $s_{m,r}^t$  */
2 for  $i = 1$  to  $h - t$  do /* lines 2-3: copy  $h - t$  evaluated values
   of  $u_{m,d_r}(y)$  from  $s_{m,r}$  to  $s_{m,r}^t$  */
3    $s_{m,r}^t = s_{m,r}^t \parallel u_{m,d_r}(e_i)$ 

```

Note that updated share block generation corresponding to final threshold t in Sun et al. (2022) scheme involves only copy operations for $t = h$ and $t = k$. Here, a copy operation refers to an assignment operation. Furthermore, for $k < t < h$, updating a share block involves $h - t$ evaluations of a univariate polynomial in addition to k copy operations (from Section 2.2). However, in the proposed scheme, to accord with the changes made in initial share blocks generation, updated share block generation involves only $k + h - t$ copy operations (from Algorithm 2). Furthermore, in the proposed scheme, each participant updates the initial share only when there is a need for an increase in the threshold.

4.4 Secret image reconstruction phase

In the secret image reconstruction phase, a combiner reconstructs the secret image using t ($k \leq t \leq h$) collected shares $S_{r_1}^t, S_{r_2}^t, \dots, S_{r_t}^t$, where t is the threshold for reconstruction, and $\{r_1, r_2, \dots, r_t\} \subseteq \{1, 2, \dots, n\}$. Note that $\forall l \in \{r_1, r_2, \dots, r_t\}, S_{r_l}^k = S_{r_l}$ because of no share update for $t = k$. The steps for reconstructing the secret image are as follows. First, divide every share $S_{r_l}^t$ into b non-overlapping blocks $s_{1,r_l}^t, s_{2,r_l}^t, \dots, s_{b,r_l}^t$. Then, using every t corresponding share blocks $s_{m,r_1}^t, s_{m,r_2}^t, \dots, s_{m,r_t}^t$, reconstruct secret block B_m using

Algorithm 3 for $k \leq t < h$ or Algorithm 4 for $t = h$, where $m \in \{1, 2, \dots, b\}$. Finally, reconstruct secret image I as $I = B_1 \parallel B_2 \parallel \dots \parallel B_b$.

Algorithm 3: Secret block reconstruction with threshold t , where $k \leq t < h$.

Input: $k; h$; threshold for reconstruction t , where $k \leq t < h$; t share blocks $s_{m,r_1}^t, s_{m,r_2}^t, \dots, s_{m,r_t}^t$, where $\{r_1, r_2, \dots, r_t\} \subseteq \{1, 2, \dots, n\}$, $s_{m,r_l}^k = s_{m,r_l}$, and $s_{m,r_l}^t = v_{m,d_{r_l}}(x) \parallel u_{m,d_{r_l}}(e_1) \parallel u_{m,d_{r_l}}(e_2) \parallel \dots \parallel u_{m,d_{r_l}}(e_{h-t})$, $\forall l \in \{1, 2, \dots, t\}$; t shares' identifying indices $d_{r_1}, d_{r_2}, \dots, d_{r_t}$; $h - t$ publicly known distinct integers $e_1, e_2, \dots, e_{h-t}$.

Output: Reconstructed secret block B_m .

- 1 **for** $i = 1$ **to** $h - t$ **do**
 - 2 Construct univariate polynomial $v_{m,e_i}(x)$ of degree $k - 1$ using Lagrange interpolation with k values $u_{m,d_{r_1}}(e_i), u_{m,d_{r_2}}(e_i), \dots, u_{m,d_{r_k}}(e_i)$ corresponding to k indices $d_{r_1}, d_{r_2}, \dots, d_{r_k}$
 - 3 Reconstruct $k(h - 1)$ coefficients in $Q_m(x, y)$ (shown in (5)) corresponding to $k(h - 1)$ greyscale values in B_m using Lagrange interpolation with h univariate polynomials $v_{m,d_{r_1}}(x), v_{m,d_{r_2}}(x), \dots, v_{m,d_{r_t}}(x)$, $v_{m,e_1}(x), v_{m,e_2}(x), \dots, v_{m,e_{h-t}}(x)$ and h indices $d_{r_1}, d_{r_2}, \dots, d_{r_t}, e_1, e_2, \dots, e_{h-t}$
-

Algorithm 4: Secret block reconstruction with threshold h .

Input: threshold for reconstruction h ; h share blocks $s_{m,r_1}^h, s_{m,r_2}^h, \dots, s_{m,r_h}^h$, where $\{r_1, r_2, \dots, r_h\} \subseteq \{1, 2, \dots, n\}$ and $s_{m,r_l}^h = v_{m,d_{r_l}}(x), \forall l \in \{1, 2, \dots, h\}$; h shares' identifying indices $d_{r_1}, d_{r_2}, \dots, d_{r_h}$.

Output: Reconstructed secret block B_m .

- 1 Reconstruct $k(h - 1)$ coefficients in $Q_m(x, y)$ (shown in (5)) corresponding to $k(h - 1)$ greyscale values in B_m using Lagrange interpolation with h univariate polynomials $v_{m,d_{r_1}}(x), v_{m,d_{r_2}}(x), \dots, v_{m,d_{r_h}}(x)$ and h indices $d_{r_1}, d_{r_2}, \dots, d_{r_h}$
-

Note that respective operations corresponding to secret block reconstruction with threshold $t = h$ and $k < t < h$ are similar in Sun et al. (2022) and the proposed schemes. The difference is that in the proposed scheme $k(h - 1)$ greyscale values per secret block, and in Sun et al. (2022) scheme kh greyscale values per secret block, have to be reconstructed. Furthermore, secret block reconstruction with threshold k in the proposed scheme involves additional interpolations of $h - k$ univariate polynomials (from lines 1–2 of Algorithm 3).

4.5 Illustration

For illustration, we consider $k = 2, h = 4, n = 4$, and a secret block B_m containing $k(h - 1) = 6$ greyscale values shown in (6). Furthermore, respective identifying indices of $n = 4$ shares are $d_1 = 1, d_2 = 2, d_3 = 3$ and $d_4 = 4$. The $h - k = 2$ integers other than identifying indices are $e_1 = 5$ and $e_2 = 6$.

$$B_m = \begin{pmatrix} 189 & 180 & 186 \\ 197 & 183 & 185 \end{pmatrix} \quad (6)$$

4.5.1 Share blocks generation

The asymmetric bivariate polynomial $Q_m(x, y)$ generated corresponding to B_m is shown in equation (7). In equation (7), coefficients 151 and 57 are random integers in the interval [1, 255].

$$Q_m(x, y) = \begin{cases} 189 + 180y + 186y^2 + 151y^3 + \\ 197x + 183xy + 185xy^2 + 57xy^3 \end{cases} \quad (7)$$

We know that $\forall i \in \{1, 2, 3, 4\}$, i^{th} initial share block $s_{m,i} = v_{m,d_i}(x) \parallel u_{m,d_i}(e_1) \parallel u_{m,d_i}(e_2)$. For $i = 1$, $v_{m,d_1}(x) = Q_m(x, d_1) = 36 + 242x$, and $u_{m,d_1}(y) = Q_m(d_1, y) = 120 + 3y + 3y^2 + 174y^3$. Now, $u_{m,d_1}(e_1) = 146$ and $u_{m,d_1}(e_2) = 201$. Hence, $s_{m,1} = v_{m,d_1}(x) \parallel u_{m,d_1}(e_1) \parallel u_{m,d_1}(e_2) = (36, 242, 146, 201)$. Similarly, $s_{m,2} = (214, 189, 141, 163)$, $s_{m,3} = (26, 28, 115, 133)$ and $s_{m,4} = (22, 154, 179, 119)$.

4.5.2 Share block update

For new threshold $t = 3$, $\forall i \in \{1, 2, 3, 4\}$, updated share block $s_{m,i}^3 = v_{m,d_i}(x) \parallel u_{m,d_i}(e_1)$. For $i = 1$, $s_{m,1}^3 = v_{m,d_1}(x) \parallel u_{m,d_1}(e_1) = (36, 242, 146)$. Similarly, $s_{m,2}^3 = (214, 189, 141)$, $s_{m,3}^3 = (26, 28, 115)$ and $s_{m,4}^3 = (22, 154, 179)$.

For new threshold $t = 4$, $\forall i \in \{1, 2, 3, 4\}$, updated share block $s_{m,i}^4 = v_{m,d_i}(x)$. For $i = 1$, $s_{m,1}^4 = v_{m,d_1}(x) = (36, 242)$. Similarly, $s_{m,2}^4 = (214, 189)$, $s_{m,3}^4 = (26, 28)$ and $s_{m,4}^4 = (22, 154)$.

4.5.3 Secret block reconstruction

For simplicity, we consider the first t (final threshold) share blocks for secret block reconstruction, where $2 \leq t \leq 4$.

For $t = 2$, the corresponding 2 share blocks are $s_{m,1} = (36, 242, 146, 201)$ and $s_{m,2} = (214, 189, 141, 163)$. Univariate polynomial $v_{m,e_1}(x)$ constructed using Lagrange interpolation with 2 values $u_{m,d_1}(e_1)$ and $u_{m,d_2}(e_1)$ corresponding to 2 indices d_1 and d_2 is $108 + 254x$. Similarly, univariate polynomial $v_{m,e_2}(x)$ constructed using Lagrange interpolation with 2 values $u_{m,d_1}(e_2)$ and $u_{m,d_2}(e_2)$ corresponding to 2 indices d_1 and d_2 is $239 + 38x$. With 4 univariate polynomials $v_{m,d_1}(x)$, $v_{m,d_2}(x)$, $v_{m,e_1}(x)$ and $v_{m,e_2}(x)$, and corresponding 4 indices d_1, d_2, e_1 , and e_2 , reconstruct B_m as in equation (6) using Lagrange interpolation.

For $t = 3$, the corresponding 3 share blocks are $s_{m,1} = (36, 242, 146)$, $s_{m,2} = (214, 189, 141)$ and $s_{m,3}^3 = (26, 28, 115)$. Univariate polynomial $v_{m,e_1}(x)$ constructed using Lagrange interpolation with 2 values $u_{m,d_1}(e_1)$ and $u_{m,d_2}(e_1)$ corresponding to 2 indices d_1 and d_2 is $108 + 254x$. With 4 univariate polynomials $v_{m,d_1}(x)$, $v_{m,d_2}(x)$, $v_{m,d_3}(x)$ and $v_{m,e_1}(x)$, and corresponding 4 indices d_1, d_2, d_3 , and e_1 , reconstruct B_m as in equation (6) using Lagrange interpolation.

For $t = 4$, the corresponding 4 share blocks are $s_{m,1} = (36, 242)$, $s_{m,2} = (214, 189)$, $s_{m,3}^3 = (26, 28)$ and $s_{m,4}^4 = (22, 154)$. With 4 univariate polynomials $v_{m,d_1}(x)$, $v_{m,d_2}(x)$, $v_{m,d_3}(x)$ and $v_{m,d_4}(x)$, and corresponding 4 indices d_1, d_2, d_3 , and d_4 , reconstruct B_m as in equation (6) using Lagrange interpolation.

5 Theoretical analysis

In this section, we first prove each initial and updated share is random (Theorems 1 and 2). Second, we prove if t is the final threshold, where $k \leq t \leq h$, then less than t colluding participants cannot reconstruct the secret image (Theorems 3–5). Third, we prove if t is the final threshold, where $k \leq t \leq h$, then an external adversary cannot reconstruct the secret image using captured initial and/or updated shares whose total count is less than t (Theorem 6). Fourth, we prove relation between k, h and n must be $n + h - k \leq 255$ (Theorem 7). Fifth, we provide and prove the asymptotic time complexities of the initial share blocks generation, updated share block generation, and secret block reconstruction algorithms (Theorems 8–11). We know from Section 2 that Category-2 TCSIS schemes (Liu et al., 2019; Sun et al., 2022) are preferable over Category-1 TCSIS schemes (Islam et al., 2016; Wu et al., 2021) for supporting the increase in the threshold. In Category-2 TCSIS schemes, Liu et al. (2019) scheme supports only three possible thresholds. However, the state-of-the-art Sun et al. (2022) scheme overcomes this demerit and supports $h - k + 1$ consecutive thresholds. Furthermore, the proposed scheme is a modified version of the Sun et al. (2022) scheme with similar properties. Hence, finally we compare the proposed and Sun et al. (2022) schemes corresponding to secret block size, number of secret blocks, share size, asymptotic time complexities, and number of basic operations.

Theorem 1: *Each initial share generated using the proposed scheme is a random image.*

Proof: We prove that each initial share block $s_{m,i}$ generated for a secret block B_m contains random greyscale values (integers) in the interval $[0, 255]$, where $i \in \{1, 2, \dots, n\}$. This proof holds for all initial share blocks, indicating that each initial share is a random image, i.e., an image containing random greyscale values in the interval $[0, 255]$.

We know that an initial share block $s_{m,i} = v_{m,d_i}(x) \parallel u_{m,d_i}(e_1) \parallel \dots \parallel u_{m,d_i}(e_{h-k})$ (from lines 7–11 of Algorithm 1). Here, $v_{m,d_i}(x) = v'_{m,0}(d_i) + v'_{m,1}(d_i)x + \dots + v'_{m,k-1}(d_i)x^{k-1}$, where $v'_{m,j}(y) = c_{j,0} + c_{j,1}y + \dots + c_{j,h-1}y^{h-1}$ and $j \in \{0, 1, \dots, k-1\}$. Furthermore, $\forall j \in \{0, 1, \dots, k-1\}$, coefficient $c_{j,h-1}$ is a random integer in the interval $[1, 255]$ (from lines 4–5 of Algorithm 1). The remaining coefficients $c_{j,l}, \forall j \in \{0, 1, \dots, k-1\}$ and $\forall l \in \{0, 1, \dots, h-2\}$, are the greyscale values in the interval $[0, 255]$ belonging to B_m (from lines 1–3 of Algorithm 1). Recall that all operations occur in $GF(2^8)$. Because $c_{j,h-1}$ is a random integer in the interval $[1, 255] \forall j \in \{0, 1, \dots, k-1\}$, $v'_{m,j}(d_i)$ is also a random integer in the interval $[0, 255]$ (Bhat, 2021). Hence, every coefficient in $v_{m,d_i}(x)$ is a random integer in the interval $[0, 255]$.

Furthermore, $u_{m,d_i}(y) = u'_{m,0}(d_i) + u'_{m,1}(d_i)y + \dots + u'_{m,h-1}(d_i)y^{h-1}$, where $u'_{m,l}(x) = c_{0,l} + c_{1,l}x + \dots + c_{k-1,l}x^{k-1}$ and $l \in \{0, 1, \dots, h-1\}$. Note that, $\forall j \in \{0, 1, \dots, k-1\}$, coefficient $c_{j,h-1}$ is a random integer in the interval $[1, 255]$ (from lines 4–5 of Algorithm 1). The remaining coefficients $c_{j,l}, \forall j \in \{0, 1, \dots, k-1\}$ and $\forall l \in \{0, 1, \dots, h-2\}$, are the greyscale values in the interval $[0, 255]$ belonging to B_m (from lines 1–3 of Algorithm 1). Thus, $u'_{m,h-1}(d_i)$ is also a random integer in the interval $[0, 255]$ because $c_{j,h-1}$ is a random integer in the interval $[1, 255], \forall j \in \{0, 1, \dots, k-1\}$ (Bhat, 2021)). Because $u'_{m,h-1}(d_i)$ is a random integer in the interval $[0, 255]$, $u_{m,d_i}(e_1), u_{m,d_i}(e_2), \dots, u_{m,d_i}(e_{h-k})$ are also random integers in the interval $[0, 255]$ (Bhat, 2021)).

Thus, $s_{m,i} = v_{m,d_i}(x) \parallel u_{m,d_i}(e_1) \parallel \dots \parallel u_{m,d_i}(e_{h-k})$ contains random greyscale values (integers) in the interval $[0, 255]$. $\square$

Theorem 2: *Each updated share generated using the proposed scheme is a random image.*

Proof: We prove that an updated share block $s_{m,r}^t$ generated corresponding to new threshold t contains random greyscale values (integers) in the interval $[0, 255]$, where $k < t \leq h$ and $r \in \{1, 2, \dots, n\}$. This proof holds for all updated share blocks, indicating that each updated share is a random image.

We know that $s_{m,r}^t$ is generated from the corresponding initial share block $s_{m,r} = v_{m,d_r}(x) \parallel u_{m,d_r}(e_1) \parallel \dots \parallel u_{m,d_r}(e_{h-k})$ by copying $v_{m,d_r}(x), u_{m,d_r}(e_1), \dots, u_{m,d_r}(e_{h-t})$ from $s_{m,r}$ into $s_{m,r}^t$ (from Algorithm 2). We know from the proof of Theorem 1 that the all coefficients in $v_{m,d_r}(x)$, and the values $u_{m,d_r}(e_1), u_{m,d_r}(e_2), \dots, u_{m,d_r}(e_{h-t})$ are random integers in the interval $[0, 255]$.

Thus, $s_{m,r}^t = v_{m,d_r}(x) \parallel u_{m,d_r}(e_1) \parallel \dots \parallel u_{m,d_r}(e_{h-t})$ contains random greyscale values (integers) in the interval $[0, 255]$. $\square$

Theorem 3: *If k is the current threshold, then a combiner (dealer or participant) requires at least k shares corresponding to threshold k to reconstruct the secret image.*

Proof: We prove that for current threshold k , a combiner requires at least k corresponding initial share blocks $s_{m,r_1}, s_{m,r_2}, \dots, s_{m,r_k}$ to reconstruct the secret block B_m , where $\{r_1, r_2, \dots, r_k\} \subseteq \{1, 2, \dots, n\}$. This proof holds for all blocks of the secret image.

To reconstruct B_m , a combiner requires $k(h-1)$ corresponding coefficients in the asymmetric bivariate polynomial $Q_m(x, y)$. An initial share block $s_{m,r_i} = v_{m,d_{r_i}}(x) \parallel u_{m,d_{r_i}}(e_1) \parallel \dots \parallel u_{m,d_{r_i}}(e_{h-k})$ (from lines 7–11 of Algorithm 1), where $i \in \{1, 2, \dots, k\}$. To reconstruct $k(h-1)$ corresponding coefficients in $Q_m(x, y)$ using polynomial interpolation, a combiner requires at least h univariate polynomials $v_{m,d_{r_1}}(x), v_{m,d_{r_2}}(x), \dots, v_{m,d_{r_k}}(x), v_{m,e_1}(x), v_{m,e_2}(x), \dots, v_{m,e_{h-k}}(x)$. First, the combiner can obtain k univariate polynomials $v_{m,d_{r_1}}(x), v_{m,d_{r_2}}(x), \dots, v_{m,d_{r_k}}(x)$ directly from at least k corresponding initial share blocks. Next, the combiner can compute each one of $h-k$ univariate polynomials $v_{m,e_1}(x), v_{m,e_2}(x), \dots, v_{m,e_{h-k}}(x)$ from at least k corresponding values $u_{m,d_{r_1}}(e_j), u_{m,d_{r_2}}(e_j), \dots, u_{m,d_{r_k}}(e_j)$, where $j \in \{1, 2, \dots, h-k\}$. Hence, a combiner requires at least k corresponding share blocks to reconstruct B_m using polynomial interpolation.

Another way to reconstruct $Q_m(x, y)$ is by solving linearly independent equations. The number of linearly independent equations formed using $k-1$ initial share blocks corresponding to kh unknowns is $k(k-1)$. Here, kh is the number of coefficients in $Q_m(x, y)$. Because $k(k-1) < kh$, a combiner cannot reconstruct $Q_m(x, y)$ using less than k shares and cannot uniquely determine any coefficient in $Q_m(x, y)$. Hence, using less than k corresponding initial share blocks, a combiner cannot reconstruct B_m by solving linearly independent equations. $\square$

Theorem 4: *If t ($k < t < h$) is the final threshold, then a combiner (dealer or participant) requires at least t shares corresponding to threshold t to reconstruct the secret image.*

Proof: We prove that for final threshold t ($k < t < h$), a combiner requires at least t corresponding updated share blocks $s_{m,r_1}^t, s_{m,r_2}^t, \dots, s_{m,r_t}^t$ to reconstruct the secret block B_m , where $\{r_1, r_2, \dots, r_t\} \subseteq \{1, 2, \dots, n\}$. This proof holds for all blocks of the secret image.

To reconstruct B_m , a combiner requires $k(h-1)$ corresponding coefficients in the asymmetric bivariate polynomial $Q_m(x, y)$. An updated share block $s_{m,r_i}^t = v_{m,d_{r_i}}(x) \parallel$

$u_{m,d_{r_i}}(e_1) \parallel \dots \parallel u_{m,d_{r_i}}(e_{h-t})$ (from Algorithm 2), where $i \in \{1, 2, \dots, t\}$. To reconstruct $k(h-1)$ corresponding coefficients in $Q_m(x, y)$ using polynomial interpolation, a combiner requires at least h univariate polynomials $v_{m,d_{r_1}}(x), v_{m,d_{r_2}}(x), \dots, v_{m,d_{r_t}}(x), v_{m,e_1}(x), v_{m,e_2}(x), \dots, v_{m,e_{h-t}}(x)$. First, the combiner can obtain t univariate polynomials $v_{m,d_{r_1}}(x), v_{m,d_{r_2}}(x), \dots, v_{m,d_{r_t}}(x)$ directly from at least t corresponding share blocks. Next, the combiner can compute each one of $h-t$ univariate polynomials $v_{m,e_1}(x), v_{m,e_2}(x), \dots, v_{m,e_{h-t}}(x)$ from at least k corresponding values $u_{m,d_{r_1}}(e_j), u_{m,d_{r_2}}(e_j), \dots, u_{m,d_{r_k}}(e_j)$, where $j \in \{1, 2, \dots, h-t\}$. Hence, a combiner requires at least t corresponding share blocks to reconstruct B_m using polynomial interpolation.

Another way to reconstruct $Q_m(x, y)$ is by solving linearly independent equations. The number of linearly independent equations formed using $t-1$ updated share blocks corresponding to kh unknowns is $k(t-1) + k(h-t) = kh - k$. Here, kh is the number of coefficients in $Q_m(x, y)$. Because $kh - k < kh$, a combiner cannot reconstruct $Q_m(x, y)$ using less than t shares, and cannot uniquely determine any coefficient in $Q_m(x, y)$. Hence, using less than t corresponding updated share blocks, a combiner cannot reconstruct B_m by solving linearly independent equations. $\square$

Theorem 5: *If h is the final threshold, then a combiner (dealer or participant) requires at least h shares corresponding to threshold h to reconstruct the secret image.*

Proof: We prove that for final threshold h , a combiner requires at least h corresponding updated share blocks $s_{m,r_1}^h, s_{m,r_2}^h, \dots, s_{m,r_h}^h$ to reconstruct the secret block B_m , where $\{r_1, r_2, \dots, r_h\} \subseteq \{1, 2, \dots, n\}$. This proof holds for all blocks of the secret image.

To reconstruct B_m , a combiner requires $k(h-1)$ corresponding coefficients in the asymmetric bivariate polynomial $Q_m(x, y)$. An updated share block $s_{m,r_i}^h = v_{m,d_{r_i}}(x)$ (from Algorithm 2), where $i \in \{1, 2, \dots, h\}$. To reconstruct $k(h-1)$ corresponding coefficients in $Q_m(x, y)$ using polynomial interpolation, a combiner requires at least h univariate polynomials $v_{m,d_{r_1}}(x), v_{m,d_{r_2}}(x), \dots, v_{m,d_{r_h}}(x)$. The combiner can only obtain these h univariate polynomials from at least h corresponding share blocks. Hence, a combiner requires at least h corresponding share blocks to reconstruct B_m using polynomial interpolation.

Another way to reconstruct $Q_m(x, y)$ is by solving linearly independent equations. The number of linearly independent equations formed using $h-1$ updated share blocks corresponding to kh unknowns is $k(h-1) = kh - k$. Here, kh is the number of coefficients in $Q_m(x, y)$. Because $kh - k < kh$, a combiner cannot reconstruct $Q_m(x, y)$ using less than h shares, and cannot uniquely determine any coefficient in $Q_m(x, y)$. Hence, using less than h corresponding updated share blocks, a combiner cannot reconstruct B_m by solving linearly independent equations. $\square$

Theorem 6: *If t is the current threshold, where $k \leq t \leq h$, then an external adversary E_A cannot reconstruct the secret image using $t-1$ captured shares corresponding to thresholds in the set $T = \{t' : k \leq t' \leq t\}$.*

Proof: We prove that for current threshold t , where $k \leq t \leq h$, E_A cannot reconstruct a secret block B_m using $t-1$ corresponding share blocks corresponding to thresholds in the set $T = \{t' : k \leq t' \leq t\}$. We provide proof corresponding to three possible cases: (i) $t = k$; (ii) $t > k$ and $t < h$; (iii) $t = h$. This proof holds for all blocks of the secret image.

Case (i) $t = k$:

Here, the set $T = \{k\}$. We know from proof of Theorem 3 that using $k - 1$ corresponding initial share blocks, E_A cannot reconstruct B_m .

Case (ii) $t > k$ and $t < h$:

Here, the set $T = \{k, t\}$. We know that the lesser the threshold, the more information about a secret block is present in each corresponding share block. Hence, the best case for E_A is to have captured $k - 1$ initial shares, and $t - k$ updated shares (total $t - 1$ shares). Without loss of generality, we assume that for B_m , $k - 1$ corresponding initial share blocks are $s_{m,1}, s_{m,2}, \dots, s_{m,k-1}$. Furthermore, $t - k$ corresponding updated share blocks are $s_{m,k}^t, s_{m,k+1}^t, \dots, s_{m,t-1}^t$. We now show that E_A cannot reconstruct B_m using these $k - 1 + t - k = t - 1$ share blocks either by polynomial interpolation or by solving linearly independent equations.

Each initial share block $s_{m,i} = v_{m,d_i}(x) \parallel u_{m,d_i}(e_1) \parallel \dots \parallel u_{m,d_i}(e_{h-k})$ (from lines 7–11 of Algorithm 1), where $i \in \{1, 2, \dots, k - 1\}$. Furthermore, each updated share block $s_{m,i}^t = v_{m,d_i}(x) \parallel u_{m,d_i}(e_1) \parallel \dots \parallel u_{m,d_i}(e_{h-t})$ (from Algorithm 2), where $i \in \{k, k + 1, \dots, t - 1\}$. To reconstruct B_m , E_A requires $k(h - 1)$ corresponding coefficients in the asymmetric bivariate polynomial $Q_m(x, y)$. To compute these coefficients using polynomial interpolation, E_A requires h univariate polynomials $v_{m,l_1}(x), v_{m,l_2}(x), \dots, v_{m,l_h}(x)$, where $\{l_1, l_2, \dots, l_h\} \subset \{d_1, d_2, \dots, d_n, e_1, e_2, \dots, e_{h-k}\}$ (from Algorithm 3). From $t - 1$ share blocks, E_A can directly obtain $t - 1$ univariate polynomials $v_{m,d_1}(x), v_{m,d_2}(x), \dots, v_{m,d_{t-1}}(x)$. However, E_A can construct only $h - t$ more univariate polynomials $v_{m,e_1}(x), v_{m,e_2}(x), \dots, v_{m,e_{h-t}}(x)$ from any k share blocks using polynomial interpolation (from lines 1–2 of Algorithm 3). Hence, there are only $t - 1 + h - t = h - 1$ univariate polynomials available, and E_A cannot reconstruct B_m using polynomial interpolation.

For $t - k < k$, the number of linearly independent equations formed corresponding to kh unknowns using $k - 1$ initial share blocks and $t - k$ updated share blocks is $(k - 1)k + (t - k)k = kt - k$. Here, kh is the number of coefficients in $Q_m(x, y)$. Similarly, for $t - k \geq k$, the number of linearly independent equations formed is $(k - 1)k + (t - k)k + (h - t)k = kh - k$. Because $kt - k < kh$ and $kh - k < kh$, E_A cannot reconstruct $Q_m(x, y)$, and cannot uniquely determine any coefficient in $Q_m(x, y)$. Hence, E_A cannot reconstruct B_m by solving linearly independent equations.

Case (iii) $t = h$:

Here, the set $T = \{k, h\}$. The best case for E_A is to have captured $k - 1$ initial shares, and $h - k$ updated shares (total $h - 1$ shares). Without loss of generality, we assume that for B_m , $k - 1$ corresponding initial share blocks are $s_{m,1}, s_{m,2}, \dots, s_{m,k-1}$. Furthermore, $h - k$ corresponding updated share blocks are $s_{m,k}^h, s_{m,k+1}^h, \dots, s_{m,h-1}^h$. We now show that E_A cannot reconstruct B_m using these $k - 1 + h - k = h - 1$ share blocks either by polynomial interpolation or by solving linearly independent equations.

Each initial share block $s_{m,i} = v_{m,d_i}(x) \parallel u_{m,d_i}(e_1) \parallel \dots \parallel u_{m,d_i}(e_{h-k})$ (from lines 7–11 of Algorithm 1), where $i \in \{1, 2, \dots, k - 1\}$. Furthermore, each updated share block $s_{m,i}^h = v_{m,d_i}(x)$ (from Algorithm 2), where $i \in \{k, k + 1, \dots, h - 1\}$. To reconstruct B_m , E_A requires $k(h - 1)$ corresponding coefficients in the asymmetric bivariate polynomial $Q_m(x, y)$. To compute these coefficients using polynomial interpolation, E_A requires h univariate polynomials $v_{m,l_1}(x), v_{m,l_2}(x), \dots, v_{m,l_h}(x)$,

where $\{l_1, l_2, \dots, l_h\} \subset \{d_1, d_2, \dots, d_n, e_1, e_2, \dots, e_{h-k}\}$ (from Algorithms 3 and 4). From $h-1$ share blocks, E_A can directly obtain $h-1$ univariate polynomials $v_{m,d_1}(x), v_{m,d_2}(x), \dots, v_{m,d_{h-1}}(x)$. However, E_A cannot construct any of univariate polynomials $v_{m,e_1}(x), v_{m,e_2}(x), \dots, v_{m,e_{h-k}}(x)$ because there are only $k-1$ evaluated values $u_{m,d_1}(e_j), u_{m,d_2}(e_j), \dots, u_{m,d_{k-1}}(e_j)$ for each $j \in \{1, 2, \dots, h-k\}$. Hence, only $h-1$ univariate polynomials are available, and E_A cannot reconstruct B_m using polynomial interpolation.

The number of linearly independent equations formed corresponding to kh unknowns using $k-1$ initial share blocks and $h-k$ updated share blocks is $(k-1)k + (h-k)k = kh - k$. Here, kh is the number of coefficients in $Q_m(x, y)$. Because $kh - k < kh$, E_A cannot reconstruct $Q_m(x, y)$, and cannot uniquely determine any coefficient in $Q_m(x, y)$. Hence, E_A cannot reconstruct B_m by solving linearly independent equations. $\square$

Theorem 7: *The relation between k , h , and n must be $n + h - k \leq 255$.*

Proof: The number of distinct identifying indices of n shares is 255 in the interval $[1, 255]$. Furthermore, maximum possible number of additional distinct integers other than identifying indices of n shares is $h-k$ in the interval $[1, 255]$ (from Algorithm 1). Hence, the relation $n + h - k \leq 255$ must hold. $\square$

We know from Section 2.1 that in Category-1 TCSIS schemes, the dealer holds the key(s) necessary for reconstructing the secret image corresponding to each one of z supported thresholds $k_1, k_2, \dots, k_z$, where $2 \leq k_1 < k_2 < \dots < k_z \leq n$, till the end of the lifespan of that secret image. If an external adversary captures k_j shares or the number of colluding participants increases to k_j before the end of the lifespan of the secret image, then the dealer chooses k_{j+1} as the threshold for reconstructing the secret image. Furthermore, the dealer broadcasts k_{j+1} and the respective key(s) to all combiners. However, an external adversary can capture the key(s) corresponding to the minimum threshold k_1 from the dealer after that dealer has securely distributed the shares to the participants. Furthermore, by capturing k_1 shares from the participants, an external adversary can reconstruct the secret image before the end of the lifespan of that secret image. However, the dealer in the proposed scheme (a Category-2 TCSIS scheme) does not hold any data after securely distributing the shares to the participants. The only data the dealer broadcasts to all combiners is the new threshold t , where $k < t \leq h$, before an external adversary captures k shares or the number of colluding participants increases to k . Then, each participant generates the updated share corresponding to t . An external adversary with a maximum of $t-1$ captured shares, or a maximum of $t-1$ colluding participants, cannot reconstruct the secret image before the end of the lifespan of that secret image. Thus, the share update mechanism in the proposed scheme is more effective than the mechanism in Category-1 TCSIS schemes.

We can evaluate a univariate polynomial of degree $t-1$ using Horner's rule with asymptotic time complexity $O(t)$ involving $t-1$ additions and $t-1$ multiplications. A combiner can reconstruct the asymmetric bivariate polynomial $Q_m(x, y)$ shown in equation (5) with h univariate polynomials $v_{m,l_1}, v_{m,l_2}, \dots, v_{m,l_h}$ using equations (8) and (9) representing Lagrange interpolation, where $\{l_1, l_2, \dots, l_h\} \subset \{d_1, d_2, \dots, d_n, e_1, e_2, \dots, e_{h-k}\}$ (from Algorithms 3 and 4). For reconstructing every $Q_m(x, y)$, a combiner needs to compute the polynomials $w_i(y) \forall i \in [1, h]$ shown in equation (9) once and store them. Note that in $Q_m(x, y)$, the k coefficients $c_{0,h-1}, c_{1,h-1}, \dots, c_{k-1,h-1}$ are random integers, which are not part of the corresponding

secret block B_m . Hence, a combiner needs to compute only $k(h-1)$ coefficients in $Q_m(x, y)$. With the stored polynomials $w_i(y) \forall i \in [1, h]$, computing a coefficient involves $h-1$ additions and h multiplications. Hence, asymptotic time complexity to compute $k(h-1)$ coefficients in $Q_m(x, y)$ is $O(kh^2)$.

$$Q_m(x, y) = \sum_{i=1}^h v_{m, l_i}(x) w_i(y) \quad (8)$$

$$w_i(y) = \prod_{j=l_1, j \neq l_i}^{l_h} \frac{y - j}{l_i - j} \quad (9)$$

Similarly, at line 2 of Algorithm 3, a combiner can compute univariate polynomial $v_{m, e_i}(x)$ with k values $u_{m, d_{r_1}}(e_i), u_{m, d_{r_2}}(e_i), \dots, u_{m, d_{r_k}}(e_i)$ corresponding to k indices $d_{r_1}, d_{r_2}, \dots, d_{r_k}$ using equations (10) and (11) representing Lagrange interpolation. Here as well, a combiner needs to compute the univariate polynomials $z_j(x) \forall j \in [1, k]$ shown in equation (11) once and store them. With these stored polynomials, a combiner can compute k coefficients in each $v_{m, e_i}(x)$ with $k-1$ additions and k multiplications per coefficient leading to asymptotic time complexity $O(k^2)$.

$$v_{m, e_i}(x) = \sum_{j=1}^k u_{m, d_{r_j}}(e_i) z_j(x) \quad (10)$$

$$z_j(x) = \prod_{l=1, l \neq j}^k \frac{x - d_{r_l}}{d_{r_j} - d_{r_l}} \quad (11)$$

Theorem 8: *The asymptotic time complexity of initial share blocks generation (Algorithm 1) is $O(n(h^2 + kh))$.*

Proof: The line numbers specified in this proof correspond to Algorithm 1. The asymptotic time complexity of operations in lines 1–6 is $O(kh)$. Line 8 contains evaluation of k univariate polynomials, each of degree $h-1$, involving $k(h-1)$ additions and $k(h-1)$ multiplications. Line 9 contains evaluation of h univariate polynomials, each of degree $k-1$, involving $h(k-1)$ additions and $h(k-1)$ multiplications. Lines 10–11 contain evaluation of $h-k$ univariate polynomials, each of degree $h-1$, involving $(h-k)(h-1)$ additions and $(h-k)(h-1)$ multiplications. Hence, the asymptotic time complexity of operations in lines 8–11 is $O(h^2 + kh)$ based on $h(h-1) + h(k-1)$ additions and $h(h-1) + h(k-1)$ multiplications. Because of repeating operations in lines 8–11 n times, asymptotic time complexity of operations in lines 9–11 is $O(n(h^2 + kh))$. Hence, total asymptotic time complexity is $O(n(h^2 + kh))$. $\square$

Theorem 8 implies that as n increases linearly, keeping k and h fixed, the execution time for initial share blocks generation increases linearly. This property holds for a linear increase in k , keeping h and n fixed. However, the increase in the execution time is quadratic for a linear increase in h , keeping k and n fixed.

Theorem 9: *The asymptotic time complexity of updated share block generation (Algorithm 2) corresponding to new threshold t is $O(k + h - t)$ for $k < t < h$, and $O(k)$ for $t = h$.*

Proof: The line numbers specified in this proof correspond to Algorithm 2. For $t = h$, a participant assigns k coefficients corresponding to $v_{m,d_r}(x)$ from initial share block $s_{m,r}$ to updated share block $s_{m,r}^t$ (line 1) with asymptotic time complexity $O(k)$. For $k < t < h$, the participant assigns additional $h - t$ values from $s_{m,r}$ to $s_{m,r}^t$ (lines 2–3) with asymptotic time complexity $O(h - t)$. Hence, asymptotic time complexity is $O(k + h - t)$ for $k < t < h$, and $O(k)$ for $t = h$. $\square$

Theorem 9 implies that the execution time for updated share block generation increases linearly for a linear increase in k and/or h , keeping t fixed, where $k < t < h$. Furthermore, the execution time decreases linearly for a linear increase in t , keeping k and h fixed. However, for $t = h$, the execution time increases linearly for a linear increase in k .

Theorem 10: *The asymptotic time complexity of secret block reconstruction with threshold t , where $k \leq t < h$ (Algorithm 3) is $O((h - t)k^2 + kh^2)$.*

Proof: The line numbers specified in this proof correspond to Algorithm 3. Lines 1–2 contain interpolation of $h - t$ univariate polynomials, each of degree $k - 1$, involving $k(h - t)(k - 1)$ additions and $k^2(h - t)$ multiplications. Hence, the asymptotic time complexity of operations in lines 1–2 is $O((h - t)k^2)$ based on $k(h - t)(k - 1) + k^2(h - t)$ additions and multiplications. Line 3 contains computation of $k(h - 1)$ coefficients in $Q_m(x, y)$ involving $k(h - 1)^2$ additions and $kh(h - 1)$ multiplications. Hence, the asymptotic time complexity of operations in line 3 is $O(kh^2)$ based on $k(h - 1)^2 + kh(h - 1)$ additions and multiplications. The total asymptotic time complexity is $O((h - t)k^2 + kh^2)$. $\square$

Theorem 10 implies that the increase in the execution time for secret block reconstruction with threshold t is quadratic for a linear increase in k , keeping h and t fixed, where $k \leq t < h$. This property holds for a linear increase in h , keeping k and t fixed. However, the execution time decreases linearly for a linear increase in t , keeping k and h fixed.

Theorem 11: *The asymptotic time complexity of secret block reconstruction with threshold h (Algorithm 4) is $O(kh^2)$.*

Proof: Line 1 of Algorithm 4 contains computation of $k(h - 1)$ coefficients in $Q_m(x, y)$ involving $k(h - 1)^2$ additions and $kh(h - 1)$ multiplications. Hence, the asymptotic time complexity of operations in line 1 is $O(kh^2)$ based on $k(h - 1)^2 + kh(h - 1)$ additions and multiplications, which itself is the total asymptotic time complexity. $\square$

Theorem 11 implies that the increase in the execution time for secret block reconstruction with threshold h is quadratic for a linear increase in h , keeping k fixed. However, the execution time increases linearly for a linear increase in k , keeping h fixed.

We now illustrate the calculation of the number of basic operations in share blocks generation, updated share block generation, and secret block reconstruction. We know that additions and multiplications are the basic operations in share blocks generation and secret block reconstruction (from the proofs of Theorems 8, 10 and 11). Furthermore, copy, i.e., assignment, operations are the basic operations in updated share block generation

(from the proof of Theorem 9). Consider that $k = 2$, $h = 10$ and $n = 10$. We know from the proof of Theorem 8 that the total number of additions and multiplications in share blocks generation is $2n(h(h-1) + h(k-1)) = 2 \times 10 \times (10 \times (10-1) + 10 \times (2-1)) = 2000$. Consider that we now generate the updated share block corresponding to the new threshold $t = 5$. We know from the proof of Theorem 9 that the total number of copy operations in updated share block generation corresponding to t , where $k < t < h$, is $k + h - t = 2 + 10 - 5 = 7$. Now, consider $t = 5$ as the final threshold for secret block reconstruction. We know from the proof of Theorem 10 that the total number of additions and multiplications in secret block reconstruction with threshold t , where $k \leq t < h$, is $k(h-t)(k-1) + k^2(h-t) + k(h-1)^2 + kh(h-1) = 2 \times (10-5) \times (2-1) + 2^2 \times (10-5) + 2 \times (10-1)^2 + 2 \times 10 \times (10-1) = 372$.

In Table 3, we compare the proposed and Sun et al. (2022) schemes corresponding to secret block size, number of secret blocks, and share size. We draw the following implications from Table 3:

- The number of blocks formed in the secret image using the proposed scheme is more than that using Sun et al. (2022) scheme.
- The size of each initial share generated using the proposed scheme is lesser than that using Sun et al. (2022) scheme.
- The size of each updated share generated using the proposed scheme is more than that using Sun et al. (2022) scheme.

Table 3 Comparison of secret block size, number of secret blocks and share size

<i>Scheme</i>	<i>Secret block size</i>	<i>Number of secret blocks</i>	<i>Initial share size</i>
Sun et al. (2022)	kh	z^*/h	$(z/h)(k+h)$
Proposed	$k(h-1)$	$z/(h-1)$	$zh/(h-1)$
<i>Updated share size</i>			
<i>Scheme</i>	$t = k$	$k < t < h$	$t = h$
Sun et al. (2022)	z	$(z/h)(k+h-t)$	G/h
Proposed	–	$(z/(h-1))(k+h-t)$	$G/(h-1)$

* $z = G/k$.

In Table 4, we compare the proposed and Sun et al. (2022) schemes corresponding to the asymptotic time complexities of the initial share blocks generation, updated share block generation, and secret block reconstruction. We draw the following implications from Table 4:

- The asymptotic time complexity of the initial share blocks generation, and that of the secret block reconstruction with threshold $t = k$, are more using the proposed scheme than that using Sun et al. (2022) scheme.
- The asymptotic time complexity of updated share block generation corresponding to threshold t , where $k < t < h$, is lesser using the proposed scheme than that using Sun et al. (2022) scheme.
- The asymptotic time complexity of updated share block generation corresponding to threshold $t = h$, and that of secret block reconstruction with threshold t , where $k < t \leq h$, are same using the proposed and Sun et al. (2022) schemes.

Table 4 Comparison of asymptotic time complexities

Scheme	Initial share blocks generation	Updated share block generation		
		$t = k$	$k < t < h$	$t = h$
Sun et al. (2022)	$O(nkh)$	$O(h)$	$O(k + (h - t)h)$	$O(k)$
Proposed	$O(n(h^2 + kh))$	–	$O(k + h - t)$	$O(k)$
Scheme	Secret block reconstruction			
	$t = k$	$k < t < h$	$t = h$	
Sun et al. (2022)	$O(hk^2)$	$O((h - t)k^2 + kh^2)$	$O(kh^2)$	
Proposed	$O((h - k)k^2 + kh^2)$	$O((h - t)k^2 + kh^2)$	$O(kh^2)$	

We know the basic operations in the shares generation, share update, and secret image reconstruction phases of the proposed and Sun et al. (2022) schemes are addition, multiplication, and copy. In Tables 5–7, we compare the proposed and Sun et al. (2022) schemes corresponding to the number of basic operations in these three phases, respectively. Note that the shares generation and secret image reconstruction phases do not have copy operations. Furthermore, the proposed scheme involves no share update phase corresponding to k as the final threshold. We draw the following implications from Tables 5–7:

Table 5 Comparison of number of basic operations in the shares generation phase

Operation	Scheme	Number of operations
Addition	Sun et al. (2022)	$z_0^*(k(h - 1) + h(k - 1))$
	Proposed	$z_1^{**}(h(h - 1) + h(k - 1))$
Multiplication	Sun et al. (2022)	$z_0(k(h - 1) + h(k - 1))$
	Proposed	$z_1(h(h - 1) + h(k - 1))$

* $z_0 = Gn/(kh)$; ** $z_1 = Gn/(k(h - 1))$.

Table 6 Comparison of number of basic operations in updating a share in the share update phase

Operation	Scheme	Number of operations		
		$t = k$	$k < t < h$	$t = h$
Addition	Sun et al. (2022)	0	$z_2^*(h - t)(h - 1)$	0
	Proposed	–	0	0
Multiplication	Sun et al. (2022)	0	$z_2(h - t)(h - 1)$	0
	Proposed	–	0	0
Copy	Sun et al. (2022)	G/k	G/h	G/h
	Proposed	–	$z_3^{**}(k + h - t)$	$G/(h - 1)$

* $z_2 = G/(kh)$; ** $z_3 = G/(k(h - 1))$.

- The respective numbers of addition and multiplication operations for generating initial shares using the proposed scheme are more than that using Sun et al. (2022) scheme. Hence, the execution time for generation of initial shares using the proposed scheme is more than that using Sun et al. (2022) scheme.

Table 7 Comparison of number of basic operations in reconstructing the secret image by a combiner in the secret image reconstruction phase

Operation	Scheme	Number of operations		
		$t = k$	$k < t < h$	$t = h$
Addition	Sun et al. (2022)	z_4^*	$(z_4 z_5^\# / h) + z_6^{**}$	z_6
	Proposed	$(z_4 z_7^\$ / (h - 1)) + z_6$	$(z_4 z_5 / (h - 1)) + z_6$	z_6
Multiplication	Sun et al. (2022)	$z_8^{\#\#}$	$(z_8 z_5 / h) + z_9^\$$	z_9
	Proposed	$(z_8 z_7 / (h - 1)) + z_9$	$(z_8 z_5 / (h - 1)) + z_9$	z_9

* $z_4 = G(k - 1)$; $\# z_5 = h - t$; ** $z_6 = G(h - 1)$; $\$ z_7 = h - k$.
 $\#\# z_8 = Gk$; $\$ z_9 = Gh$.

- The execution time for updating a share corresponding to threshold t , where $k < t < h$, using the proposed scheme is lesser than that using Sun et al. (2022) scheme. The reason is that in the proposed scheme, updating a share involves only copy (assignment) operations. However, in Sun et al. (2022) scheme, updating a share involves addition, multiplication, and copy operations. The number of copy operations to update a share using the proposed scheme is lesser than the total number of addition, multiplication, and copy operations to update a share using Sun et al. (2022) scheme. We know that the execution time of a copy (assignment) operation is lesser than that of an addition or multiplication operation. Hence, the proposed scheme requires comparatively lesser execution time.
- The execution time for updating a share corresponding to threshold $t = h$ using the proposed scheme is more than that using Sun et al. (2022) scheme. The reason is the number of copy operations to update a share using the proposed scheme is more than that using Sun et al. (2022) scheme. Note that the share update phase corresponding to threshold $t = h$ has only copy operations in both proposed and Sun et al. (2022) schemes.
- The respective numbers of addition and multiplication operations for reconstructing the secret image with threshold t using the proposed scheme are more than that using Sun et al. (2022) scheme, where $k \leq t < h$. Hence, the respective execution time for reconstruction of the secret image using the proposed scheme is more than that using Sun et al. (2022) scheme.
- The respective numbers of addition and multiplication operations for reconstructing the secret image with threshold $t = h$ using the proposed and Sun et al. (2022) schemes are the same. Hence, the respective execution time for reconstruction of the secret image using the proposed scheme is approximately equal to that using Sun et al. (2022) scheme.

6 Experimental results

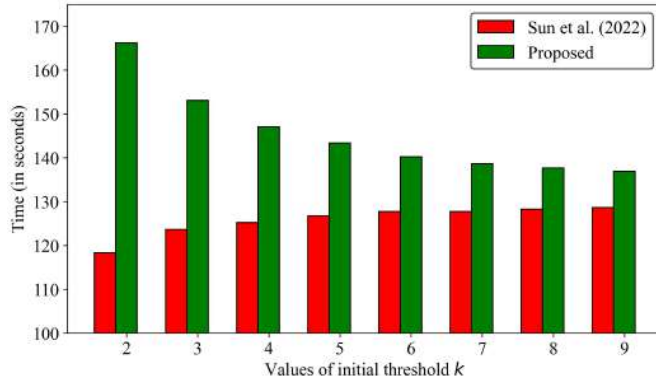
In this section, we first compare the execution time for generation of initial shares, updating a share, and reconstruction of the secret image using the state-of-the-art Sun et al. (2022) and the proposed schemes. We already have provided the reason for comparing the execution time only with Sun et al. (2022) scheme in Section 5. Second, we show examples of visual information leakage through individual share generated using Liu et al. (2019) and Sun et al.

(2022) schemes. Third, we show an example of visually random-looking shares generated using the proposed scheme. Finally, we compare the Correlation Coefficient (CC) of adjacent pixels in individual share generated using Liu et al. (2019), Sun et al. (2022), and the proposed schemes.

We use Python3 programming language in a computer system with Windows 10 64-bit operating system, Intel(R) Core(TM) i7-9700 CPU @ 3.00GHz processor, and 16.0 GB RAM for experimentation. We have performed experimentation with several images as secret. However, we explain the execution time corresponding to a greyscale secret image of size 2520×2520 . Furthermore, we show and compare the randomness in each generated share corresponding to a greyscale secret image of size 336×336 .

In Figure 2, we compare Sun et al. (2022) and the proposed schemes corresponding to the execution time for generation of initial shares with $k \in [2, 9]$, $h = 10$, and $n = 10$. We observe that the execution time corresponding to the proposed scheme is more than that corresponding to Sun et al. (2022) scheme as indicated in Section 5. For the proposed scheme, we observe that the execution time decreases with an increase in k . The reason is as follows. We know that the total number of basic operations in the generation of initial shares is $2Gnh(h + k - 2)/(k(h - 1))$ (from Table 5). Because the value of the expression $(h + k - 2)/k$ decreases for increase in k , keeping G , n , and h fixed, the number of basic operations also decreases. Due to similar reasons, the execution time for generating initial shares using the proposed scheme increases by individually increasing h , n , and G , keeping the other variables fixed.

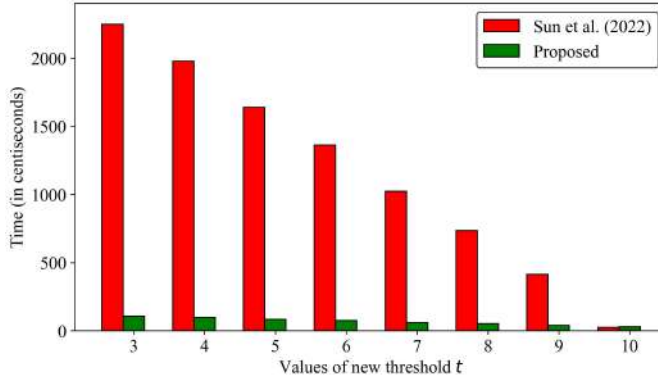
Figure 2 Comparison of execution time for generation of initial shares with $k \in [2, 9]$, $h = 10$, $n = 10$ and the greyscale secret image of size 2520×2520 (see online version for colours)



In Figure 3, we compare Sun et al. (2022) and the proposed schemes corresponding to the execution time for updating a share corresponding to new threshold $t \in [3, 10]$ with $k = 2$, $h = 10$, and $n = 10$. For $t \in [3, 9]$, we observe that the execution time corresponding to the proposed scheme is lesser than that corresponding to Sun et al. (2022) scheme. Note that for $t = 10$, the execution time corresponding to the proposed and Sun et al. (2022) schemes visually look approximately equal. However, the execution time in centiseconds corresponding to the proposed and Sun et al. (2022) schemes are 28.4041 and 26.1603, respectively. Hence, the execution time corresponding to the proposed scheme is more than that corresponding to Sun et al. (2022) scheme. These results accord with the implications of

theoretical comparison in Section 5. Furthermore, for the proposed scheme, we observe that the execution time decreases with an increase in t . The reason is as follows. We know that the total number of basic operations in updating a share corresponding to t , where $k < t \leq h$, is $G(k + h - t)/(k(h - 1))$ (from Table 6). Because the value of the expression $k + h - t$ decreases for increase in t , keeping G , k , and h fixed, the number of basic operations also decreases. Because of similar reasons, the execution time for updating a share using the proposed scheme increases by individually increasing h and G , and by individually decreasing k , keeping the other variables fixed.

Figure 3 Comparison of execution time for updating a share corresponding to new threshold $t \in [3, 10]$ with $k = 2$, $h = 10$, $n = 10$ and the greyscale secret image of size 2520×2520 (see online version for colours)



In Figure 4, we compare Sun et al. (2022) and the proposed schemes corresponding to the execution time for reconstruction of the secret image corresponding to threshold $t \in [2, 10]$ with $k = 2$, $h = 10$, and $n = 10$. For $t \in [2, 9]$, we observe that the execution time corresponding to the proposed scheme is more than that corresponding to Sun et al. (2022) scheme. Furthermore, for $t = 10$, the execution time corresponding to the proposed scheme is approximately equal to that corresponding to Sun et al. (2022) scheme. These results accord with our findings in Section 5. For the proposed scheme, we observe that the execution time decreases with an increase in t . The reason is as follows. We know that the total number of basic operations in reconstructing the secret image corresponding to t , where $k \leq t \leq h$, is $(G(k - 1)(h - t)/(h - 1)) + G(h - 1) + (Gk(h - t)/(h - 1)) + Gh$ (from Table 7). Because the value of the expression $h - t$ decreases for an increase in t , keeping G , k , and h fixed, the number of basic operations also decreases. Due to similar reasons, the execution time for reconstructing the secret image using the proposed scheme increases by individually increasing h , k , and G , keeping the other variables fixed.

In Figure 5, we show the initial and updated shares generated using Sun et al. (2022) scheme under $GF(2^8)$ with $k = 2$, $h = 4$, and $n = 4$. We observe visual information leakage of the secret image via each initial and updated share except in S_4^3 and S_4^4 . Furthermore, in Figure 6, we show the initial and updated shares generated using Liu et al. (2019) scheme with $k = 3$, $k_1 = 2$, $k_2 = 4$, and $n = 4$. We observe visual information leakage of the secret image via each initial and updated share. Hence, as mentioned in Section 2, a participant can reveal the secret image using a share generated using Sun et al. (2022) and Liu et al. (2019) schemes.

Figure 4 Comparison of execution time for reconstruction of the secret image corresponding to threshold $t \in [2, 10]$ with $k = 2$, $h = 10$, $n = 10$ and the greyscale secret image of size 2520×2520 (see online version for colours)

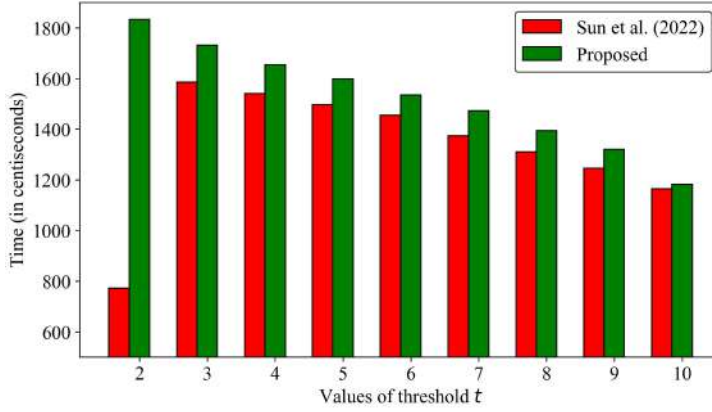


Figure 5 Initial and updated shares generated using Sun et al. (2022) scheme under $GF(2^8)$ with $k = 2$, $h = 4$, and $n = 4$

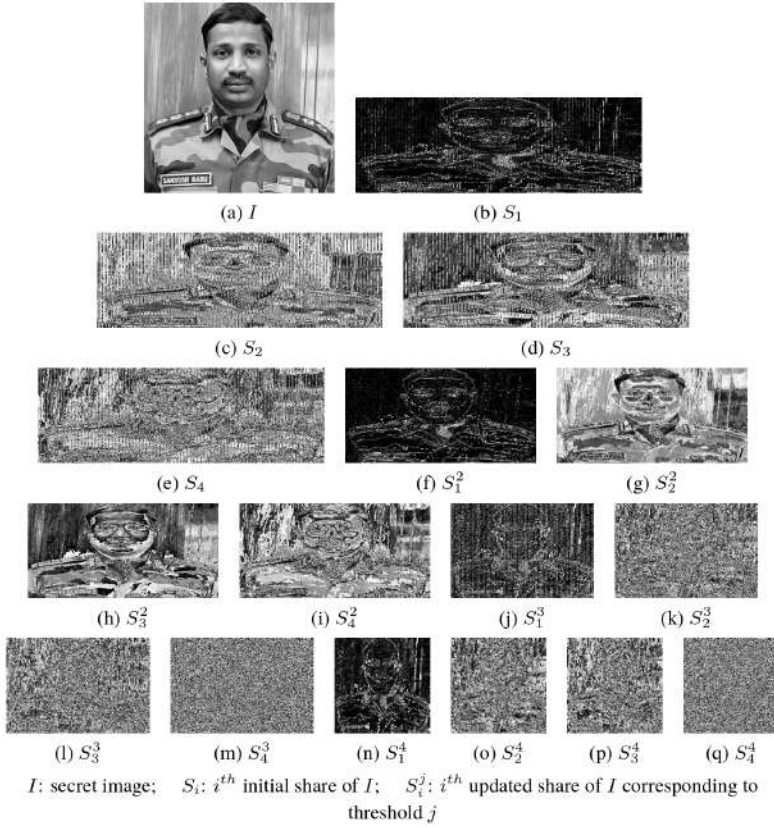
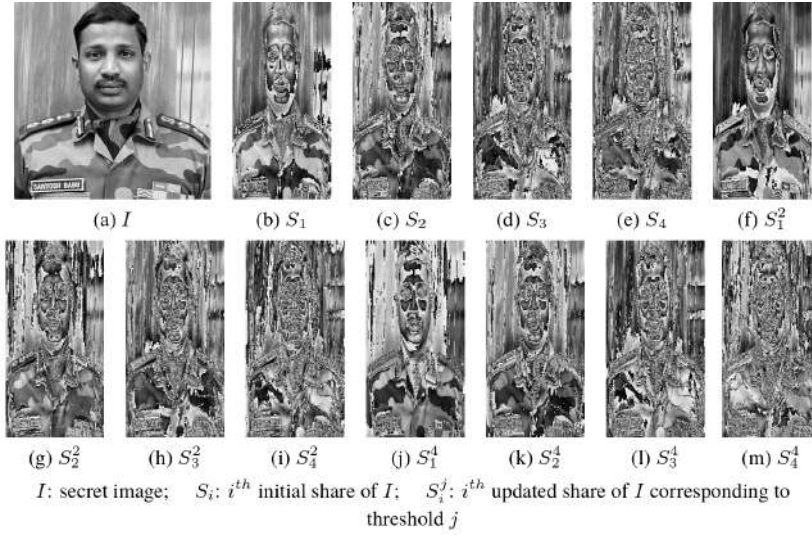


Figure 6 Initial and updated shares generated using Liu et al. (2019) scheme with $k = 3$, $k_1 = 2$, $k_2 = 4$, and $n = 4$



In Figure 7, we show the initial and updated shares generated using the proposed scheme with $k = 2$, $h = 4$, and $n = 4$. We observe that each initial and updated share visually looks random and reveals no information about the secret image. These results accord with Theorems 1 and 2.

Figure 7 Initial and updated shares generated using the proposed scheme with $k = 2$, $h = 4$, and $n = 4$

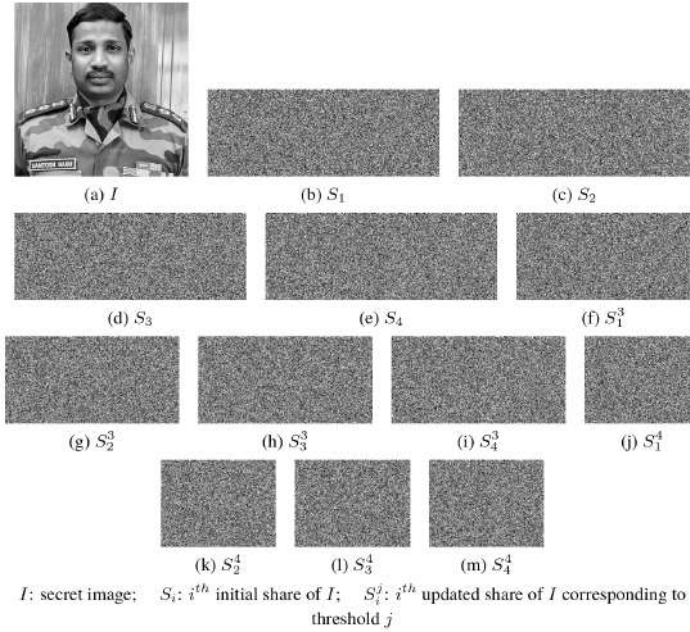


Table 8 Comparison of CC values of adjacent pixels along horizontal direction in individual share shown in Figures 5–7

<i>Share</i>	<i>Liu et al.</i> (2019)	<i>Sun et al.</i> (2022)	<i>Proposed</i>	<i>Share</i>	<i>Liu et al.</i> (2019)	<i>Sun et al.</i> (2022)	<i>Proposed</i>
S_1	0.5292	0.3704	−0.0042	S_1^3	–	0.0212	−0.0003
S_2	0.4400	0.3475	−0.0011	S_2^3	–	0.1191	−0.0001
S_3	0.3701	0.4171	−0.0039	S_3^3	–	0.1430	−0.0062
S_4	0.3074	0.2360	0.0008	S_4^3	–	0.0135	0.0014
S_1^2	0.6350	0.4608	–	S_1^4	0.5427	0.3252	0.0110
S_2^2	0.4081	0.6519	–	S_2^4	0.4380	0.2253	−0.0028
S_3^2	0.3735	0.6592	–	S_3^4	0.3758	0.2469	−0.0051
S_4^2	0.3058	0.4470	–	S_4^4	0.2696	0.0168	0.0070

Table 9 Comparison of CC values of adjacent pixels along vertical direction in individual share shown in Figures 5–7

<i>Share</i>	<i>Liu et al.</i> (2019)	<i>Sun et al.</i> (2022)	<i>Proposed</i>	<i>Share</i>	<i>Liu et al.</i> (2019)	<i>Sun et al.</i> (2022)	<i>Proposed</i>
S_1	0.6866	0.3036	0.0013	S_1^3	–	0.3780	0.0013
S_2	0.6401	0.4863	0.0021	S_2^3	–	0.1672	0.0024
S_3	0.5672	0.4681	0.0023	S_3^3	–	0.1901	0.0016
S_4	0.5226	0.3251	0.0038	S_4^3	–	0.0022	0.0057
S_1^2	0.7502	0.2300	–	S_1^4	0.7346	0.3820	0.0029
S_2^2	0.6761	0.6162	–	S_2^4	0.6374	0.2453	0.0006
S_3^2	0.5852	0.5588	–	S_3^4	0.5887	0.2824	0.0049
S_4^2	0.5519	0.4711	–	S_4^4	0.5397	0.0085	0.0051

Table 10 Comparison of CC values of adjacent pixels along diagonal direction in individual share shown in Figures 5–7

<i>Share</i>	<i>Liu et al.</i> (2019)	<i>Sun et al.</i> (2022)	<i>Proposed</i>	<i>Share</i>	<i>Liu et al.</i> (2019)	<i>Sun et al.</i> (2022)	<i>Proposed</i>
S_1	0.4881	0.2251	0.0025	S_1^3	–	−0.0094	0.0005
S_2	0.3978	0.2496	0.0011	S_2^3	–	0.0631	0.0048
S_3	0.3325	0.2917	0.0013	S_3^3	–	0.0676	0.0041
S_4	0.2758	0.1656	0.0001	S_4^3	–	0.0022	0.0058
S_1^2	0.5979	0.2282	–	S_1^4	0.5147	0.2335	−0.0037
S_2^2	0.3799	0.5159	–	S_2^4	0.3984	0.1424	0.0025
S_3^2	0.3368	0.4921	–	S_3^4	0.3470	0.1396	0.0078
S_4^2	0.2732	0.3219	–	S_4^4	0.2481	0.0041	0.0077

We use CC to measure the correlation between two sequences of numbers (Yan et al., 2019). If two sequences of numbers are correlated, then their CC is either 1 or −1 depending on the nature of their relation. If they are uncorrelated, then their CC is approximately 0. In Tables 8–10, we compare the CC values of adjacent pixels along horizontal, vertical,

and diagonal directions in individual share shown in Figures 5–7, respectively. Note that in Liu et al. (2019) and the proposed schemes, there is no share update corresponding to the initial threshold as the final threshold. Hence, we fill the respective entries with ‘-’. We observe that CC values corresponding to the proposed scheme are comparatively lesser (except for S_4^3 and S_4^4) and close to 0. These results indicate that each initial and updated share generated using the proposed scheme is a random image and hides the correlation among adjacent pixels in the secret image.

7 Conclusion

We have proposed a TCSIS scheme based on univariate and asymmetric bivariate polynomials. The proposed scheme supports $h - k + 1$ consecutive thresholds $k, k + 1, \dots, h$ with k as the initial threshold. Furthermore, the proposed scheme supports the increase in threshold from k to $t \in [k + 1, h]$ by updating each share corresponding to t . We have proved the following three properties of the proposed scheme, which indicates that the proposed scheme overcomes the three corresponding demerits existing in the related schemes. First, each initial and updated share generated using the proposed scheme is a random image revealing no visual information about the secret image. Second, the number of colluding participants less than the current threshold cannot reconstruct the secret image before the end of the secret image’s lifespan. Third, an external adversary cannot reconstruct the secret image before the end of the secret image’s lifespan, using the captured initial and/or updated shares whose total count is less than the current threshold. We have compared the size of each initial and updated share, asymptotic time complexities, and the number of basic operations in the three phases of the proposed scheme with that of the state-of-the-art scheme. We have found that the size of each initial share and the execution time for updating a share (except for threshold h) is comparatively lesser in the proposed scheme. The CC values obtained via experimentation infer that each initial and updated share generated using the proposed scheme is random and reveals no information about the secret image. Furthermore, the CC values corresponding to the proposed scheme are lesser than those corresponding to the related schemes. Hence, the proposed scheme is suitable for usage against the increasing capabilities of an external adversary and the increasing number of colluding participants in the semi-honest model.

The proposed scheme supports the increase in the threshold once from k to $t \in [k + 1, h]$. The assumption is that an external adversary does not have the capability to capture t shares, and the number of colluding participants does not increase to t , before the end of the secret image’s lifespan. However, in practice, these assumptions may fail leading to the reconstruction of the secret image by an external adversary and colluding participants. Hence, future work can focus on increasing the threshold multiple times to overcome this problem. Furthermore, the execution time for generating the initial shares and for reconstructing the secret image (except for threshold h) in the proposed scheme is more than that in the state-of-the-art scheme. Note that the dealer holds the secret image until the generation of initial shares. If the time taken for generating initial shares is more, then an external adversary can capture the secret image directly from the dealer in the real world. If the time taken for reconstructing the secret image is more, a critical action that a combiner takes using the reconstructed secret image can be delayed. Thus, future work can also focus on reducing the time taken to generate initial shares and reconstruct the secret image using linear-time operations by keeping the merits intact.

References

- Bhat, K., Jinwala, D., Prasad, Y. and Zaveri, M. (2023) 'Security analysis of visual based share authentication and algorithms for invalid shares generation in malicious model', *Proceedings of the 2023 5th International Conference on Image Processing and Machine Vision*, Macau, China, pp.86–92.
- Bhat, K., Reddy, K.R., Kumar, U., Kumar HS, R. and Mahto, D. (2021) 'A novel scheme for lossless authenticated multiple secret images sharing using polynomials and extended visual cryptography', *IET Information Security*, Vol. 15, No. 1, pp.13–22.
- Bhat, K. (2021) 'A verifiable lossless multiple secret images sharing scheme', *Information Systems Security. ICISS 2021. Lecture Notes in Computer Science*, Springer, Patna, India, Vol. 13146, pp.157–172.
- Ding, J., Ke, P., Lin, C. and Wang, H. (2022) 'Bivariate polynomial-based secret sharing schemes with secure secret reconstruction', *Information Sciences*, Vol. 593, 398–414.
- Ding, J., Lin, C., Lin, F. and Wang, H. (2023) 'Full threshold change range of threshold changeable secret sharing', *Designs, Codes and Cryptography*, Vol. 91, No. 7, pp.2421–2447.
- Gong, Q., Wang, Y., Yan, X. and Liu, L. (2019) 'Efficient and lossless polynomial-based secret image sharing for color images', *IEEE Access*, Vol. 7, 113216–113222.
- Harn, L. and Hsu, C-F. (2015) 'Dynamic threshold secret reconstruction and its application to the threshold cryptography', *Information Processing Letters*, Vol. 115, No. 11, pp.851–857.
- Harn, L., Hsu, C. and Xia, Z. (2022) 'A novel threshold changeable secret sharing scheme', *Frontiers of Computer Science*, Vol. 16, No. 1, pp.1–7.
- Harn, L., Hsu, C-F., Xia, Z. and Zhou, J. (2017) 'How to share secret efficiently over networks', *Security and Communication Networks*, Vol. 2017, Article ID 5437403, pp.1–6.
- Harn, L., Hsu, C., Xia, Z., Xu, H., Zeng, S. and Pang, F. (2023) 'Simple and efficient threshold changeable secret sharing', *Journal of Information Security and Applications*, Vol. 77, p.103576.
- Hilmi, A., Mezroui, S. and Oualkadi, A.E. (2023) 'A secret image sharing based on logistic-chebyshev chaotic map and Chinese remainder theorem.', *Computer Science Journal of Moldova*, Vol. 31, No. 1, pp.86–110.
- Hu, W., Wu, T., Chen, Y., Shen, Y. and Yuan, L. (2021) 'A lossless secret image sharing scheme using a larger finite field', *Multimedia Tools and Applications*, Vol. 80, No. 19, pp.28731–28743.
- Hua, Z., Wang, Y., Yi, S., Zhou, Y. and Jia, X. (2022) 'Reversible data hiding in encrypted images using cipher-feedback secret sharing', *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 32, No. 8, pp.4968–4982.
- Islam, N., Yuan, L., Li, M., Guo, C., Hu, W. and Luo, X. (2016) 'Secret image sharing scheme with threshold changeable capability', *Mathematical Problems in Engineering*, Vol. 2016, Article ID 9576074, pp.1–11.
- Jia, X., Wang, D., Nie, D., Luo, X. and Sun, J.Z. (2019) 'A new threshold changeable secret sharing scheme based on the chinese remainder theorem', *Information Sciences*, Vol. 473, pp.13–30.
- Li, L., Lu, Y., Yan, X., Liu, L. and Tan, L. (2019) 'Lossless (k, n) -threshold image secret sharing based on the Chinese remainder theorem without auxiliary encryption', *IEEE Access*, Vol. 7, pp.75113–75121.
- Li, W., Peng, C., Tan, W., Xu, Y. and Niu, K. (2023) 'A reversible and lossless secret image sharing scheme with authentication for color images', *Journal of King Saud University-Computer and Information Sciences*, Vol. 35, No. 10, pp.101854.
- Liu, Y-X., Yang, C-N., Wu, C-M., Sun, Q-D. and Bi, W. (2019) 'Threshold changeable secret image sharing scheme based on interpolation polynomial', *Multimedia Tools and Applications*, Vol. 78, No. 13, pp.18653–18667.

- Luo, S., Liu, Y., Yan, X. and Sun, Y. (2024) 'Robust secret image sharing scheme with improved anti-noise capability', *Signal Processing*, Vol. 219, p.109423.
- Luo, S., Liu, Y., Yan, X. and Yu, Y. (2023) 'Secret image sharing scheme with lossless recovery and high efficiency', *Signal Processing*, Vol. 206, p.108931.
- Mashhadi, S. and Saeedi, Z. (2023) 'A (t, n) -secret image sharing with steganography based on rook polynomial and lwe problem', *Multimedia Tools and Applications*, Vol. 82, pp.39077–39097.
- Meng, K., Miao, F., Huang, W. and Xiong, Y. (2020) 'Threshold changeable secret sharing with secure secret reconstruction', *Information Processing Letters*, Vol. 157, Article ID 105928, pp.1–10.
- Meng, K., Miao, F., Xiong, Y. and Chang, C-C. (2021) 'A reversible extended secret image sharing scheme based on Chinese remainder theorem', *Signal Processing: Image Communication*, Vol. 95, pp.116221.
- Pande, D., Rawat, A.S., Deshmukh, M. and Singh, M. (2023) 'Single secret sharing scheme using chinese remainder theorem, modified ShamirTM scheme and xor operation', *Wireless Personal Communications*, Vol. 130, No. 2, pp.957–985.
- Patil, S.M. and B.R., P. (2021) 'Pixel co-ordinate-based secret image sharing scheme with constant size shadow images', *Computers and Electrical Engineering*.
- Paul, A., Kandar, S. and Dhara, B.C. (2022) 'Boolean operation based lossless threshold secret image sharing', *Multimedia Tools and Applications*, Vol. 81, No. 24, pp.35293–35316.
- Prashanti, G. and Bhat, M.N. (2024) 'Cheating identifiable polynomial based secret sharing scheme for audio and image', *Multimedia Tools and Applications*, Vol. 83, No. 1, pp.403–423.
- Ren, Z., Li, P. and Wang, X. (2022) 'An implementation of image secret sharing scheme based on matrix operations', *Mathematics*, Vol. 10, No. 6, pp.1–18.
- Saha, S., Chattopadhyay, A.K., Barman, A.K., Nag, A. and Nandi, S. (2023) 'Secret image sharing schemes: A comprehensive survey', *IEEE Access*, Vol. 11, pp.98333–98361.
- Sardar, M.K. and Adhikari, A. (2020) 'A new lossless secret color image sharing scheme with small shadow size', *Journal of Visual Communication and Image Representation*, Vol. 68, Article ID 102768, pp.1–11.
- Steinfeld, R., Pieprzyk, J. and Wang, H. (2006) 'Lattice-based threshold-changeability for standard crt secret-sharing schemes', *Finite Fields and Their Applications*, Vol. 12, No. 4, pp.653–680.
- Steinfeld, R., Pieprzyk, J. and Wang, H. (2007) 'Lattice-based threshold changeability for standard Shamir secret-sharing schemes', *IEEE Transactions on Information Theory*, Vol. 53, No. 7, pp.2542–2559.
- Sun, Q., Cao, H., Li, S., Song, H. and Liu, Y. (2022) 'Changing the threshold in a bivariate polynomial based secret image sharing scheme', *Mathematics*, Vol. 10, No. 5, pp.1–11.
- Wang, H-K., Xu, G-B. and Jiang, D-H. (2023) 'Quantum grayscale image encryption and secret sharing schemes based on rubik's cube', *Physica A: Statistical Mechanics and its Applications*, Vol. 612, pp.128482.
- Wu, X. and Yao, P. (2023) 'Boolean-based two-in-one secret image sharing by adaptive pixel grouping', *ACM Transactions on Multimedia Computing, Communications and Applications*, Vol. 19, No. 1, pp.1–23.
- Wu, X., Yang, C-N. and Yang, Y-Y. (2020) 'A hybrid scheme for enhancing recovered image quality in polynomial based secret image sharing by modify-and-recalculate strategy', *Journal of Information Security and Applications*, Vol. 51, Article ID 102452, pp.1–13.
- Wu, G., Wang, M., Wang, Q., Yao, Y., Yuan, L. and Miao, G. (2021) 'A novel threshold changeable secret image sharing scheme', *Symmetry*, Vol. 13, No. 2, pp.1–20.
- Wu, B., Xie, D., Chen, F., Wang, X. and Zeng, Y. (2022) 'A multi-party secure encryption-sharing hybrid scheme for image data base on compressed sensing', *Digital Signal Processing*, Vol. 123, pp.103391.

- Wu, X., Xu, Z. and Yan, W. (2023) 'Sharing visual secrets among multiple groups with enhanced performance', *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 33, No. 11, pp.6503–6518.
- Xie, D., Wu, B., Chen, F., Wang, T., Hu, Z. and Zhang, Y. (2023) 'A low-overhead compressed sensing-driven multi-party secret image sharing scheme', *Multimedia Systems*, Vol. 29, pp.1187–1202.
- Xiong, L., Han, X., Yang, C-N. and Xia, Z. (2023a) 'RDH-DES: Reversible data hiding over distributed encrypted-image servers based on secret sharing', *ACM Transactions on Multimedia Computing, Communications and Applications*, Vol. 19, No. 1, pp.1–19.
- Xiong, L., Han, X., Yang, C-N. and Zhang, X. (2023b) 'Reversible data hiding in shared images based on syndrome decoding and homomorphism', *IEEE Transactions on Cloud Computing*, Vol. 11, No. 3, pp.3085–3098.
- Yan, X., Li, L., Chen, J. and Sun, L. (2023) 'Public key based bidirectional shadow image authentication without pixel expansion in image secret sharing', *Frontiers of Information Technology and Electronic Engineering*, Vol. 24, No. 1, pp.88–103.
- Yan, X., Li, J., Lu, Y., Liu, L., Yang, G. and Chen, H. (2020), Relations between secret sharing and secret image sharing', *Security with Intelligent Computing and Big-data Services*, Springer International Publishing, Cham, pp.79–93.
- Yan, X., Liu, L., Li, L. and Lu, Y. (2021) 'Robust secret image sharing resistant to noise in shares', *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, Vol. 17, No. 1, pp.1–22.
- Yan, X., Liu, L., Lu, Y. and Gong, Q. (2019) 'Security analysis and classification of image secret sharing', *Journal of Information Security and Applications*, Vol. 47, pp.208–216.
- Yang, C-N., Wu, X., Chung, M-J. and Zhang, X. (2022) 'Ambtc-based secret image sharing by simple modular arithmetic', *Journal of Visual Communication and Image Representation*, Vol. 84, Article ID 103482, pp.1–13.
- Yang, C-N., Li, P. and Kuo, H-C. (2023a) '(k, n) secret image sharing scheme with privileged set', *Journal of Information Security and Applications*, Vol. 73, pp.103413.
- Yang, C-N., Zheng, C-E., Lu, M-C. and Wu, X. (2023b) 'Secret image sharing by using multi-prime modular arithmetic', *Signal Processing*, Vol. 205, pp.108882.
- Ye, C. and Chen, C. (2023) 'Secure medical image sharing for smart healthcare system based on cellular neural network', *Complex and Intelligent Systems*, Vol. 9, No. 2, pp.1653–1670.
- Yu, L., Liu, L., Xia, Z., Yan, X. and Lu, Y. (2020) 'Lossless and efficient secret image sharing based on matrix theory modulo 256', *Mathematics*, Vol. 8, No. 6, pp.1–17.
- Yuan, L., Li, M., Guo, C., Choo, K-K.R. and Ren, Y. (2016) 'Novel threshold changeable secret sharing schemes based on polynomial interpolation', *PloS one*, Vol. 11, No. 10, pp.1–19.
- Zhang, Z., Chee, Y.M., Ling, S., Liu, M. and Wang, H. (2012) 'Threshold changeable secret sharing schemes revisited', *Theoretical Computer Science*, Vol. 418, 106–115.
- Zhang, L., Zhang, J., Sun, J. and Chen, Q. (2024) 'A global progressive image secret sharing scheme under multi-group joint management', *Mathematical Biosciences and Engineering*, Vol. 21, No. 1, pp.1286–1304.