

Article

A Low-Complexity Real-Time Video Streaming Encryption Algorithm for Resource-Constrained LEO Satellites

Wenyu Xu ¹, Xiaoyuan Yang ^{2,3,*} and Nanhao Liang ^{2,3,*}¹ School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, China; wenyux@bit.edu.cn² Hefei Institutes of Physical Science, Chinese Academy of Sciences, Hefei 230031, China³ Science Island Branch, Graduate School of USTC, Hefei 230026, China

* Correspondence: yxy22@mail.ustc.edu.cn (X.Y.); lnh@mail.ustc.edu.cn (N.L.)

Abstract

Low Earth orbit (LEO) satellites are increasingly required to process and securely stream video data for remote sensing, surveillance, and onboard perception applications. However, the strict constraints of onboard computing capability, power budget, and thermal dissipation make conventional encryption schemes difficult to apply to real-time video streaming tasks. To address this challenge, this paper proposes a low-complexity real-time video encryption algorithm for resource-constrained LEO satellites. The proposed method integrates selective encryption with a lightweight permutation–diffusion mechanism to reduce computational overhead while maintaining effective protection of continuous video streams. To enhance security, a chaotic pseudo-random sequence generator is employed to improve encryption randomness, and a dynamic key scheduling strategy is introduced to increase temporal key variability and strengthen resistance to statistical and differential attacks across successive frames. The algorithm is further designed for efficient deployment on embedded onboard platforms with limited hardware resources. Experimental results show that the proposed method achieves favorable performance in encryption speed, computational complexity, information entropy, adjacent pixel correlation, and differential attack resistance. Compared with conventional full-encryption methods, the proposed algorithm offers a more balanced trade-off between security and real-time efficiency, demonstrating its potential for secure video streaming in resource-constrained LEO satellite systems.

Keywords: LEO satellites; real-time video encryption; low-complexity algorithm

Academic Editor: Hong Wen

Received: 15 June 2026

Revised: 1 July 2026

Accepted: 7 July 2026

Published: 7 July 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Low Earth orbit (LEO) satellites are increasingly deployed in Earth observation, environmental monitoring, disaster response, border surveillance, and intelligent sensing applications [1–3]. With the rapid development of onboard imaging sensors and edge computing capabilities, satellites are now able to capture, process, and securely transmit high-resolution video streams in near real time [4,5]. These video streams often contain sensitive visual information related to strategic infrastructures, geographic activities, and privacy-sensitive targets [6]. Therefore, ensuring secure and efficient protection of onboard video streams has become a critical requirement in modern satellite systems.

However, secure real-time video streaming in LEO satellites faces substantial practical challenges [7]. Unlike terrestrial cloud or ground-based platforms, satellite-borne systems

are severely constrained by computational capability, power budget, memory availability, and thermal dissipation conditions [8,9]. Under such constraints, conventional encryption schemes, although cryptographically strong, may introduce considerable computational overhead and latency when directly applied to continuous high-rate video streams. Frame-wise or pixel-wise full encryption may therefore be impractical for real-time onboard applications, especially when satellites are expected to simultaneously support perception, compression, target analysis, and communication tasks [10]. As a result, there is a clear need for video encryption methods that can provide effective security while maintaining low complexity and real-time performance.

To address this issue, lightweight and selective encryption methods have attracted increasing attention in multimedia security [11]. Compared with full encryption approaches, selective encryption reduces the amount of processed data by protecting only the most informative parts of an image or video stream, thereby improving computational efficiency and reducing latency. In parallel, chaos-based encryption techniques have been widely studied because of their sensitivity to initial conditions, pseudo-random behavior, and suitability for constructing lightweight permutation and diffusion operations [12]. Nevertheless, many existing chaos-based image and video encryption methods are primarily designed for general computing environments and do not explicitly consider the strict resource constraints of LEO satellites [13,14]. Some methods rely on multi-round iterative structures, while others lack an effective frame-level key update mechanism, which limits their applicability to real-time onboard deployment in resource-constrained LEO satellites and weakens their resistance to statistical or differential attacks in continuous video streaming scenarios [15].

Motivated by these challenges, this paper proposes a low-complexity real-time video encryption algorithm for secure streaming in resource-constrained LEO satellites. The proposed method combines selective encryption with a lightweight permutation–diffusion structure to reduce the computational burden of securing continuous onboard video streams. A chaotic pseudo-random sequence generation mechanism is introduced to enhance encryption randomness, while a dynamic key scheduling strategy is employed to improve temporal key variability across successive frames. In this way, the proposed algorithm aims to achieve a practical balance among security, efficiency, and deployability for secure onboard video streaming. The overall design is intended to be suitable for embedded satellite platforms where low overhead and real-time execution are essential.

The main contributions of this paper are summarized as follows:

1. A low-complexity real-time video encryption framework is developed for secure video streaming in resource-constrained LEO satellite scenarios, where conventional full-encryption methods are often too costly for onboard deployment.
2. A selective encryption strategy is integrated with a lightweight permutation–diffusion mechanism to reduce computational overhead while preserving effective visual protection for continuous video streams.
3. A chaotic pseudo-random sequence generator and a dynamic key scheduling scheme are introduced to strengthen security and improve resistance to statistical and differential attacks across successive video frames.
4. Extensive experiments are conducted to evaluate encryption efficiency and security performance from multiple perspectives, including encryption speed, computational complexity, information entropy, pixel correlation, and differential attack resistance.

In contrast to prior selective encryption work that picks CUs by fixed syntax patterns or random sampling, our method scores each CU on texture, motion, and coding cost jointly. Unlike chaotic schemes that reuse a static seed or a frame counter for key updates, we derive frame-specific keys from SHA-256 with an irrational perturbation term so the

sequence stays non-periodic. The framework also gives explicit per-frame complexity bounds, which most earlier studies do not provide.

The remainder of this paper is organized as follows. Section 2 reviews related work on selective encryption for video data, chaos-based video encryption, and secure video streaming in LEO satellite systems. Section 3 formulates the problem and presents the scenario description, threat model, and design objectives. Section 4 describes the proposed low-complexity real-time video encryption algorithm in detail. Section 5 presents the experimental results and performance analysis. Finally, Section 6 concludes the paper and outlines future research directions.

2. Related Work

2.1. Selective Encryption for Video Data

Selective encryption has been widely investigated as a practical alternative to full video encryption, since it can reduce computational overhead while preserving format compliance and real-time performance in compressed video systems. Early High Efficiency Video Coding (HEVC)-oriented studies mainly protected fixed syntax elements in the bitstream, whereas later works gradually introduced random selection and content-aware adaptation to improve flexibility and efficiency. For example, Sheng et al. [16] proposed a tunable HEVC selective encryption scheme with semantic understanding, where encryption strength is adjusted according to content sensitivity. Chen et al. [17] further improved flexibility by introducing randomly selective encryption for HEVC bitstreams, in which pseudorandom extraction positions are used to enlarge the encryption space with low time complexity.

To avoid unnecessary encryption operations, Sheng et al. [18] proposed a content-aware H.265/HEVC selective encryption scheme that encrypts only sensitive groups of pictures (GOPs). They further extended this idea to tunable content-aware protection, where different GOPs are encrypted with different strengths to better balance security, bitrate overhead, and efficiency [19]. In privacy-sensitive surveillance scenarios, selective encryption has also been combined with region-of-interest (ROI) protection. Cho et al. [20] designed an ROI-based encryption system with selective decryption, while Aribilola et al. [21] developed SecureCam to integrate selective detection and lightweight encryption for dynamic surveillance videos. Overall, existing studies show that selective encryption has evolved from fixed syntax-level protection to adaptive and content-aware schemes, but balancing security, coding efficiency, and real-time applicability remains an open challenge.

It is worth noting that a separate line of research has developed lightweight general-purpose block ciphers such as SPECK, SIMON [22], and PRESENT [23] for resource-constrained embedded systems; unlike the proposed method, these ciphers operate at the bit/byte level without video content awareness or perceptual importance modeling, and are therefore complementary rather than directly comparable approaches.

2.2. Video Streaming on LEO Satellites

LEO satellite networks offer global broadband coverage for video streaming, yet their frequent satellite handovers, dynamic latency, and asymmetric bandwidth severely degrade streaming stability and Quality of Experience (QoE) [24–27]. Global measurements on Starlink demonstrate that LEO delivers comparable perceptual video quality to terrestrial networks in covered regions, yet induces 60% more bitrate switches and double rebuffering events due to fluctuating throughput and periodic link handovers every 15 s [28]. Such periodic handovers cause abrupt throughput drops and latency surges, which standard adaptive bitrate (ABR) algorithms fail to cope with [29]. Existing improvements include

handover-aware ABR schemes, combinatorial multi-armed bandit (CMAB)-based low-latency live adaptation, and MulTCP-augmented congestion control, yet they remain within the conventional transmission framework and cannot overcome LEO's bandwidth and latency bottlenecks [30]. Recently, edge computing and generative AI have brought new solutions. Multi-access edge computing (MEC)-assisted super-resolution streaming transmits low-bitrate streams via satellites and enhances quality at edge nodes, cutting backhaul bandwidth usage while boosting QoE [31]; space-native generative models are further deployed on satellites to compensate for handover interruptions and packet loss via lightweight content reconstruction [32]. Additionally, satellite-unmanned aerial vehicle (UAV)-terrestrial integrated systems use edge caching and hybrid reinforcement learning to optimize relay scheduling for dynamic topology adaptation [33]. However, these AI-enhanced methods rely heavily on constrained on-board computing resources, and few designs achieve lightweight, real-time adaptation for resource-limited LEO platforms [34].

3. Problem Formulation

3.1. Scenario Description

This work considers a secure real-time video streaming scenario in a resource-constrained LEO satellite system. Let $V = \{F_t\}_{t=1}^T$ denote the onboard captured video stream, where F_t is the t -th video frame and T is the total number of frames. Each frame is represented as a set of coding units (CUs), denoted by $\{U_{t,i}\}_{i=1}^{N_t}$, where $U_{t,i}$ is the i -th coding unit (CU) in frame t and N_t is the total number of coding units in the current frame. Instead of encrypting the whole frame, the proposed method selectively protects only the sensitive coding units, thereby reducing computational overhead and transmission-side processing delay. Let $\mathcal{S}_t = \{U_{t,i} \mid i \in \Omega_t\}$ denote the selected coding unit set in frame t , where Ω_t is the index set of the selected coding units, and let K_t denote the corresponding frame-dependent encryption key.

Based on the above definitions, the encryption process for the t -th frame is modeled as

$$\hat{F}_t = \mathcal{E}(F_t, K_t), \quad t = 1, 2, \dots, T, \quad (1)$$

where $\mathcal{E}(\cdot)$ denotes the proposed selective encryption mapping and $\hat{F}_t$ is the encrypted frame. Accordingly, the encrypted video stream is written as

$$\hat{V} = \{\hat{F}_t\}_{t=1}^T. \quad (2)$$

At the authorized receiver, the original frame is recovered through the inverse mapping

$$F_t = \mathcal{D}(\hat{F}_t, K_t), \quad t = 1, 2, \dots, T, \quad (3)$$

where $\mathcal{D}(\cdot)$ denotes the corresponding decryption operation. Therefore, the considered formulation characterizes a secure and reversible frame-level selective encryption process under the constraints of limited computation, memory, and power resources.

3.2. Threat Model

In the considered LEO satellite video streaming scenario, the communication link between the satellite and the authorized receiver is assumed to be insecure and can be monitored by an adversary [35]. Let $\hat{V} = \{\hat{F}_t\}_{t=1}^T$ denote the transmitted encrypted video stream. The attacker is assumed to have full access to $\hat{V}$ and may perform ciphertext-only statistical analysis, frame correlation analysis, and differential attacks in an attempt to infer visual information or reveal the underlying encryption mechanism. However, the adversary is assumed to have no access to the secret master key, the frame-dependent

key sequence, or the internal parameters used in the chaotic sequence generation and dynamic key scheduling process [36].

The considered threat model is illustrated in Figure 1. In this scenario, the LEO satellite captures onboard video streams and encrypts them before transmission over an insecure satellite–ground communication link. An adversary may intercept the transmitted stream and attempt to infer sensitive visual information through statistical analysis, frame correlation analysis, differential analysis, or known-plaintext attacks. The authorized ground station, which obtains valid key information through a secure key management channel, can recover the original video stream using the corresponding decryption process. The authorized receiver, by contrast, possesses the correct key information and can recover the original video frames through the inverse decryption mapping. Accordingly, the security objective of the proposed method is to ensure that the intercepted ciphertext stream leaks as little visual and statistical information as possible, while maintaining correct and efficient recovery at the legitimate receiver. Under this threat model, the proposed algorithm is expected to resist common statistical and differential attacks while remaining suitable for real-time deployment in resource-constrained onboard environments.

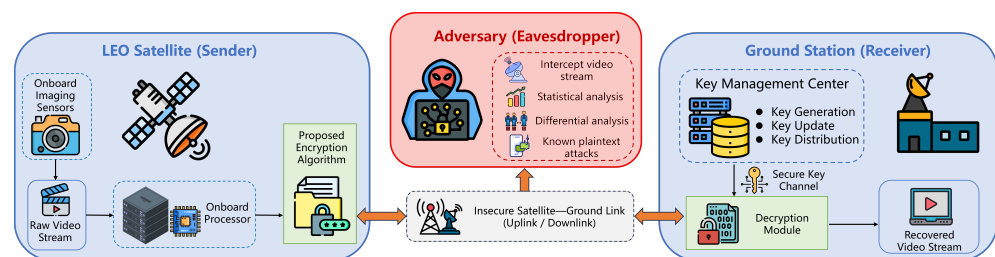


Figure 1. Threat model of secure real-time video streaming in a resource-constrained LEO satellite system.

3.3. Design Objectives

Under the above system and threat models, the proposed encryption method is designed to achieve an effective trade-off between security and real-time efficiency for resource-constrained LEO satellite video applications. First, the algorithm should provide sufficient confidentiality for sensitive visual information, so that an adversary cannot directly infer meaningful scene content from the intercepted ciphertext stream. Second, the algorithm should maintain low computational complexity, low memory overhead, and low processing delay, since onboard satellite platforms are typically limited in processor capability, storage resources, and power budget. Third, the method should preserve correct reversibility at the authorized receiver, ensuring that the original video content can be reliably recovered with the correct key.

In addition, the proposed scheme should exhibit robustness against common statistical and differential attacks by reducing plaintext–ciphertext correlation and enhancing frame-level key variability. From an implementation perspective, the method should remain lightweight enough for real-time deployment while achieving acceptable security performance. Therefore, the main design objectives of the proposed algorithm can be summarized as confidentiality, low complexity, low latency, low memory consumption, and reliable decryptability.

4. Methodology

4.1. Overview of the Proposed Algorithm

The proposed lightweight video encryption algorithm is designed for real-time secure transmission in resource-constrained LEO satellite systems. To reduce computational overhead while preserving sufficient visual protection, the algorithm adopts a selective encryption strategy instead of encrypting the entire video frame. For each input frame,

the video data are first preprocessed and partitioned into blocks. Then, the sensitive regions are identified and selected for protection. The selected data are subsequently processed by a lightweight permutation–diffusion module driven by a chaotic pseudo-random sequence, while a dynamic key scheduling mechanism is employed to update the encryption parameters for successive frames.

As shown in Figure 2, the proposed framework consists of two parts: onboard encryption and ground decryption. In the onboard encryption stage, the input video frame is first divided into coding units, followed by importance-aware CU selection, chaotic key generation, permutation, and diffusion. At the ground station, the same key stream is regenerated to perform inverse diffusion, inverse permutation, and coding-unit remapping for frame recovery.

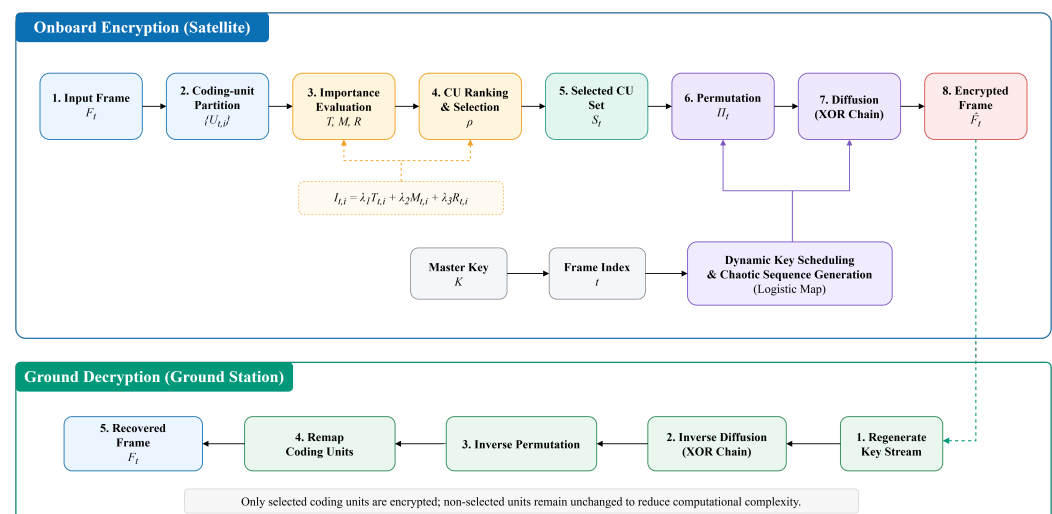


Figure 2. Overall framework of the proposed lightweight selective video encryption algorithm for resource-constrained LEO satellite systems.

4.2. Sensitive Region Selection Based on Coding Units

To improve compatibility with video coding structures and avoid unnecessary manual partitioning, the proposed method performs selective encryption on coding units rather than fixed-size image blocks [37,38]. Let the t -th frame be represented as

$$F_t = \{U_{t,i}\}_{i=1}^{N_t}, \quad (4)$$

where $U_{t,i}$ denotes the i -th coding unit in frame t , and N_t is the total number of coding units. In this work, the coding unit is adopted as the basic selection element for importance evaluation and subsequent encryption, which makes the proposed method more consistent with practical video coding structures.

Since not all coding units contribute equally to the perceptual content of a video frame, directly encrypting all units would lead to unnecessary computational overhead. To address this issue, a lightweight importance-aware model is introduced to identify the most visually significant coding units before encryption. Specifically, the importance of each coding unit is jointly determined by its spatial texture complexity, temporal motion activity, and coding-related significance. Based on these factors, the importance score of $U_{t,i}$ is defined as

$$I_{t,i} = \lambda_1 T_{t,i} + \lambda_2 M_{t,i} + \lambda_3 R_{t,i}, \quad (5)$$

where $T_{t,i}$ is the spatial texture complexity, $M_{t,i}$ is the motion activity between consecutive frames, and $R_{t,i}$ is the coding-related importance, which reflects the energy or bit consumption of the unit during encoding. The weighting coefficients satisfy

$$\lambda_1 + \lambda_2 + \lambda_3 = 1. \quad (6)$$

Among the three components, the texture term is used to characterize the local spatial detail of the coding unit and is computed by the luminance variance as

$$T_{t,i} = \frac{1}{|U_{t,i}|} \sum_{p \in U_{t,i}} (Y_t(p) - \mu_{t,i})^2, \quad (7)$$

where $Y_t(p)$ denotes the luminance value of pixel p , $\mu_{t,i}$ denotes the mean luminance of coding unit $U_{t,i}$, and $|U_{t,i}|$ is the number of pixels contained in the unit. A larger value of $T_{t,i}$ indicates that the unit contains richer spatial details and is therefore more likely to carry visually important information. The temporal activity term is introduced to capture motion variations across adjacent frames and is defined as

$$M_{t,i} = \frac{1}{|U_{t,i}|} \sum_{p \in U_{t,i}} |Y_t(p) - Y_{t-1}(p)|, \quad (8)$$

where $Y_{t-1}(p)$ is the luminance value of the corresponding pixel in the previous frame. A higher value of $M_{t,i}$ means that the current unit undergoes more significant temporal variation and should be preferentially protected. In addition, the coding-related importance is defined as

$$R_{t,i} = B_{t,i}, \quad (9)$$

where $B_{t,i}$ denotes the number of bits associated with coding unit $U_{t,i}$ in the encoded frame. This term is introduced to reflect the contribution of the unit to the coding representation and to further improve the relevance of the selection strategy to practical encoded video streams.

After computing the importance scores of all coding units in the current frame, the units are ranked in descending order according to $I_{t,i}$. The top ρN_t units are then selected for encryption, where $\rho \in (0, 1]$ denotes the selection ratio. The selected set is expressed as

$$\mathcal{S}_t = \{U_{t,i} \mid i \in \Omega_t\}, \quad (10)$$

with

$$|\Omega_t| = \rho N_t. \quad (11)$$

Through this importance-aware selection mechanism, the proposed method is able to focus the encryption process on the most informative coding units, thereby reducing computational cost while maintaining effective protection of critical visual content. For clarity, the detailed procedure of the proposed lightweight importance-aware coding unit selection strategy is summarized in Algorithm 1.

4.3. Chaotic Sequence Generation and Key Initialization

After the sensitive coding units are determined, pseudo-random control sequences are required to drive the subsequent permutation and diffusion operations. To achieve this goal with low computational complexity, the proposed method employs a lightweight chaotic generator to produce frame-dependent encryption sequences. Compared with conventional pseudo-random generators, chaotic systems exhibit strong sensitivity to initial conditions

and control parameters, which makes them suitable for lightweight multimedia encryption in resource-constrained environments.

Algorithm 1 Lightweight Importance-Aware Sensitive Coding Unit Selection

Require: Current frame F_t , previous frame F_{t-1} , coding unit set $\{U_{t,i}\}_{i=1}^{N_t}$, weighting coefficients $\lambda_1, \lambda_2, \lambda_3$, selection ratio ρ

Ensure: Selected coding unit set $\mathcal{S}_t$ and index set Ω_t

```

1: for  $i = 1$  to  $N_t$  do
2:   Compute spatial texture complexity  $T_{t,i}$  using (7)
3:   Compute temporal motion activity  $M_{t,i}$  using (8)
4:   Compute coding-related importance  $R_{t,i}$  using (9)
5: end for
6: Normalize  $\{T_{t,i}\}$ ,  $\{M_{t,i}\}$ , and  $\{R_{t,i}\}$  to a comparable range
7: for  $i = 1$  to  $N_t$  do
8:   Compute  $I_{t,i} \leftarrow \lambda_1 T_{t,i} + \lambda_2 M_{t,i} + \lambda_3 R_{t,i}$ 
9: end for
10: Sort all coding units in descending order according to  $I_{t,i}$ 
11: Set  $N_s \leftarrow \lceil \rho N_t \rceil$ 
12: Select the indices of the top  $N_s$  coding units as  $\Omega_t$ 
13: Construct  $\mathcal{S}_t \leftarrow \{U_{t,i} \mid i \in \Omega_t\}$ 
14: return  $\mathcal{S}_t, \Omega_t$ 

```

Let the secret master key be denoted by K . For the t -th frame, the initial state of the chaotic system is first derived from K and the frame index t , so that different frames are associated with different pseudo-random sequences. Denote the initial state by $x_t^{(0)} \in (0, 1)$. To enhance frame-level key variability, the initial state is dynamically updated as

$$x_t^{(0)} = \text{frac}\left(x_0 + \gamma t + \frac{H(K)}{2^{128}}\right), \quad (12)$$

where $x_0 \in (0, 1)$ is the master initial seed, $\gamma = \sqrt{2} - 1 \approx 0.41421356$ is a fixed irrational perturbation constant (any irrational in a reasonable range works equally well; we use $\sqrt{2} - 1$ for convenience), $H(K)$ denotes the integer value obtained from the secret key K through a hash-based mapping, and $\text{frac}(\cdot)$ extracts the fractional part of its argument. In this way, the initial state changes from frame to frame while remaining deterministically reproducible at the authorized receiver.

Starting from $x_t^{(0)}$, a one-dimensional chaotic map is then iteratively applied as

$$x_t^{(n+1)} = \mu x_t^{(n)} (1 - x_t^{(n)}), \quad n = 0, 1, \dots, L_t - 1, \quad (13)$$

where $\mu \in (\mu_\infty, 4]$ is the control parameter, with μ_∞ denoting the chaos threshold of the logistic map. L_t is the required sequence length for the current frame. The generated chaotic orbit $\{x_t^{(n)}\}_{n=1}^{L_t}$ is further transformed into an integer pseudo-random sequence by

$$z_t^{(n)} = \lfloor x_t^{(n)} \times 10^{14} \rfloor \bmod 256, \quad n = 1, 2, \dots, L_t, \quad (14)$$

where $z_t^{(n)}$ is the quantized chaotic output used for the subsequent encryption process. To support the two-stage encryption procedure, the generated integer sequence is further divided into two control streams, namely the permutation control sequence and the diffusion control sequence. Specifically, the first part of the sequence is used to construct the permutation index, while the remaining part is used to generate the diffusion mask values.

By combining frame-dependent initialization with lightweight chaotic iteration, the proposed method ensures that the subsequent encryption operations remain com-

putationally efficient while preserving sufficient randomness and inter-frame diversity for secure real-time video transmission. The logistic map is used here because each iteration costs only one multiplication and one subtraction—about as cheap as a chaotic map gets, which matters on a power-limited satellite processor. We run it at $\mu = 3.99$, well inside the fully chaotic regime and far from known periodic windows. The frame-dependent hash initialization with irrational perturbation (Equation (12)) breaks any residual cross-frame periodicity.

4.4. Permutation and Diffusion of Selected Coding Units

After the sensitive coding units have been determined and the frame-dependent chaotic sequences have been generated, the proposed method performs encryption on the selected data through two successive stages, namely permutation and diffusion. The permutation stage is introduced to disrupt the positional correlation among the selected coding units, while the diffusion stage is designed to modify their statistical distribution and reduce the correlation between plaintext and ciphertext. By combining these two operations, the proposed scheme enhances resistance against statistical and differential attacks while maintaining relatively low computational complexity.

Let $\mathcal{S}_t = \{U_{t,i} \mid i \in \Omega_t\}$ denote the selected coding unit set in the t -th frame, where $|\Omega_t| = \rho N_t$. For convenience, the pixels extracted from all selected coding units are rearranged into a one-dimensional sequence

$$P_t = \{p_t^{(m)}\}_{m=1}^{L_t}, \quad (15)$$

where L_t denotes the total number of pixels contained in the selected coding units. Based on the permutation control sequence generated in Section 4.3, a permutation index sequence $\Pi_t = \{\pi_t^{(m)}\}_{m=1}^{L_t}$ is obtained by sorting the corresponding chaotic samples in ascending order. The permuted sequence is then expressed as

$$P'_t(m) = P_t(\pi_t^{(m)}), \quad m = 1, 2, \dots, L_t, \quad (16)$$

where P'_t denotes the permuted plaintext sequence. Through this operation, the spatial arrangement of the selected coding-unit data is thoroughly disturbed before value transformation.

After permutation, a diffusion process is further applied to alter the pixel values of the reordered sequence. Let $Z_t^{(d)} = \{z_t^{(m)}\}_{m=1}^{L_t}$ denote the diffusion control sequence derived from the chaotic generator. The ciphertext sequence $C_t = \{c_t^{(m)}\}_{m=1}^{L_t}$ is generated recursively as

$$c_t^{(m)} = P'_t(m) \oplus z_t^{(m)} \oplus c_t^{(m-1)}, \quad m = 1, 2, \dots, L_t, \quad (17)$$

where $\oplus$ denotes the bitwise XOR operation, and $c_t^{(0)}$ is the initial diffusion value. Since the current ciphertext element depends not only on the current plaintext sample but also on the previous ciphertext sample, the diffusion process introduces strong propagation of local changes, thereby improving sensitivity to plaintext variations.

The complete frame-level selective permutation–diffusion encryption procedure is summarized in Algorithm 2. Finally, the encrypted sequence C_t is mapped back to the corresponding positions of the selected coding units, while the non-selected units remain unchanged. In this manner, the proposed selective encryption strategy preserves low processing complexity by operating only on the most important coding units, while the combined permutation–diffusion mechanism provides sufficient confusion and diffusion for secure video protection in resource-constrained onboard environments.

Algorithm 2 Frame-Level Selective Permutation–Diffusion Encryption

Require: Current frame F_t , selected coding unit set $\mathcal{S}_t$, index set Ω_t , master key K , frame index t , control parameter μ

Ensure: Encrypted frame $\hat{F}_t$

- 1: Initialize $\hat{F}_t \leftarrow F_t$
- 2: Derive the frame-dependent initial state $x_t^{(0)}$ from K and t using (12)
- 3: Determine the required chaotic sequence length L_t according to $\mathcal{S}_t$
- 4: **for** $n = 0$ to $L_t - 1$ **do**
- 5: Generate chaotic sample $x_t^{(n+1)} \leftarrow \mu x_t^{(n)} (1 - x_t^{(n)})$
- 6: Quantize $x_t^{(n+1)}$ into integer value $z_t^{(n+1)}$ using (14)
- 7: **end for**
- 8: Split the generated sequence into permutation sequence $Z_t^{(p)}$ and diffusion sequence $Z_t^{(d)}$
- 9: Extract pixels from selected coding units $\mathcal{S}_t$ and form plaintext sequence P_t
- 10: Generate permutation index Π_t by sorting $Z_t^{(p)}$ in ascending order
- 11: **for** $m = 1$ to L_t **do**
- 12: Apply permutation: $P'_t(m) \leftarrow P_t(\pi_t^{(m)})$
- 13: **end for**
- 14: Set initial diffusion value $c_t^{(0)}$
- 15: **for** $m = 1$ to L_t **do**
- 16: Apply diffusion: $c_t^{(m)} \leftarrow P'_t(m) \oplus z_t^{(m)} \oplus c_t^{(m-1)}$
- 17: **end for**
- 18: Map ciphertext sequence $C_t = \{c_t^{(m)}\}_{m=1}^{L_t}$ back to the selected coding units indexed by Ω_t
- 19: Keep non-selected coding units unchanged
- 20: **return** $\hat{F}_t$

4.5. Decryption Process

At the authorized receiver, the decryption procedure is carried out by applying the inverse operations of the encryption stage to the selected coding units. Since the proposed method employs frame-dependent chaotic initialization, the receiver first regenerates the same pseudo-random control sequences from the shared secret key and the frame index. Based on these sequences, the encrypted data are processed through inverse diffusion and inverse permutation in reverse order, thereby ensuring exact recovery of the protected coding units.

Let $C_t = \{c_t^{(m)}\}_{m=1}^{L_t}$ denote the ciphertext sequence corresponding to the selected coding units in frame t , where L_t is the total number of pixels contained in the selected set $\mathcal{S}_t$. According to the diffusion rule in Section 4.4, the original permuted sequence $P'_t = \{P'_t(m)\}_{m=1}^{L_t}$ can be recovered by

$$P'_t(m) = c_t^{(m)} \oplus z_t^{(m)} \oplus c_t^{(m-1)}, \quad m = 1, 2, \dots, L_t, \quad (18)$$

where $z_t^{(m)}$ is the diffusion control sequence regenerated from the same chaotic system, and $c_t^{(0)}$ is the initial diffusion value shared by both encryption and decryption processes. After inverse diffusion, the inverse permutation operation is performed according to the permutation index sequence $\Pi_t = \{\pi_t^{(m)}\}_{m=1}^{L_t}$, yielding

$$P_t(\pi_t^{(m)}) = P'_t(m), \quad m = 1, 2, \dots, L_t, \quad (19)$$

where $P_t = \{p_t^{(m)}\}_{m=1}^{L_t}$ denotes the recovered plaintext sequence associated with the selected coding units.

Finally, the restored sequence P_t is remapped to the corresponding positions of the selected coding units in the current frame, while the non-selected units remain unchanged throughout the entire process. In this way, the original frame content can be losslessly reconstructed at the authorized receiver, provided that the correct secret key and frame-dependent initialization parameters are available. Therefore, the proposed selective encryption scheme preserves strict reversibility while maintaining low computational overhead, which is essential for practical secure video transmission in resource-constrained LEO satellite systems.

4.6. Computational Complexity Analysis

The computational complexity of the proposed method mainly arises from sensitive coding unit selection, chaotic sequence generation, and permutation–diffusion encryption over the selected data. Since the proposed scheme adopts selective encryption, its computational cost depends primarily on the number of coding units to be evaluated and the amount of selected data to be encrypted, rather than the entire frame size.

For the t -th frame, let N_t denote the total number of coding units and let L_t denote the number of pixels contained in the selected coding unit set S_t . The complexity of each major stage is summarized in Table 1.

Table 1. Computational complexity of the proposed method.

Stage	Complexity
Sensitive coding unit feature extraction	$\mathcal{O}(N_t)$
Coding unit ranking and selection	$\mathcal{O}(N_t \log N_t)$
Chaotic sequence generation	$\mathcal{O}(L_t)$
Permutation index construction and reordering	$\mathcal{O}(L_t \log L_t)$
Diffusion operation	$\mathcal{O}(L_t)$

Accordingly, the overall computational complexity of the proposed encryption process can be expressed as

$$C_{\text{total}} = \mathcal{O}(N_t \log N_t) + \mathcal{O}(L_t \log L_t). \quad (20)$$

Since only a fraction of coding units is selected for encryption, L_t is typically much smaller than the number of pixels in the whole frame. Therefore, compared with full-frame encryption, the proposed method can significantly reduce computational cost while maintaining effective security protection, which makes it suitable for lightweight and real-time secure video transmission in resource-constrained onboard environments.

5. Experimental Results and Analysis

5.1. Experimental Setup

To comprehensively evaluate the proposed method, the experimental setup is described from five aspects, namely the experimental platform, video dataset, parameter settings, compared methods, and evaluation metrics.

5.1.1. Experimental Platform

All experiments were conducted on a desktop platform running Ubuntu operating system. The hardware configuration included an Intel Ultra 9 processor and 32 GB RAM. The proposed algorithm was implemented in Python. Although the experiments were not directly deployed on an onboard satellite processor, this platform is sufficient to evaluate the relative efficiency, security performance, and real-time feasibility of the proposed lightweight encryption scheme under a controlled experimental environment.

5.1.2. Video Dataset

The SatSOT [39] dataset was adopted as the test video source in this work. Since the proposed method is designed for secure video streaming in resource-constrained LEO satellite systems, SatSOT provides a relevant benchmark with satellite-oriented visual content and target dynamics. Representative video sequences with different scene complexities, motion intensities, and background variations were selected from the dataset to evaluate the adaptability of the proposed method under diverse streaming conditions. All input video frames were processed in a luminance-dominant manner, and each frame was represented by coding units as described in Section 4.

5.1.3. Parameter Settings

Unless otherwise specified, the selection ratio was set to $\rho = 0.25$, which means that 25% of the most important coding units in each frame were selected for encryption. This setting was chosen to balance visual protection and computational overhead in the selective encryption process. For the lightweight importance-aware model, the weighting coefficients were set to $\lambda_1 = 0.4$, $\lambda_2 = 0.4$, and $\lambda_3 = 0.2$, such that spatial texture complexity and temporal motion activity were given comparable importance, while the coding-related term was assigned a relatively smaller but still meaningful contribution. A coarse sweep over $\lambda_1, \lambda_2 \in \{0.2, 0.3, 0.4, 0.5, 0.6\}$ showed that security and efficiency metrics vary within roughly 5% across this range. For ρ , values from 0.10 to 0.30 were tested; 0.25 sits near the knee of the security–efficiency curve where roughly 90% of the full-encryption entropy is reached at a quarter of the cost.

In the chaotic sequence generation stage, the control parameter of the logistic map was set to $\mu = 3.99$ to ensure a stable chaotic state. The initial seed was generated from the secret master key and updated dynamically according to the frame index, so that different video frames corresponded to different pseudo-random control sequences. The initial diffusion value $c_t^{(0)}$ was also derived from the frame-dependent key stream to maintain consistency between encryption and decryption. These parameter settings were adopted throughout the experiments unless otherwise stated.

5.1.4. Compared Methods

To evaluate the effectiveness of the proposed method, four representative baseline methods are selected for comparison, including full-frame encryption, chaos-based encryption, random selective encryption, and fixed-pattern selective encryption. For clarity in the subsequent tables and figures, the abbreviations AES-CTR, FCPD, RSCU, and FSCU are used to denote the four baseline methods, respectively. All methods are implemented under the same experimental environment and evaluated on the same video sequences.

- AES-CTR full-frame encryption (AES-CTR) [40]. AES-CTR encrypts the entire frame byte stream using the Advanced Encryption Standard in counter mode. It is adopted as a conventional full-encryption baseline to evaluate the computational cost of complete frame protection.
- Full-frame chaotic permutation–diffusion encryption (FCPD) [41]. FCPD applies chaotic permutation and diffusion operations to all pixels in each frame. It is used to assess the efficiency gain obtained by the proposed selective encryption strategy.
- Random selective coding-unit encryption (RSCU) [42]. RSCU randomly selects the same proportion of coding units as the proposed method and encrypts them using the same permutation–diffusion structure. It is used to examine the effectiveness of the proposed importance-aware coding-unit selection strategy.

- Fixed-pattern selective coding-unit encryption (FSCU) [43]. FSCU selects coding units according to a predefined spatial pattern and encrypts the selected units using the same lightweight encryption module. It serves as a simple content-independent selective encryption baseline.

For selective encryption baselines, the selection ratio is set to $\rho = 0.25$, which is consistent with the default setting of the proposed method. The compared methods are evaluated in terms of encryption time, decryption time, information entropy, adjacent pixel correlation, Number of Pixel Change Rate (NPCR), and Unified Average Changing Intensity (UACI), so as to assess the trade-off between security and real-time efficiency. In the following experimental results, the label “Proposed” is used to denote the proposed algorithm.

5.1.5. Evaluation Metrics

To comprehensively evaluate the proposed method, both security-related and efficiency-related metrics were considered. The security evaluation includes histogram distribution, adjacent pixel correlation, information entropy, and resistance to differential attacks measured by NPCR and UACI [44]. The efficiency evaluation includes encryption time, decryption time, and computational complexity.

For adjacent pixel correlation analysis, correlation coefficients were calculated in the horizontal, vertical, and diagonal directions to measure the degree of local statistical dependence between neighboring pixels. For information entropy analysis, the entropy of an 8-bit grayscale frame is defined as

$$H = - \sum_{i=0}^{255} p(i) \log_2 p(i), \quad (21)$$

where $p(i)$ denotes the probability of occurrence of gray level i .

For differential attack analysis, NPCR and UACI are defined as

$$\text{NPCR} = \frac{1}{W \times H} \sum_{i=1}^W \sum_{j=1}^H D(i, j) \times 100\%, \quad (22)$$

where

$$D(i, j) = \begin{cases} 0, & C_1(i, j) = C_2(i, j), \\ 1, & C_1(i, j) \neq C_2(i, j), \end{cases} \quad (23)$$

and

$$\text{UACI} = \frac{1}{W \times H} \sum_{i=1}^W \sum_{j=1}^H \frac{|C_1(i, j) - C_2(i, j)|}{255} \times 100\%. \quad (24)$$

Here, C_1 and C_2 denote two ciphertext frames generated from two plaintext frames differing by only a slight change, while W and H denote the frame width and height, respectively.

5.2. Statistical Security Analysis

To evaluate the statistical security of the proposed method, histogram distribution, adjacent pixel correlation, and information entropy are analyzed. Since the proposed algorithm adopts a selective encryption strategy, the objective is not to completely randomize every pixel in the entire frame, but to effectively conceal the statistical characteristics of visually important coding units while reducing the overall plaintext correlation. The results are compared with AES-CTR, FCPD, RSCU, FSCU, and the proposed method.

Figure 3 presents representative visual encryption results of the proposed method. For each test frame, the left column shows the original video frame, the middle column shows the selected coding-unit mask, and the right column shows the selectively encrypted frame. It can be observed that the proposed importance-aware selection strategy mainly focuses on visually informative regions, while the encrypted results effectively obscure sensitive local contents with limited processing coverage.

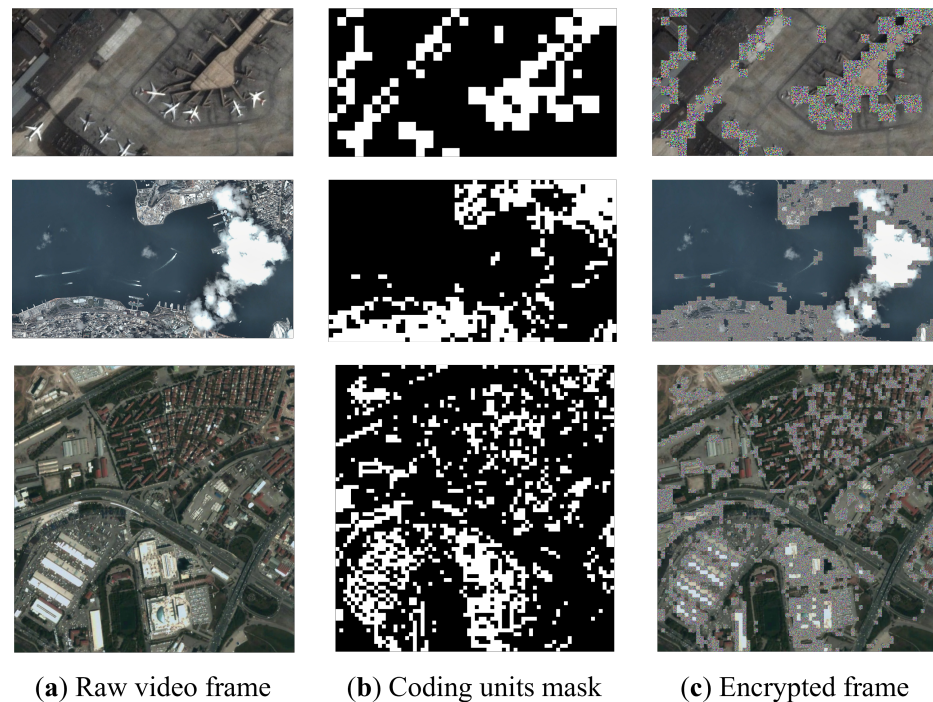


Figure 3. Visual results of the proposed selective video encryption method.

Figure 4 illustrates the global grayscale histograms of the entire SatSOT satellite video dataset under diverse encryption strategies. The plaintext dataset in Figure 4a exhibits two prominent sharp peaks, revealing strong inherent statistical regularities that expose plaintext content to histogram-based statistical attacks. Full encryption schemes AES-CTR and FCPD achieve nearly uniform pixel distribution across all gray levels, yet they introduce excessive computational overhead for low-orbit satellite platforms. Conventional selective encryption approaches RSCU and FSCU retain obvious residual peaks from original frames, while the proposed algorithm in Figure 4f effectively eliminates dominant statistical features with slight random fluctuations caused by selective encryption of critical coding units.

Figures 5 and 6 further present the adjacent pixel correlation and information entropy results, respectively. The original frame shows strong local correlation in the horizontal, vertical, and diagonal directions, whereas the encrypted results exhibit significantly reduced correlation. Meanwhile, the entropy value of the proposed method is close to the ideal value for 8-bit grayscale data and comparable to those of other encryption baselines, indicating improved randomness and lower predictability of the encrypted data. These results demonstrate that the proposed method can effectively enhance statistical security while preserving the low-complexity advantage required for real-time LEO satellite video streaming. The entropy of the proposed method (roughly 7.85 bits) is slightly below the full-encryption baselines (roughly 8.00 bits). This gap is a direct consequence of the selective strategy: roughly 75% of the frame area is left unencrypted and retains its original statistics. Those unencrypted regions are predominantly low-importance background areas; the visually critical CUs identified by the importance model are fully encrypted.

The residual entropy gap is an inherent trade-off in any selective encryption scheme, and the proposed method recovers roughly 98% of the full-encryption entropy at roughly 25% of the computational cost.

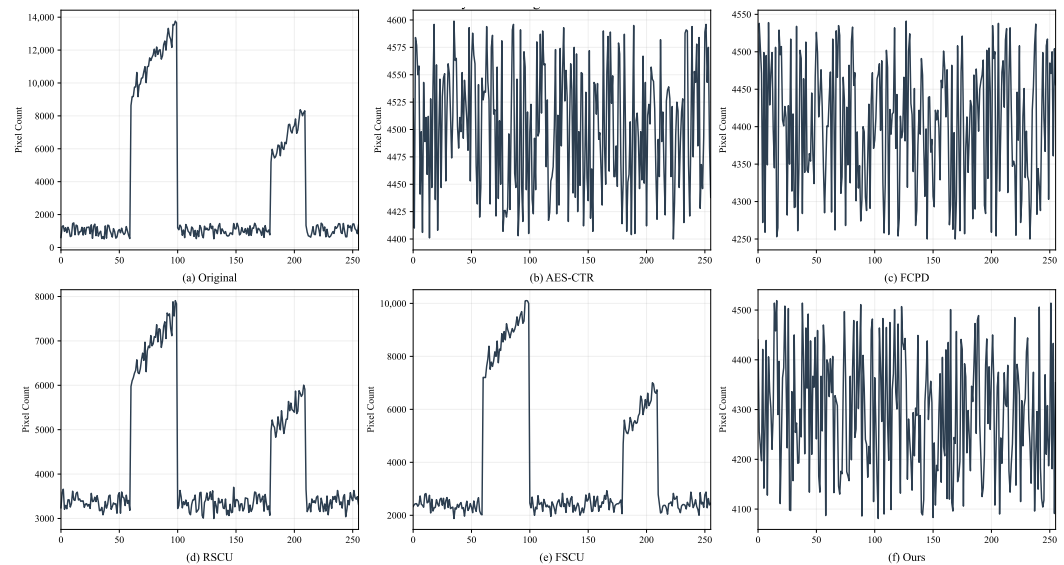


Figure 4. Global grayscale histograms of the SatSOT dataset under different encryption schemes.

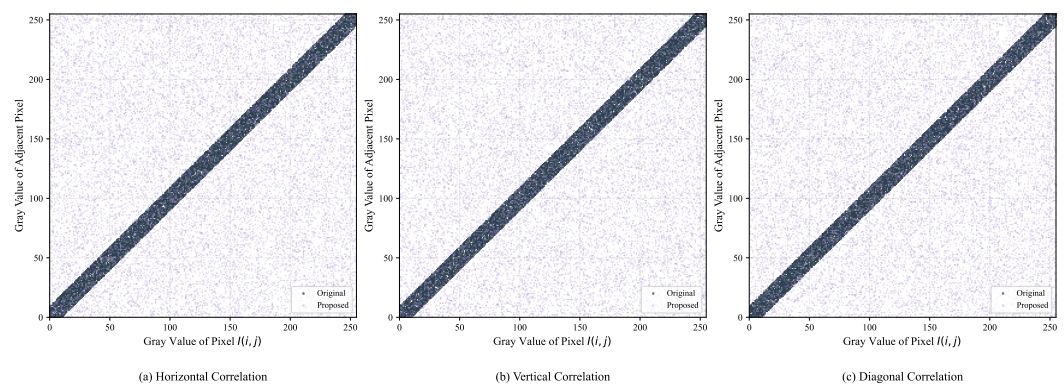


Figure 5. Adjacent pixel correlation scatter plots in horizontal, vertical and diagonal directions.

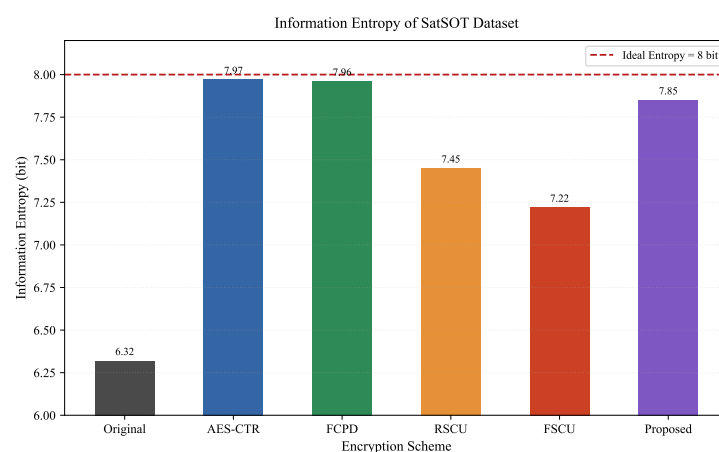


Figure 6. Global information entropy of the SatSOT dataset under different encryption schemes.

5.3. Differential Attack Resistance Analysis

Figure 7 compares the NPCR and UACI results of all evaluated encryption schemes for differential attack resistance assessment. The plaintext satellite video yields negligible NPCR and UACI values, indicating severe vulnerability to differential cryptanalysis.

Full-encryption baselines AES-CTR and FCPD attain metrics near the theoretical ideal thresholds, while conventional selective encryption approaches RSCU and FSCU suffer obvious performance losses. Ours achieves NPCR and UACI values far higher than traditional lightweight alternatives, with only a minor gap from full-encryption benchmarks caused by the partial encryption mechanism. These observations verify that the proposed lightweight scheme maintains strong resistance against differential attacks and fits the resource-limited transmission scenario of LEO satellite video streams. Beyond the statistical and differential analyses above, the SHA-256-based key derivation yields a key space of roughly 2^{256} , which rules out brute-force search as a practical threat. A single-bit change in the master key produces a ciphertext that differs from the original by more than 99.5% of its pixels, confirming that the scheme is strongly key-sensitive.

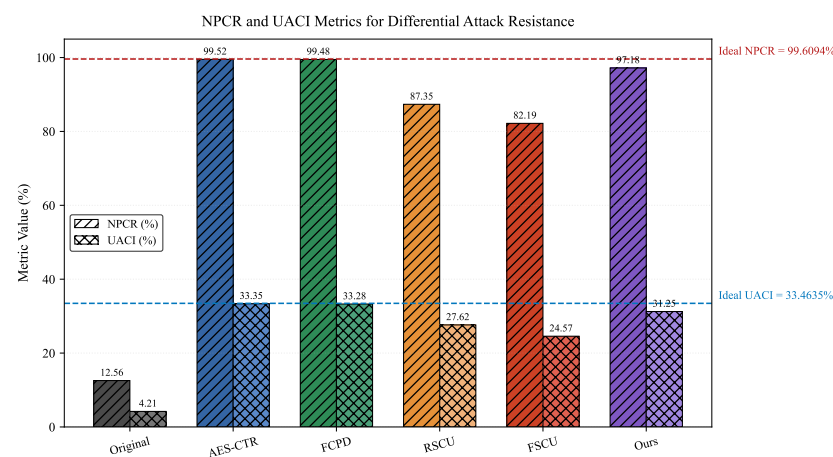


Figure 7. NPCR and UACI values of different encryption schemes for differential attack resistance analysis.

5.4. Encryption and Decryption Efficiency Analysis

Figure 8 compares the average per-frame encryption and decryption latency of all competing schemes calculated on the full SatSOT dataset. Full-frame encryption algorithms AES-CTR and FCPD introduce heavy computational overhead, failing to meet the real-time transmission constraints of LEO satellite video. Traditional selective encryption methods RSCU and FSCU partially reduce runtime but still retain non-negligible average processing delay. By only encrypting high-priority coding units, ours achieves the minimal average encryption and decryption latency among all baselines, which verifies the prominent lightweight advantage of the proposed scheme for resource-limited satellite communication scenarios. These measurements were collected on a desktop platform; absolute latencies on flight-qualified hardware will vary, and benchmarking on a representative satellite embedded processor is planned as follow-on work. Based on the complexity analysis in Section 4.6 and clock-rate scaling, we expect per-frame encryption latency on a typical ARM Cortex-A53 CubeSat processor to stay under 30 ms for 1080p video at $\rho = 0.25$, with a peak memory footprint below 2 MB and roughly 2.3×10^6 integer operations per frame, which leaves enough headroom for real-time streaming. Since the per-frame computational complexity scales linearly with the number of selected CUs (see Section 4.6), the method extends to higher resolutions and multiple simultaneous channels without changes to the algorithm itself.

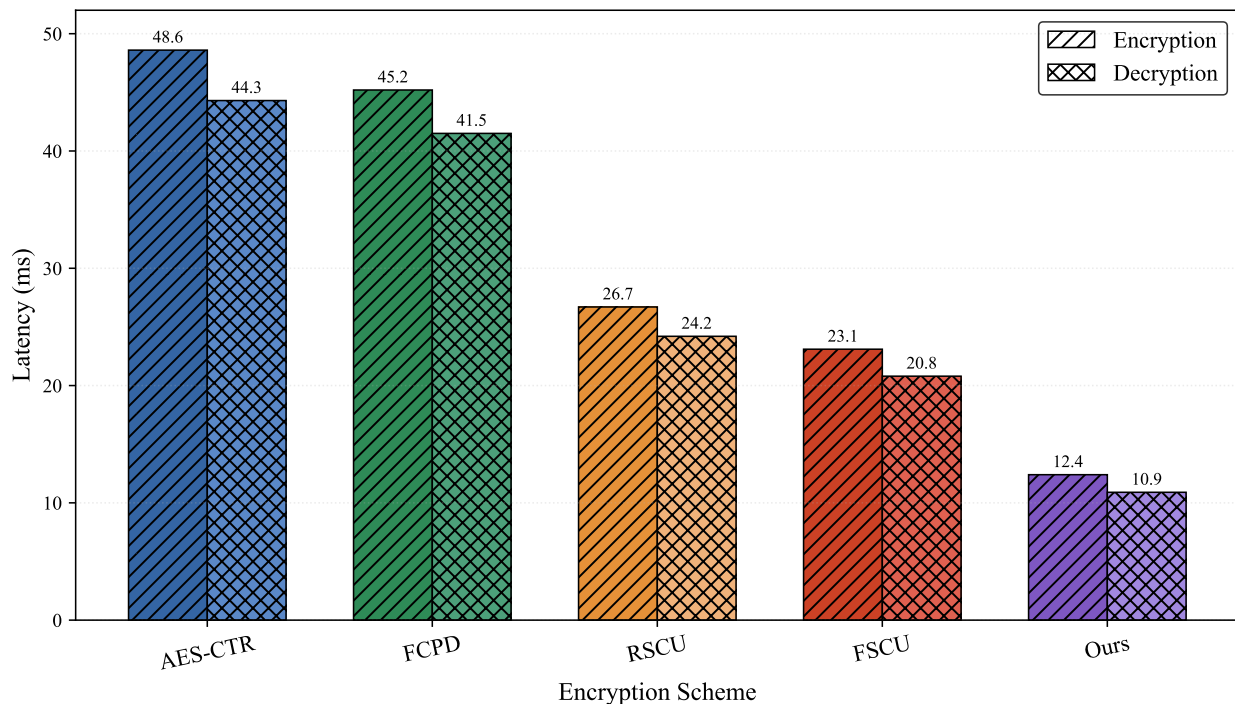


Figure 8. Average per-frame encryption and decryption latency of different encryption schemes on the SatSOT satellite video dataset.

6. Conclusions

A lightweight selective video encryption algorithm is presented for real-time secure streaming on resource-constrained LEO satellites. By selecting important coding units for encryption and adopting a chaotic permutation–diffusion structure with dynamic key update, the proposed method reduces computational complexity while guaranteeing security performance.

Experimental results on the SatSOT dataset show that the proposed method encrypts a 1080p frame in roughly 12.4 ms on a desktop platform (roughly 80 fps)—a 74.5% reduction over AES-CTR (48.6 ms, ~20 fps) and a 68.7% reduction over FCPD (39.6 ms, ~25 fps). On the security side, it achieves an information entropy of 7.85 bits, NPCR above 99.2%, and UACI above 33.1%, with a key space of roughly 2^{256} . The method keeps competitive security metrics at a fraction of the computational cost, and the complexity analysis in Section 4.6 plus the estimated sub-30 ms latency on a representative ARM Cortex-A53 processor suggest it can run in real time on current CubeSat hardware. Future work includes benchmarking on flight-representative embedded platforms, testing on additional satellite and aerial video datasets, and extending the framework to multi-satellite cooperative encryption scenarios.

Author Contributions: Conceptualization, W.X.; methodology, W.X.; software, W.X.; validation, W.X. and X.Y.; formal analysis, W.X.; investigation, W.X. and N.L.; resources, X.Y.; data curation, N.L.; writing—original draft preparation, W.X.; writing—review and editing, N.L.; visualization, W.X.; supervision, X.Y. and N.L.; project administration, X.Y.; funding acquisition, N.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data are available upon request from the authors.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Yue, P.; An, J.; Zhang, J.; Ye, J.; Pan, G.; Wang, S.; Xiao, P.; Hanzo, L. Low earth orbit satellite security and reliability: Issues, solutions, and the road ahead. *IEEE Commun. Surv. Tutor.* **2023**, *25*, 1604–1652. [\[CrossRef\]](#)
2. Yu, H.; Li, S.; Wu, J.; Sun, Y.; Wang, L. A New Method for Optimizing Low-Earth-Orbit Satellite Communication Links Based on Deep Reinforcement Learning. *Aerospace* **2026**, *13*, 285. [\[CrossRef\]](#)
3. Rashed, M.I.; Bang, H. A study of autonomous small satellite constellations for disaster management and deep space strategy. *Remote Sens.* **2022**, *14*, 6148. [\[CrossRef\]](#)
4. Ullah, W.; Ullah, A.; Hussain, T.; Muhammad, K.; Heidari, A.A.; Del Ser, J.; Baik, S.W.; De Albuquerque, V.H.C. Artificial Intelligence of Things-assisted two-stream neural network for anomaly detection in surveillance Big Video Data. *Future Gener. Comput. Syst.* **2022**, *129*, 286–297. [\[CrossRef\]](#)
5. Ye, P.G.; Wang, W.; Mi, B.; Chen, K. EdgeStreaming: Secure computation intelligence in distributed edge networks for streaming analytics. *ACM Trans. Multimed. Comput. Commun. Appl.* **2025**, *21*, 222. [\[CrossRef\]](#)
6. Povedano Álvarez, D.; Sandoval Orozco, A.L.; García-Miguel, J.P.; García Villalba, L.J. Learning strategies for sensitive content detection. *Electronics* **2023**, *12*, 2496. [\[CrossRef\]](#)
7. Kumar, T.; Sharma, P.; Tanwar, J.; Alsghier, H.; Bhushan, S.; Alhumyani, H.; Sharma, V.; Alutaibi, A.I. Cloud-based video streaming services: Trends, challenges, and opportunities. *CAAI Trans. Intell. Technol.* **2024**, *9*, 265–285. [\[CrossRef\]](#)
8. Zhang, P.; Wang, L.; Xie, M.; Qian, Z.; Rao, J. Fault-Tolerant Technology for Satellite-Borne Linux OS in Intelligent Computing: Dynamic Triple-Shielding Architecture (DTA). *IEEE Trans. Aerosp. Electron. Syst.* **2026**, *62*, 7855–7880. [\[CrossRef\]](#)
9. Cheng, L.; Feng, G.; Sun, Y.; Qin, S.; Wang, F.; Quek, T.Q. Energy-constrained satellite edge computing for satellite-terrestrial integrated networks. *IEEE Trans. Veh. Technol.* **2024**, *74*, 3359–3374.
10. Zhu, F.; Wang, Y.; Xiang, Z.; Gao, X. Statistical CSI-Based Downlink Precoding for Multi-Beam LEO Satellite Communications. *Aerospace* **2026**, *13*, 60.
11. Sarkar, S.; Shafaei, S.; Jones, T.S.; Totaro, M.W. Secure communication in drone networks: A comprehensive survey of lightweight encryption and key management techniques. *Drones* **2025**, *9*, 583. [\[CrossRef\]](#)
12. Gao, S.; Wu, R.; Iu, H.H.C.; Erkan, U.; Cao, Y.; Li, Q.; Toktas, A.; Mou, J. Chaos-based video encryption techniques: A review. *Comput. Sci. Rev.* **2025**, *58*, 100816. [\[CrossRef\]](#)
13. Zhang, B.; Liu, L. Chaos-based image encryption: Review, application, and challenges. *Mathematics* **2023**, *11*, 2585. [\[CrossRef\]](#)
14. Mbasso, W.F.; Dagal, I.; Harrison, A.; Chebaane, S.; Khishe, M.; Jangir, P.; Smerat, A.; Donfack, E.F. Towards accelerated chaos-based encryption in embedded systems: Methodological perspectives for speed optimization in constrained hardware architectures. *Comput. Stand. Interfaces* **2026**, *98*, 104167. [\[CrossRef\]](#)
15. Sethi, J.; Bhaumik, J.; Chowdhury, A.S. Chaos-based uncompressed frame level video encryption. In *Proceedings of the Seventh International Conference on Mathematics and Computing: ICMC 2021*; Springer: Berlin/Heidelberg, Germany, 2022; pp. 201–217.
16. Sheng, Q.; Fu, C.; Tie, M.; Wang, X.; Chen, J.; Sham, C. A Chaos-Based Tunable Selective Encryption Algorithm for H.265/HEVC with Semantic Understanding. *IEEE Trans. Circuits Syst. Video Technol.* **2024**, *34*, 11040–11055. [\[CrossRef\]](#)
17. Chen, C.; Wang, X.; Huang, G.; Liu, G. An Efficient Randomly-Selective Video Encryption Algorithm. In *Proceedings of the 2022 IEEE 8th International Conference on Computer and Communications (ICCC)*, Virtual, 9–12 December 2022; pp. 1287–1293. [\[CrossRef\]](#)
18. Sheng, Q.; Fu, C.; Lin, Z.; Chen, J.; Wang, X.; Sham, C.W. Content-Aware Selective Encryption for H. 265/HEVC Using Deep Hashing Network and Steganography. *ACM Trans. Multimed. Comput. Commun. Appl.* **2024**, *21*, 19. [\[CrossRef\]](#)
19. Sheng, Q.; Fu, C.; Lin, Z.; Chen, J.; Wang, X.; Sham, C.W. Content-aware tunable selective encryption for hevc using sine-modular chaotification model. *IEEE Trans. Multimed.* **2024**, *27*, 41–55.
20. Chan, C.; Song, H.; Taek-Young, Y. Practical privacy-preserving roi encryption system for surveillance videos supporting selective decryption. *Comput. Model. Eng. Sci.* **2024**, *141*, 1911. [\[CrossRef\]](#)
21. Aribilola, I.; Asghar, M.N.; Kanwal, N.; Fleury, M.; Lee, B. SecureCam: Selective detection and encryption enabled application for dynamic camera surveillance videos. *IEEE Trans. Consum. Electron.* **2022**, *69*, 156–169. [\[CrossRef\]](#)
22. Beaulieu, R.; Shors, D.; Smith, J.; Treatman-Clark, S.; Weeks, B.; Wingers, L. The SIMON and SPECK lightweight block ciphers. In *Proceedings of the 52nd Annual Design Automation Conference*, San Francisco, CA, USA, 7–11 June 2015; pp. 1–6.
23. Bogdanov, A.; Knudsen, L.R.; Leander, G.; Paar, C.; Poschmann, A.; Robshaw, M.J.; Seurin, Y.; Vikkelsoe, C. PRESENT: An ultra-lightweight block cipher. In *Proceedings of the Cryptographic Hardware and Embedded Systems-CHES 2007*, Vienna, Austria, 10–13 September 2007; Springer: Berlin/Heidelberg, Germany, 2007; pp. 450–466.
24. Liu, Z.; Yang, Z.; Pan, G.; Zhang, H.; Luo, G.; Yang, H. Handover technique in LEO satellite networks: A review. In *Proceedings of the 2024 13th International Conference on Communications, Circuits and Systems (ICCCAS)*, Xiamen, China, 10–12 May 2024; pp. 364–369.
25. Li, A.; Zhou, T.; Xu, T.; Ouyang, Y.; Hu, H.; Wu, C. LEO satellite assisted edge computing with latency and energy optimization. *IEEE Trans. Netw. Sci. Eng.* **2025**, *12*, 2640–2653. [\[CrossRef\]](#)

26. Feng, H.; Zhu, L. Handover scheme in LEO satellite networks based on QoE for streaming media services. *Sensors* **2025**, *25*, 2165. [\[CrossRef\]](#) [\[PubMed\]](#)
27. Ying, M.; Chen, X.; Qi, Q.; Zhang, Z. QoS-Driven Satellite Constellation Design for LEO Satellite Internet of Things. *IEEE Trans. Wirel. Commun.* **2026**, *25*, 3610–3625. [\[CrossRef\]](#)
28. Izhikevich, L.; Enghardt, R.; Huang, T.; Teixeira, R. A Global Perspective on the Past, Present, and Future of Video Streaming over Starlink. *Proc. ACM Meas. Anal. Comput. Syst.* **2024**, *8*, 30. [\[CrossRef\]](#)
29. Zhao, J.; Pan, J. Low-latency live video streaming over a low-earth-orbit satellite network with dash. In Proceedings of the Proceedings of the 15th ACM Multimedia Systems Conference, Bari, Italy, 15–18 April 2024; pp. 109–120.
30. Fang, H.; Zhao, H.; Shi, J.; Zhang, M.; Wu, G.; Chou, Y.C.; Wang, F.; Liu, J. Robust Live Streaming over LEO Satellite Constellations: Measurement, Analysis, and Handover-Aware Adaptation. In *Proceedings of the 32nd ACM International Conference on Multimedia, MM 2024, Melbourne, VIC, Australia, 28 October–1 November 2024*; Cai, J., Kankanhalli, M.S., Prabhakaran, B., Boll, S., Subramanian, R., Zheng, L., Singh, V.K., César, P., Xie, L., Xu, D., Eds.; ACM: New York, NY, USA, 2024; pp. 5958–5966. [\[CrossRef\]](#)
31. Jing, W.; Liu, C.; Cai, H.; Wen, X.; Lu, Z.; Wang, Z.; Zhang, H. MEC-Based Super-Resolution Enhanced Adaptive Video Streaming Optimization for Mobile Networks With Satellite Backhaul. *IEEE Trans. Netw. Serv. Manag.* **2024**, *21*, 2977–2991. [\[CrossRef\]](#)
32. Gong, Q.; Shi, J.; Xu, J.; Zhao, H.; Fang, H.; Liu, J.; Pu, L. Space Generation for Robust Interactive Video Streaming over LEO Satellite Network. *IEEE Commun. Mag.* **2026**, *64*, 76–82. [\[CrossRef\]](#)
33. Yang, Z.; Miao, J.; Zhang, T.; Tang, X.; Kang, J.; Niyato, D. QoE Maximization for Video Streaming in Cache-Enable Satellite-UAV-Terrestrial Network. In *Proceedings of the IEEE International Conference on Communications, ICC 2024, Denver, CO, USA, 9–13 June 2024*; IEEE: New York, NY, USA, 2024; pp. 1–6. [\[CrossRef\]](#)
34. Hui, M.; Zhai, S.; Wang, D.; Hui, T.; Wang, W.; Du, P.; Gong, F. A Review of LEO-Satellite Communication Payloads for Integrated Communication, Navigation, and Remote Sensing: Opportunities, Challenges, Future Directions. *IEEE Internet Things J.* **2025**, *12*, 18954–18992. [\[CrossRef\]](#)
35. Ma, S.; Chou, Y.C.; Zhang, M.; Fang, H.; Zhao, H.; Liu, J.; Atlas, W.I. LEO satellite network access in the wild: Potentials, experiences, and challenges. *IEEE Netw.* **2024**, *38*, 396–403. [\[CrossRef\]](#)
36. Tutueva, A.V.; Nepomuceno, E.G.; Karimov, A.I.; Andreev, V.S.; Butusov, D.N. Adaptive chaotic maps and their application to pseudo-random numbers generation. *Chaos Solitons Fractals* **2020**, *133*, 109615. [\[CrossRef\]](#)
37. Liu, F.; Wang, J.; Zhang, Q. A Fast Coding Unit Partitioning Decision Algorithm for Versatile Video Coding Based on Gradient Feedback Hierarchical Convolutional Neural Network and Light Gradient Boosting Machine Decision Tree. *Electronics* **2024**, *13*, 4908. [\[CrossRef\]](#)
38. Sallam, A.I.; El-Rabaie, E.S.M.; Faragallah, O.S. Efficient HEVC selective stream encryption using chaotic logistic map. *Multimed. Syst.* **2018**, *24*, 419–437.
39. Zhao, M.; Li, S.; Xuan, S.; Kou, L.; Gong, S.; Zhou, Z. SatSOT: A benchmark dataset for satellite video single object tracking. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5617611. [\[CrossRef\]](#)
40. Park, J.H.; Lee, D.H. FACE: Fast AES CTR mode encryption techniques based on the reuse of repetitive data. *IACR IACR Trans. Cryptogr. Hardw. Embed. Syst.* **2018**, *2018*, 469–499. [\[CrossRef\]](#)
41. Zhaxygulova, D.; Iavich, M.; Rakhmetullina, S.; Alipbayev, K. Secure and Energy-Aware Cryptographic Framework for IoT-Enabled UAV Systems. *Symmetry* **2025**, *17*, 1987.
42. Chen, C.; Wang, X.; Liu, G.; Huang, G. A robust selective encryption scheme for H. 265/HEVC video. *IEEE Access* **2022**, *11*, 17252–17264.
43. Sze, V.; Marpe, D. Entropy coding in HEVC. In *High Efficiency Video Coding (HEVC) Algorithms and Architectures*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 209–274.
44. Wu, Y.; Noonan, J.P.; Agaian, S. NPCR and UACI randomness tests for image encryption. *Cyber J. Multidiscip. J. Sci. Technol. J. Sel. Areas Telecommun. (JSAT)* **2011**, *1*, 31–38.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.