Accelerated Security Model-Driven Encryption with Remote Control for Satellite Imagery

Salah-Eddine Tbahriti
Center of Satellites Development – Algerian Space Agency
Oran, Algeria

***Abstract*—In light of the increasing reliance on Small Satellites for Earth Observation, ensuring the security of satellite imagery is paramount, particularly given their limited onboard resources. In this paper, we introduce a security model that formally and semantically specifies the security requirements for satellite imagery. Building on this foundation, we propose a secure encryption protocol that ranks images by security importance, filtering them according to predefined specifications so that only those necessitating protection are selected. This dual-stage process enables selective control and optimal resource allocation, while our fast encryption algorithm is explicitly designed to be aware of onboard resource constraints and incorporates a remote-control mechanism to dynamically adjust security parameters.**

Keywords—Security Specification, Selected Encryption, Image encryption, Security Analysis

I. INTRODUCTION

The emergence of Small Satellites for Earth Observation (SSEO) has fundamentally transformed the acquisition, storage, and transmission of satellite imagery (*SaI*) enabling applications such as wildfire detection, hurricane diagnosis, traffic incident management, and emergency response coordination. However, SSEO are constrained by limited onboard resources, including processing power, storage capacity, and energy supply. These limitations hinder the ability to perform complex tasks such as image encryption directly in space [1], [2]. In these scenarios, decision makers require immediate access to extensive satellite data in order to devise timely and effective countermeasures. However, while SSEO systems are indispensable for time-sensitive operations, the data security is further complicated by various factors, particularly when considering the feasibility of implementing AES encryption on resource-constrained onboard systems [3]. While certain hardware implementations have demonstrated efficient AES processing, its conventional application in satellite imagery presents significant challenges that undermine its effectiveness. One key issue is the lack of precise criteria for identifying sensitive satellite imagery. It is crucial to rigorously define what constitutes sensitive data to ensure that only relevant imagery undergoes higher levels of encryption. Not all satellite imagery requires uniform protection; for instance, low-resolution environmental images may be suitable for public release, while images of strategic locations demand stronger encryption. Therefore, a well-defined data sensitivity framework is essential to enable tiered security protocols, where encryption intensity is applied based on the sensitivity classification of the data. A further problem arises from the indiscriminate application of AES encryption to all satellite imagery, regardless of the data's sensitivity. This blanket approach fails to differentiate between high-priority and low-sensitivity data, leading to unnecessary processing for information that does not require such stringent protection. As a result, comprehensive AES encryption can impose excessive computational demands, particularly when large portions of the data are of low sensitivity, diverting valuable resources away from critical satellite functions. To overcome these limitations, we propose a dual-faceted cryptosystem that integrates a formal security model with an adaptive, lightweight encryption protocol based on an enhanced Galois Linear Feedback Shift Register (G-LFSR) [4]. The formal security model dynamically classifies image content into sensitivity level using a rigorous mathematical framework, allowing for the dynamic control of cryptographic resources. Images of higher sensitivity are assigned more robust protection, with the security algorithm applied based on the sensitivity class and the importance of the original image. The cryptosystem allows controlling the encryption in a manner that is both optimized and resilient. Thus, only sensitive images requiring protection are encrypted, thereby conserving valuable resources. Our lightweight encryption protocol is based on an enhanced G-LFSR architecture. We validate the security properties of the encryption protocol through extensive experiments. The paper is structured as follows: Section 2 overview the related work. Section 3 details the security model, Section 4 introduces the security protocol with a fast encryption algorithm, Section 5 presents the experiments and demonstrates resistance to attacks, and Section 6 concludes the paper.

II. RELATED WORK

In satellite applications, various cryptographic approaches have been explored. The work of [5] analysed lightweight algorithms like PRESENT and SPECK, optimized for resource-constrained devices. PRESENT uses a substitution-permutation network with a 4-bit S-Box and a 64-bit block size, ensuring efficient hardware performance with minimal rounds. SPECK, based on a Feistel network and ARX (Addition, Rotation, XOR) operations, offers efficiency across hardware and software platforms common in space systems. Both algorithms have been assessed for resistance to differential and linear cryptanalysis, proving secure and efficient for satellite data encryption in radiation-exposed environments. The work of [6] integrates DNA-based substitution techniques, though careful selection of feedback polynomials remains critical to avoid vulnerabilities. Chaos-based encryption,

exploiting chaotic systems' sensitivity and randomness, provides strong diffusion and confusion, making it suitable for secure communication. Research of [7] shows that such methods ensure significant changes in ciphertext with even minor plaintext alterations, ideal for real-time systems like IoT. In [8] a hybrid method combining hyper-chaotic systems is proposed, Singular Value Decomposition, RC5 operations, and a custom S-box, merged into a single augmented image before encryption. Although multi-layered encryption strengthens security while optimizing computational resources, improper chaotic map selection can reduce randomness, making systems vulnerable to cryptanalysis. Additionally, hardware implementations of chaos-based encryption are vulnerable to side-channel attacks such as power analysis [7][9]. In contrast, our work presents a rule-based security model, providing a concise framework for specifying security by classifying images based on both their characteristics and intended use, marking a departure from traditional methods. It defines a protocol to refine the importance of satellite imagery, applying dynamic encryption algorithms based on sensitivity levels, optimizing both security and resource use.

III. Security Model

The security model we propose enables the definition and control of security requirements for satellite imagery. It addresses the specific security needs of satellite data by employing fine-grained security rules that facilitate precise control over security specifications. This model allows the SSEO mission manager to define security specifications, outlining the control measures that a requester must meet to access the provided *SaI*.

A. Security Rules

The security specification of a *SaI* is defined according to dimensions set called security rule, noted as SR_l where each rule is defined by a tuple-topic **<Rep_i, Pur_j, Lev_k>** where:

Rep_i topic describes the recipient of *SaI* and mentions to whom *SaI* can be transmitted. Recipient topic will consist of a set of names, listing legal each entity that could possibly receive *SaI*. Satellites Data recipients play a crucial role in the degree of security protection and data collection.

Pur_j topic describes the purpose of *SaI* requester and states the intent for which this *SaI* will be used by the requester. Purpose topic outlines the operational reasons for requiring access to *SaI*.

Lev_k topic specifies the security level of the topic rule, which is associated to the couple <Rep_i, Pur_j>. In this paper and for the sake of simplicity, we consider three security levels; where for each level an encryption algorithm, e.g., AES or another one more lightweight, will be applied. Note that, the values of *Rep* and *Pur* topics are defined according to a finite set (which is based on ontology domain) that enumerates the legal possible values that can be taken by the corresponding topic. For instance, a subset of domain for the recipient topic Rep_i is {Government, Corporation, University, Research-Lab, Public}, and a subset of domain for the purpose topic Pur_j is {Defense, Land-Observation, Research, Education}. To ensure clarity and expressiveness, each security rule is linked to a specific value of the pair <Rep_i, Pur_j>. The use of recipient and purpose as key dimensions for defining security rules is essential for precise and context-aware security control. The recipient dimension aligns with **access control** principles and domain-specific jurisdictional requirements, while the purpose dimension corresponds to regulatory constraints and sanctioned objectives (e.g., defense, land observation). Thus, the security rules control the conditions under which a user can access *SaI*, defining the possible values of recipient and purpose. They also control the application of security parameters, such as encryption mechanisms defined by the level topic Lev_k. The values within the pair <Rep_i, Pur_j> are defined according to a finite ontology-driven set. This ontology represents the domain-specific context, enumerating legally permissible and contextually relevant values for each topic. By aligning security rules with ontology, the security model enforces context-aware access control restrictions and automatically adjusts security levels based on evolving *SaI* sensitivity and policy requirements. In fact, the security model incorporates dynamic security level adjustment, enabling the initial value of Lev_k to adapt in real-time based on several contextual factors. One notable factor is its responsiveness to on-board environmental conditions, such as the impact of Single Event Upsets (SEU), that are caused by high-energy particles such as cosmic rays and solar radiation and pose significant risks to on-board systems, which can induce bit flips in critical components of onboard systems. By monitoring SEU stamps in real time, the system adjusts Lev_k to enhance protection when the integrity of the cryptographic process might be compromised, ensuring resilient and secure operations. For instance, let us consider the set of security rules of Table I. Deciding about the choice of the encryption mechanism to apply to *SaI* depends on two parameters: the security specifications class that *SaI* belongs to and the second parameter is the *set* conditions that *SaI* fulfil that we define in the Section 3.

TABLE I. Example of security rules

SR_i	Rep_i = {Government, Corporation, University, Research-Lab, Public}	Pur_j= {Defense, Land-Observation, Research, Education}
SR_1	Rep_1 = Government	Pur_1 = Defense
SR_2	Rep_1 = Government	Pur_2 = Land-Observation
SR_5	Rep_2 = Corporation	Pur_3 = Research
SR_{11}	Rep_3 = University	Pur_3 = Research
SR_{12}	Rep_3 = University	Pur_4 = Education

B. Security Specifications Class

For each *SaI*, SSEO mission manager defines a set of security specifications class, noted as SSC_{SaI}, through the to the security rules statement. Thus, SSC_{SaI} describes how sensitive a set of security specifications related to *SaI* is, and the possible sensitivity classes can be *high sensitive*, *sensitive*, *less sensitive*, and *public*, which correspond respectively to four classes. Thus, each class refers to a one security level, which is linked to a given encryption algorithm. For instance, we describe in Table II three classes. The $class_1$ contains rules as recipient topic Rep_1 the government value <Rep_1, Pur_i>. Then, $class_1$ is considered as highly sensitive. Then, full AES is applied to *SaI* specified within $class_1$.

The $class_2$ contains rules as recipient topic Rep_2 the university value $<Rep_2, Pur_i>$ and $class_3$ contains as recipient Rep_3 the university value $<Rep_3, Pur_i>$. The security expert is responsible to define the rank of sensitivity. Furthermore, the satellite owner may have different views about which *SaI* is considered as sensible, since the same *SaI* can belong more than one class (classified as sensible and public) according to the requesters and the given purposes topics. In order to avoid, such a conflict classification, we introduce the notion of ***security subsumption***, noted as $\subseteq_S$, to capture the security semantic importance relation among values of security specifications class. The value *Government* for example, is more important than any other value of Rep_i with respect to security consideration. In addition, the class ranking sensitivity depends on the level that the class belongs to. Then, $class_3$ is less sensitive than $class_2$ and this latter is less sensitive than $class_1$.

TABLE II. EXAMPLE OF SECURITY SPECIFICATIONS CLASSES

SaI	$class_i$	$<Rep_i, Pur_j>$	Lev_k= {AES, Fast Encryption}
img_1	$class_1$	$<Rep_1, Pur_1>$	Lev_1 = AES
	$class_1$	$<Rep_3, Pur_2>$	Lev_1 = AES
img_2	$class_2$	$<Rep_2, Pur_2>$	Lev_2 = Fast Encryption otherwise AES
	$class_2$	$<Rep_3, Pur_2>$	Lev_2 = Fast Encryption otherwise AES
img_3	$class_3$	$<Rep_3, Pur_3>$	Lev_2 = Fast Encryption
	$class_3$	$<Rep_4, Pur_4>$	Lev_2 = Fast Encryption

For instance, let us consider three satellite images img_1, img_2 and img_3 of Fig.1. The corresponding security specifications sets of img_1, img_2 and img_3 are respectively:

img_1 {$<Rep_1, Pur_1>$, $<Rep_3, Pur_2>$} img_2 {$<Rep_2, Pur_2>$, $<Rep_3, Pur_2>$} img_3 {$<Rep_3, Pur_3>$, $<Rep_4, Pur_4>$}

img_1 is requested by: Rep_1=*Government* and Rep_3=*University* then, according to security subsumption img_1 is classified in $class_1$ since *University* $\subseteq_S$ *Government* and AES algorithm will be applied on img_1. The same reasoning is applying to img_2 where *Research-Lab* $\subseteq_S$ *Corporation,* hence, img_2 is classified in $class_2$ and the security algorithm that will be applied on img_2 is fast encryption algorithm otherwise AES.

img_1 img_2 img_3

Fig. 1. Sub-set of satellite imagery

$<Rep_2, Pur_2>$ call Lev_2 of security level 2. This means that for level 2**,** we can call the lightweight protocol to encrypt img_3 otherwise full AES will applied. Note that, each pair of *Rep* and *Pur* defines a distinct security context for accessing *SaI*. The condition to call a corresponding security algorithm will be discussed in Sect. IV.

C. Security Specifications Enforcement

The enforcement of the security model on-board is based on guided through by parser-query mapping. The ground station sends a query to the satellite for capturing images, with each *SaI* corresponding to a specific geographic area. This request includes information for each *SaI*, such as the recipient type (e.g., government, research institution, etc.) and the purpose of the request. The on-board processor parses and analyses the request (which includes the values $\{Rep_i, Pur_j\}$ related to requested *SaI*) and the security model extracts and assesses these details. Based on this analysis, the security model assigns a security classification to each requested *SaI*, determining the level of protection and handling required. The ontology, stored locally in a structured database, is queried to validate and interpret Rep_i and Pur_j. Based on the values retrieved, the mapping assigns a security level Lev_k from a predefined set. The, each $<Rep_i, Pur_j, Lev_k>$ tuple triggers a policy stored onboard, dictating the encryption level, access restrictions, and retention period. This process ensures that *SaI* capture and data handling are aligned with security specifications.

IV. FAST ENCRYPTION PROTOCOL

The security model selectively filters satellite images based on predefined security specifications, ensuring that only those requiring protection undergo encryption. The security protocol refines this filtered set by determining the most appropriate encryption algorithm to apply. This two-tier approach facilitates controlled, resource-aware encryption while maintaining robust security standards. The proposed lightweight encryption protocol comprises two stages. Initially, the data sensitivity of *SaI* is assessed using an importance function that quantifies its confidentiality level based on two metrics: entropy and correlation. Subsequently, leveraging these metrics, a lightweight encryption algorithm is applied that maintains the requisite level of security without imposing significant computational overhead.

A. Entropy metric

The entropy metric, as in formula (1), quantifies the potential leakage of sensitive information in terms of bits, thereby assessing the security importance of the satellite imagery. Essentially, entropy measures the uncertainty inherent in *SaI* by quantifying the average information content of the associated random variable. A higher entropy value indicates increased unpredictability, making it more challenging for an adversary to infer the underlying data [10] [11]. Consequently, greater entropy corresponds to enhanced randomness in *SaI*.

$$E(SaI) = -\sum p_i \log_2(p_i),\ i \leq N \qquad (1)$$

where N is gray levels number and p_i $(0 < p_i < 1)$ is the proportion of points of *SaI* image having the gray level i.

B. Correlation coefficient metric

The correlation coefficient, as defined in formula (2), quantifies the degree of association among variables within *SaI*. In a plaintext satellite image, adjacent pixels typically exhibit high

correlation, meaning that gray values vary minimally over larger regions. Such correlation can be exploited by attackers for statistical analysis-based attacks. For a given *SaI*, the variables correspond to adjacent pixels, for example, x_i, y_i of *SaI*. The strength of the relationship varies in degree based on the value of correlation.

$$R_{xy} = cov(x, y) / \sigma(x)\sigma(y) \qquad (2)$$

where *cov* is the covariance and σ the standard deviation. We randomly select 100 pairs of neighbouring pixels from *SaI* in each orientation to assess the intrinsic importance of the original image based on pixel correlation. Correlation coefficient values approaching one indicate a strong linear relationship between the compared pixels, whereas values near zero suggest a weak association.

C. The Importance function

Previous metrics allow to quantify the importance of an original image. Drawing on these metrics, we define the importance function that looks for the high value, noted r_{SaI}, which is calculated through the formula (3):

$$r_{SaI} = Min\ (Norm[E(SaI), R_{xy}]) \qquad (3)$$

The output $r_{SaI} \in [0, 1]$ is computed by applying a gradual conjunction operator (*Min*) to the normalized entropy and correlation metrics (normalized via the function *Norm*). This importance function subsequently ranks the original images prior to encryption. In accordance with the established security model and classification, only images deemed highly important are subjected to full AES encryption. For *SaI* initially classified as $class_2$, its entropy and correlation coefficients are calculated using formulas (1) and (2), respectively, and then r_{SaI} is computed using formula (3). A set of thresholds τ_k is defined by the security expert. If $r_{SaI} < \tau_k$, for *SaI* in $class_k$ (or lower), the first security algorithm is applied, otherwise the second is used. For example, for img_3, ($class_2$) with $E(img_3) = 5.3362$, correlation = 0,73 and $\tau_2 = 0.7$, formula (3) yields $r_{img3} = 0.6$. Then, lightweight encryption is applied on img_3.

D. Galois-LFSR-based Fast Encryption Algorithm

The fast encryption algorithm we develop employs a Galois-mode Linear Feedback Shift Register (G-LFSR) augmented by two novel capabilities: 1) a nonlinear filtering function for pseudo-random sequence generation and 2) a dynamic confusion operation to enhance resistance against analytical attacks. The algorithm adds nonlinearity and complexity, thereby strengthening security without incurring the computational overhead of S-boxes. We consider *SaI* as a grayscale image with 8-bit pixels and LFSR of length $n \geq 128$ across four stages. The inherent linearity of a baseline G-LFSR, despite its computational efficiency, renders it susceptible to attacks such as correlation cryptanalysis [4]. To address this vulnerability, we introduce a nonlinear filtering function *f* that is applied to a subset of the G-LFSR's internal state. This function obscures the linear structure and yields a more secure pseudo-random sequence *k*. Specifically; *f* must be at least a degree-3 nonlinear Boolean function, incorporating logical product terms to produce quadratic and cubic components. The generator polynomial is given as in formula (4):

$$g(x) = g_0 + g_1x + g_2x^2 + \ldots + g_{L-1}x^{L-1} + g_Lx^L \ (mod\ 2) \qquad (4)$$

where $g_0,,\ldots,g_{L-1} \in GF(2)$ represent the coefficients of generator of variable *x*, and $g_0 = g_L = 1$.*L* is the order of the polynomial, which represents the number of G-LFSR stages.

$$g(x) = x^3 + x^1 + 1 \qquad (5)$$

For generating the key, which is a pseudo random sequence denoted as *n*, we consider a primitive polynomial of order $L = 3$ and $n = 2^{607}$-1, then the polynomial given in the formula (5), is irreducible, meaning it cannot be factored over $GF(2)$. This is a crucial property that ensures the cryptographic safety of the generator [11] [12]. In this paper, we consider the length of G-LFSR structure as at least 607 bits to produce a binary pseudo random sequence. To this aim, the primitive polynomial is for instance as in formula (6).

$$g(x) = x^{607} + x^{105} + 1 \qquad (6)$$

Assuming the initial state is never allowed to be the all zero state. According to the formula (6), the key generator can produce a maximum-length pseudo-random sequence of period $2^{607} - 1$. Thus, the encryption function is a SPN which is composed of three ordered stages: 1) performing the permutation of pixels related to plaintext image; 2) XOR each bit of 128-bit plaintext with a 607-bit pseudo-number sequence to encrypt a data; 3) performing confusion and diffusion operations. To introduce confusion, we apply a dynamic bit rotation to each pixel value (of the plaintext image) based on the pseudo-random key k, followed by a modified XOR operation. The proposed n-bit grouped operations model may significantly improve computational efficiency while preserving cryptographic security. Moreover, to manage the keystream sequences generated by the proposed G-LFSR generator, we propose a strict one-time policy for the keystream, where each key segment employed in the encryption process is labelled as consumed and permanently invalidated to prevent any potential reuse. The keystream sequences management maintains a binary state vector that tracks the usage status of each key segment (where "1" indicates an unused segment and "0" otherwise). This operation is performed in real-time during the encryption process. The proposed cryptosystem also implements a secure garbage collection mechanism that continuously erases consumed key segments from the memory, ensuring they cannot be recovered or inadvertently redeployed.

V. Implementation and Security Performances Evaluation

The work presented in this paper is fully conducted as part of a real-project of CubeSat platform, which is part of the national program of the Algerian Space Agency. Satellite imagery was sourced from the Landsat platform, comprising over than 1,000 images (size varies from 1 to 300 MB). Encryption experimentation were conducted using MATLAB on 3.00 GHz

Intel Core i5-4590S CPU with 4 GB RAM under Windows 7 Professional. Computational performance was measured on three grayscale images, and the security robustness of our proposed algorithms was evaluated against standard criteria [7][11], including statistical, differential, and entropy attack analyses.

A. Computational Time

In the context of constrained onboard resource, the encryption processing time of imagery cryptosystem impacts the onboard system's efficiency, reliability, and mission success. Time in the experimentation set is in second. Fig.2-part a) displays the time processing of key sequences generation and keystream encryption processing. Regardless of the imagery size, the time processing of the key sequences of generation processing remains almost constant (0.9 second). This demonstrates the robustness of the generator. Furthermore, the encryption time naturally increases slightly with the size of the imagery. AES is renowned for its robustness and efficiency. We compared the processing time of our proposed cryptosystem to a streamlined implementation of AES-128 CTR mode. Fig.2-part b) displays the results of time encryption of our approach and the AES-128 CTR mode. In the proposed approach, time increases relatively proportionally with image size. Compared to AES-128 the percentage of performance gain can reach up to 60% for image sizes exceeding 200 MB.

B. Statistical attack Analysis

The encryption function aims at first one to reduce the correlation between adjacent pixels (i.e., in horizontal, vertical and diagonal directions), and the smaller the correlation, the better the encryption effect, the higher the security. Fig.3 shows that the correlations of the plaintext image img_3 and encrypted version through G-LFSR-based lightweight encryption algorithm. Parts a), b) and c) of Fig.3 (which respectively correspond to horizontal, vertical and diagonal directions) show that there is a strong correlation between adjacent pixels of plaintext image img_3, showing a linear relationship, and for encryption image) of Fig. 3, this correlation is greatly weakened, showing a strong randomness. Gray image histogram is also an important feature in security analysis. It reflects the distribution of gray level in the image and describes the number of pixels of each gray level in the image. Figure 4 illustrates the following: panel (a) displays the ciphertext of img_3; panel (b) presents the histogram of the plaintext image; and panel (c) shows the histogram of the ciphertext. The ciphertext histogram is nearly uniform, in stark contrast to that of the plaintext image. The probability distribution of the encrypted image closely approximates a uniform distribution, thereby revealing no exploitable statistical patterns. Consequently, the proposed fast G-LFSR-based encryption algorithm exhibits strong resistance to statistical attacks.

C. Differential attack Analysis

In order to analysis the resistance of our proposed algorithms against the differential attack [7], we test the effect of changing a single pixel in the original image on the encrypted image. Then, we measured two metrics: *NPCR* that measures the number of pixels change rate, and *UACI*, which measures the unified average changing intensity. Table III shows the resultants of these two metrics. With respect to ideal values of *NPCR* (i.e., 99%) and *UACI* (i.e., [33% 30%]), the proposed fast encryption algorithm satisfies security requirements.

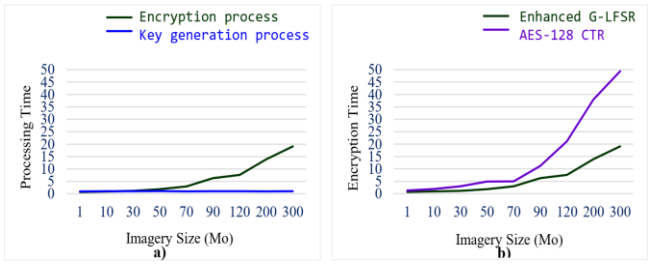

Fig. 2. Encryption Computational Time of the proposed algorithm and AES

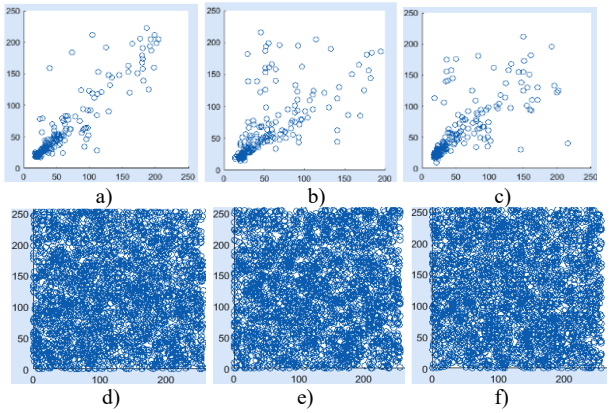

Fig. 3. Correlation Analysis of Adjacent pixels in original img_3 and ciphertext obtained using G-LFSR fast encryption algorithm; a) horizontal adjacent pixels; b) vertical adjacent pixels, c) diagonal adjacent pixels; d), e) and f) correlations of ciphered image corresponding to a), b), c) respectively.

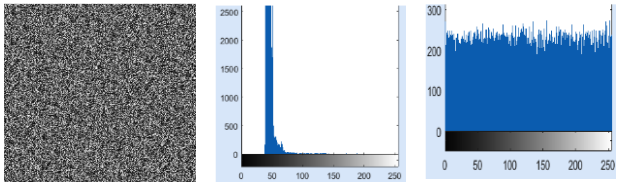

Fig. 4. Histogram analysis. a) ciphertext image of img_3; b) the histogram of plaintext image img_3; c) the histogram of ciphertext image of img_3 by G-LFSR fast encryption

D. Entropy attack Analysis

In order to test the resistance of the encryption algorithms against the entropy attack, we compute the entropy of the encryption images. Recall that when entropy value of encryption image is equal to zero, it means there is no uncertainty about the signal prediction. When entropy of encryption image has its maximum value (i.e., 8), it means that all of the symbols of the signal (i.e., the pixel levels) are equally likely to occur. Therefore, it can be said that higher the entropy better is the performance of security

algorithm. To measure the entropy for the encrypted image, Formula (1) is applied. Results reported in Table IV, demonstrate that encryption images are very close to the ideal value of 8. Then, both proposed algorithms are also secure against entropy attack. With the respect of the above experiments results, we believe that the proposed algorithms may not pose a threat to the security in a traditional way. While these performances give the security requirements proofs, it still not derisory to analyse the proposed algorithms by using more advanced cryptanalysis techniques as yoyo and boomerang [12].

TABLE III. *NPCR* AND *UACI* METRICS TEST

	img_1	img_2	img_3
NPCR (%) AES CTR-128	99.61	99.61	99.60
UACI (%) AES CTR-128	33.31	32.39	32.59
NPCR (%) Enhanced G-LFSR	99.53	99.63	99.31
UACI (%) Enhanced G-LFSR	33.37	30.31	30.13

TABLE IV. *ENTROPY* EVALUATION

	img_1	img_2	Img_3
Entropy AES-128 CTR	7.9999	7.9999	7.9999
Entropy enhanced G-LFSR	7.9996	7.9969	7.9996

E. Brute force attack Analysis

An exhaustive attack on the enhanced G-LFSR algorithm involves testing all possible initial states and feedback polynomials. For the initial state s_0, the seed must begin with a non-zero state, as an all-zero state produces no output and locks in zero. These constraint leaves $2^{607} - 1$ valid initial state. The polynomial, $g(x)$, must be primitive to generate a sequence with a maximal period of $2^{607} - 1$. The total number of such primitive polynomials is defined as: $N_{g(x)} = \frac{\phi(2^n-1)}{n}$, where ϕ is Euler's totient function. Then, the total Key Space for an exhaustive attack is: $E_{serach} = (2^n - 1) \times N_{g(x)}$. Then, in our case, $n = 607$, $E_{serach} \approx 2^{606}/607$ makes exhaustive attacks computationally infeasible. Indeed, testing all possible keys would require an impractical amount of computation. The observation is that large enhanced G-LFSR, with large n, ensures robust security against exhaustive attacks. Otherwise, The NIST SP800-22 standards is designed specifically to assess the randomness of binary sequences generated by random or pseudorandom number generators for cryptographic applications. Each test result is expressed as a ρ-value, which represents the probability that a perfect pseudo-random sequence generator would produce a sequence less random than the tested sequence. This variable follows a uniform distribution in [0, 1]. If ρ-value is <0.01, then it is concluded that the sequence is non-random.

VI. CONCLUSION

In this paper, we presented a security model for SSEO that enables the hierarchical ranking of security specifications for satellite imagery. In alignment with the assessed security importance, we proposed an encryption-optimized protocol incorporating a fast, lightweight encryption algorithm based on enhanced G-LFSR. The protocol controls the dynamic sensitivity classification of the satellite imagery set by integrating metrics such as entropy and correlation. This ensures that imagery set is classified not only by static rules but also by the intrinsic properties of the data, allowing for more nuanced and context-aware **c**ontrol of its security. Our approach meticulously controls the selection of the encryption algorithm by leveraging the security class and importance function computed for each satellite imagery. The proposed lightweight encryption method not only meets stringent security requirements—demonstrating robust resistance to cryptanalytic attacks—but also enhances the efficient utilization of on-board resources. It is designed with a keen awareness of the resource constraints and limited capabilities inherent in SSEO systems. As future work, we plan to extend the protocol by incorporating an accelerated physical process on FPGA and a remote-control mechanism to predict and enhance encryption quality in response to external spatial parameters such as radiation.

References

[1] A. Cratere, et al. "On-Board Computer for CubeSats: State-of-the-Art and Future Trends", *IEEE Access*, vol. *12*, pp. *99537–99569*, 2024.

[2] M. Manulis, C. P. Bridges, R. Harrison, V. Sekar, and A. Davis, ''Cyber security in new space: Analysis of threats, key enabling technologies and challenges,'' *Int. J. Inf. Secur.*, vol. *20*, no. *3*, pp. *287–311*, 2021.

[3] C. Paar, et al. "The Advanced Encryption Standard (AES)". *Understanding Cryptography*, 2024

[4] P. L'Ecuyer, (2012), "*Random number generation*" (pp. 35-71).

[5] A. Bogdanov, et al. "ALE: AES-based lightweight authenticated encryption". In *Fast Software Encryption: 20th Inter. Workshop, FSE 2013, Singapore, 2013. Revised Selected Papers 20* (pp. 447-466).

[6] J. Chen, et al. "Cryptanalysis of a DNA-based image encryption scheme". *Information Sciences*, vol *520*, pp. *130-141*, 2020.

[7] Y. Alghamdi, A. Munir, "Image encryption algorithms: a survey of design and evaluation metrics", *Journal of Cybersecurity and Privacy*, vol. 4, no.*12*, p.3917, 2024.

[8] M. Youssef, et al. "Enhancing satellite image security through multiple image encryption via hyperchaos, SVD, RC5, and dynamic S-Box generation", *IEEE Access*, 2024.

[9] B. Zhang, et al. "Chaos-based image encryption: Review, application, and challenges", *Mathematics,* vol. *11*, no. *11*, p. 2585, 2023.

[10] H. Hirata, et al, "All You Need Is Fault: Zero-Value Attacks on AES and a New λ-Detection M&M", *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol. *2024*, no. *1*, pp.*133-156*, 2024.

[11] C. Li, Y. Zhang, E. Y. Xie (2019). "When an attacker meets a cipher-image in 2018: A year in review". *Journal of Information Security and Applications*, vol. *48*, 102361.

[12] M. Rahman, et al. "Boomeyong: Embedding Yoyo within Boomerang and its Applications to Key Recovery Attacks on AES and Pholkos" *IACR Transactions on Semmetric Cryptology*, 2021.