Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 10, No. 01, March 2024.

TO DISTRIBUTE THE ABNORMAL EVENT USING CM-DDS ALGORITHM, WITH SECURELY USING RSA-IVDE ALGORITHM

Kande Archana and V. Kamakshi Prasad

(Received: 25-Oct.-2023, Revised: 18-Dec.-2023, Accepted: 1-Jan.-2024)

ABSTRACT

Automating the secure distribution and storage of video data addresses inefficiencies in manual processes. The emphasis on secure distributed storage aligns with the increasing importance of safeguarding video data. Traditional approaches in distributed video-data storage face limitations, like lack of scalability, data duplication, privacy and time complexity challenges in surveillance systems and suboptimal storage efficiency. This research addresses the existing limitations in distributed storage for video data by proposing a Central Moment-driven Distributed Structure (CM-DDS) and an incremental RSA-driven Incremental Video Data Encryption (RSA-IVDE) method. The framework optimizes video-data distribution across nodes, enhancing redundancy and fault tolerance for continuous data availability, even in the event of node failures. Emphasizing simplicity and accessibility, the user-friendly interface facilitates efficient management of large video datasets across various applications, including social networks, video games, surveillance and transportation networks. This comprehensive approach aims to provide a robust solution for secure distributed storage and transmission of video data while addressing the challenges identified in the existing research. The proposed approach surpasses other traditional methods with minimal decryption mean time (0.208s), minimal encryption mean time (0.182s) and minimal distribution mean time (1.620s).

KEYWORDS

Access control, Automation, Distributed storage, Scalability, Security, Video data.

1. INTRODUCTION

In today's world, the widespread use of video data has surged significantly, establishing itself as a valuable asset across various industries, particularly in security, surveillance, media and entertainment. This increasing prevalence of video data underscores the need for secure storage and efficient management practices. The exponential increase in video data poses challenges in storage and management, especially in distributed environments with complexities related to data scale, security and accessibility standards [1].

The realm of video-data management is inherently tied to the challenge of dealing with the sheer magnitude of data generated daily. This surge is fueled by technological advancements, consumer demands and the ubiquity of devices equipped with high-quality cameras [2]. As video data becomes integral to various aspects of modern life, including surveillance, filmmaking and social media, the demand for robust infrastructure has surged. Such infrastructure must not only accommodate the influx of video data, but also ensure its security, seamless accessibility and efficient organization. To address the challenges linked with video data storage, a nuanced understanding of the distributed-storage environment becomes crucial.

The inherent complexities of distributed storage systems, where data is spread across multiple nodes or locations, necessitate innovative solutions to optimize efficiency while safeguarding against security threats and ensuring timely access to critical data. This intricate balance between security, accessibility and efficiency underscores the contemporary challenges faced in managing video data within distributed-storage frameworks [3]-[4].

Turning to existing approaches in the realm of video-data management, various methodologies have emerged to tackle the challenges of secure and efficient storage in distributed environments. One of the prevalent approaches involves the use of advanced compression algorithms tailored for video data.

K. Archana and V. K. Prasad are with CSE, Jawaharlal Nehru Technological University (JNTU), Hyderabad, Telangana-500085, India.
Emails: kande.archana@gmail.com and kamakshiprasad@jntuh.ac.in

These algorithms play a crucial role in mitigating the storage burden by reducing the size of video files without compromising content quality. However, finding the right balance between compression ratios and video quality remains a nuanced challenge. Another significant avenue of exploration revolves around distributed-storage architectures, like cloud-based solutions. Cloud storage platforms offer scalability and flexibility, allowing organizations to adapt to the dynamic nature of video-data requirements. Additionally, cloud-based solutions often integrate robust security measures and facilitate seamless accessibility, making them increasingly popular in various sectors reliant on video data [5]-[6].

Furthermore, the integration of Artificial Intelligence (AI) [7] and Machine Learning (ML) [8] and encryption technologies [9]-[10] has marked a transformative paradigm in video-data management. AI-powered algorithms contribute to automated video analysis, content tagging and real-time monitoring, streamlining the storage process and enhancing the overall utility of video data. The multi-layered landscape of video-data management demands a comprehensive exploration of these diverse approaches to effectively navigate and address the associated complexities. Current video-data storage and management approaches face limitations in scalability, data duplication, security, access latency and time complexity. Ongoing research is needed to address these inaccuracies and develop more robust solutions capable of adapting to the challenges of video-data and distributed-storage environments. The major contributions of the research work are highlighted below.

- This research work integrates a CM-DDS for video frames, enhancing the framework's efficiency in managing and securing video data through statistical methods. CM-DDS excels by offering enhanced statistical representation, increased sensitivity to distribution nuances, adaptability to diverse datasets and robustness to outliers.
- It introduces an incremental RSA-IVDE method, ensuring enhanced security through the utilization of incremental RSA mechanisms. RSA-IVDE's incremental encryption ensures enhanced security and efficiency, adapting well to evolving data, supporting real-time encryption and facilitating efficient key management.
- It optimizes the distribution of video data across multiple nodes to enhance the framework's redundancy and fault tolerance. Strategies are implemented to ensure continuous data availability, even in the event of node failures.
- Finally, it emphasizes simplicity and accessibility in the framework's design, featuring a user-friendly interface for efficient management of large volumes of video data by diverse users and organizations.

The research paper is organized as follows. Section 2 reviews recent research, identifying the limitations of foundational methods for data distribution and cryptography. Section 3 proposes solutions, including an automated framework for secure distributed storage. Section 4 presents results and discussions on the framework's performance. Section 5 concludes the research paper, emphasizing its contributions and suggesting future research.

2. RECENT RESEARCH REVIEWS

After establishing the research context and introducing the baseline methods, the subsequent section delves into the latest enhancements made beyond these initial approaches.

In recent years, there has been significant research interest in the field of distributed storage for video data, with several works addressing the challenges in this domain. This sector postured numerous concerns that require attention. Yu et al. [11] have introduced two distinct architectures for the decentralized storage of surveillance video data; namely, "Centipede: Leveraging the Distributed Camera Crowd for Cooperative Video Data Storage" and "A Distributed Storage System for Robust, Privacy-Preserving Surveillance Cameras." The first work focused on a system that collaboratively stored and managed video data, leveraging the crowdsourcing capabilities of distributed cameras. It employed a hierarchical storage structure and data redundancy to ensure trustworthiness and scalability. The second work advocated for a decentralized storage system to protect individuals' constitutionally protected right to privacy when using surveillance cameras. It ensured the confidentiality of camera positions and operator identities while providing a secure means for data transmission. Wu et al. [12] proposed HuaVideo as a method to deliver secure, scalable and compatible HTML5 video content. This hybrid design combines client-server and peer-to-peer models, incorporating a novel access control

method to restrict video-content access to authorized users. Mar et al. [13] presented a private cloud storage system addressing privacy and security issues in cloud-based storage. The proposed system utilized encryption and a decentralized architecture to enhance fault tolerance and scalability. Ma et al. [14] suggested dynamic video switching for a distributed crane-control system, emphasizing bandwidth and resource utilization issues. The developed method adapted to workload changes and resource availability variations. The literature review provided an overview of recent research on secure data storage and transmission for video, spanning various applications. Ambani and Atkotiya [15] introduced a robust data-mining technique, ensuring user privacy in social networks. Besancon et al. [16] explored blockchain interoperability to enhance data flow in video games. Bálint [17] proposed decentralized methods for video storage based on blockchain technology. Du and Li [18] employed visual cryptography for privacy protection in video surveillance. Mishra and Mukhopadhyay [19] suggested a strategy for secure video storage and transmission involving encryption and compression. Jacob and Rekha [20] presented a secure file-sharing method using erasure correction code and deduplication to ensure accessibility and reduce storage space. Fitwi et al. [21] proposed a blockchain-based privacy-protection system for intelligent surveillance, prioritizing a decentralized approach. Chen et al. [22] introduced cooperative caching for popular videos among mobile users to enhance service quality and reduce network congestion. Ling et al. [23] developed a distributed multilevel storage system for visual surveillance in intelligent transportation networks, offering scalability and efficiency. Harnal and Chauhan [24] examined covert cryptography approaches for multimedia cloud-storage systems, evaluating safety, performance and computational complexity. Khan et al. [25] introduced the blockchain resource constraints in the Internet of Medical Things (BIoMT) consortium architecture, incorporating blockchain hyperledger fabric for improved security, integrity and transparency in health-related transactions.

The reviewed papers provide diverse solutions for secure data exchange, privacy preservation and effective multimedia data storage in applications like surveillance, transportation networks and cloud environments. These solutions are presented in various forms throughout the papers. The subsequent part addresses the mathematical presentation of existing-system issues.

2.1 Foundational Method for Data Distribution

Following the establishment of the research context in the previous part, this part delves into the baseline method for data distribution. Assuming that the total video data, represented as $V[\,]$, comprises n video frames, each frame denoted as f_i, where $i = 1,2,\dots,n$, this is expressed in Equation (1) as follows:

$$V[] =< f_1, f_2, \dots . f_n > \tag{1}$$

Moreover, the set of frames, denoted as $F[]$, represents segmented image frames $fp\ s$ from the entire video data based on time t. This formulation is expressed in Equation (2) as follows:

$$F[] = \frac{d}{dt} V[], t = fps \tag{2}$$

Additionally, presuming that the assembly of storage clusters, denoted as $S[\,]$, accommodates multiple clusters with each cluster represented as S_i, where $i = 1,2,\dots,k$. Thus, for a total of k clusters, this is expressed in Equation (3) as:

$$S[] =< S_1, S_2, \dots S_k > \tag{3}$$

Ultimately, assuming that the arbitrary function λ is a first-come, first-served basis function for assigning frame sequences to storage clusters, this is expressed in Equation (4) as:

$$\lambda\{F[] :: S[]\} \rightarrow \{ f_i \rightarrow S_i \} \tag{4}$$

Therefore, according to the baseline method, the ultimate allocation of $f_i \rightarrow S_i$ is achieved through the distribution key. In the subsequent sections of this study, the recent enhancements made to the baseline methods are delved into.

2.2 Foundational Method for Cryptography

This sub-section provides an overview of the traditional cryptography method. Assuming that a given

frame, f_x is encrypted or decrypted using a specified key referred to as "key," the resulting encrypted pixel matrix, $P_x[][]$ is expressed in Equation (5). In this sub-section, the traditional cryptography method is furnished.

$$P_X[][] = \sum_{i=1}^{n}\sum_{j=1}^{m}\{f_X[i][j] \oplus Key\} \tag{5}$$

Therefore, $P_x[][]$ represents the ultimate encrypted pixel matrix. By applying the reverse operation to the encrypted data, the pixel matrix is subsequently decrypted. In the following sub-sections, the recent advancements of the baseline methods are delved into.

2.3 Problem Formulation – Mathematical Model

After realizing the baseline methods and the current developments over the baseline techniques, in this sub-section of the work, the persistent research problems are outlined. The existing research in distributed storage for video data demonstrates a growing concern for security and privacy, leading to various proposed architectures and methods. However, these solutions exhibit limitations and gaps that need addressing. Challenges include the need for trustworthy and scalable frameworks, ensuring privacy in surveillance systems and optimizing storage efficiency for multimedia content. Despite the multitude of proposed techniques, a comprehensive analysis and comparison of their effectiveness based on safety, performance and computational complexity are lacking. Additionally, there is a need for validation and accuracy assessment of established systems using diverse datasets. This research is motivated by these gaps by providing a robust and efficient solution for secure distributed storage and transmission of video data, considering the diverse applications in social networks, video games, surveillance and transportation networks.

Moving forward, let's first address the data-duplication issue. Assuming that the frame collection consists of two distinct frames, f_1 and f_2; furthermore, when considering two separate storage clusters labeled S_1 and S_2, this results in the distribution possibilities shown in Equations (6), (7) and (8):

$$f_1 \rightarrow S_1 \tag{6}$$

and

$$f_2 \rightarrow S_2 \tag{7}$$

or,

$$f_1, f_2 \rightarrow S_1, S_2 \tag{8}$$

However, it is evident that the redundancy resulting from these distributions directly contributes to a higher space complexity. Secondly, it is assumed that the total time complexity for the retrieval operation of video data is T_1, formulated for n number of frames and k number of storage clusters, as denoted in Equation (9):

$$T_1 = n * k \tag{9}$$

This is reformulated by Equation (10),

$$T_1(n) = n^2 \mid n \approx k \tag{10}$$

Video-data encryption safeguards against unauthorized access, but presents challenges like computational complexity, compatibility issues and key management concerns. Sharing encryption keys between devices is difficult, leading to higher costs and performance overhead. Despite encryption, vulnerability to decryption attacks remains a risk.

Therefore, it is assumed that the time complexity O for distributed video data is T_2, formulated for ($n1$*$m1$) number of pixels per frame, a total of n frames and k number of distributions, then Equations (11) and (12) are formulated as:

$$T_2 = \{(n1 * m1) * n\} * k \tag{11}$$

or,

$$T_2(n) = O(n^4) \mid n1 \approx m1 \approx n \approx k \tag{12}$$

Further, the proposed solutions are discussed in the next section of this work.

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 10, No. 01, March 2024.

3. PROPOSED SOLUTIONS

After thoroughly examining the current advancements in research and identifying the limitations of traditional systems, this section presents the proposed solutions. The first method involves a hierarchical tree-storage structure driven by central moments for frames with distribution. The second method is an incremental RSA-driven approach for encrypting video data.

The proposed system employs a fundamental technique for image identification; namely, central-moment technique. Central moments are statistical measures widely utilized in image processing and computer vision to characterize the spatial distribution of pixel intensities within an image. Specifically, central moments offer a means to quantify the distribution of pixel intensities relative to the mean intensity of an image. These moments facilitate the computation of various image features, including the image centroid orientation and scale, all of which are essential for numerous applications like object recognition, tracking and image registration. Therefore, a thorough understanding of central moments and their properties are crucial for the effective analysis and manipulation of images across various domains.

Assuming that the moments' array $M[][]$ for the entire pixel matrix collection is represented by $P_x[][]$, the regions for segmented objects need to be initially separated from the rest of the image and the intensity I is expressed in Equation (13) as follows.

$$I = RGB\{P_X[][]\} \tag{13}$$

Further, the central moment of the frames is formulated by Equation (14),

$$M[][] = \sum_{i=1}^{n1}\sum_{j=1}^{m1} {}^{*I}_{*}\{(P_X[i][] - Mean(P_X[i][]))^{n1} * (P_X[][j] - Mean(P_X[][j]))^{m1}\} \tag{14}$$

Moreover, these moment values are adequate for uniquely identifying the frames and they are employed to construct the tree-storage structure as a distributed-storage system. Assuming that the storage tree $T[]$, is built using the arbitrary function τ, the relationship is expressed by Equation (15).

$$T[] = \tau\{M[][]\} \tag{15}$$

Assuming that the time complexity for data retrieval is T_3, this is formulated by Equations (16) and (17) as follows:

$$T_3 = n + \frac{n}{2} + \frac{n}{4} + \frac{n}{8} + ... + 1 \tag{16}$$

or,

$$T_3(n) = \log_2(n) \tag{17}$$

Secondly, the incremental encryption for video data is presented. Assuming two frames denoted as f_1 f_1 and f_2 and utilizing the arbitrary function Δ, the difference $D[]$, between these two frames in terms of data is calculated by Equation (18):

$$D[] = \Delta\{f_1, f_2\} \tag{18}$$

Further, the encrypted data, $E[]$, is built as in Equations (19) and (20),

$$E[] = RSA\{f_i\} + RSA\{D[i+1]\} \tag{19}$$

Naturally,

$$Size\{f_{i+1}\} >> Size\{D[i+1]\} \tag{20}$$

Therefore, the improvements are easily achieved as f_{i+1} and $D[i+1]$. Furthermore, building upon the proposed methods, the upcoming sub-section establishes the proposed algorithms.

3.1 Proposed Algorithms

Following a comprehensive understanding of the proposed methods, this sub-section introduces the proposed algorithms.

Algorithm - I: CM-DDS Algorithm

Input: Video data as *V*[]
Output: Tree Structure as *T*[]

Process:

1. Read in the video data as *V*[]
2. Calculate the overall frames in the video.
3. For each frame as *V*[*i*]
    1. Starting from the root node, each level of the tree will have twice as many nodes as the level above it.
    2. The height of the tree will be the minimum integer value that satisfies $2^h >=$ total number of frames.
    3. Initialize a binary tree data structure with the appropriate number of nodes for the given height.
    4. The nodes of the tree should be numbered in a left-to-right, top-to-bottom order.
4. For each frame as *V*[*j*]:
    1. The first frame will be stored in the root node.
    2. For each subsequent frame, traverse down the tree from the root node to the appropriate child node based on the frame index.
    3. Store the frame in the corresponding node.
5. Return the Tree Structure as *T*[]

Algorithm - II: RSA-IVDE Algorithm

Input: Tree Structure as *T*[]
Output: Encrypted data as *TE*[]

Process:

1. Select two large prime numbers, *p* and *q*
2. Compute *n*=*p***q*
3. Compute *I*(*n*)=(*p*-1)*(*q*-1)

encrypt ()

4. Select an encryption exponent, e, that is relatively prime to *I*(*n*)
5. Calculate the decryption exponent *d*
6. Convert the video data into a series of integers that represent the bytes of the video.
7. Divide the video data into blocks of a fixed size.
8. For each block
    1. Convert it to a single integer using a fixed-length encoding scheme.
    2. Encrypt each integer using the RSA algorithm:
    3. Raise the integer to the power of the encryption exponent *e*, modulo *n*
    4. Store the encrypted integers in a file or stream as *TE*[]

decrypt ()

9. Read the encrypted integers from the file or stream.
10. Decrypt each integer using the RSA algorithm
11. Raise the integer to the power of the decryption exponent d, modulo n
12. Convert each decrypted integer back to a block of bytes using the fixed-length decoding scheme.
13. Concatenate the decrypted blocks to reconstruct the original video data.

Firstly, the CM-DDS algorithm [26] is presented. Central moments, a set of statistical measures, offer insights into the distribution of pixel intensities in an image. They find application in image processing and computer vision, extracting valuable features like object size, shape and orientation. Central moments are particularly advantageous due to their invariance to translation, remaining consistent even if the image undergoes shifts or translations. The central moment of an image is defined as the expected value of a function of pixel intensities, with the function centred around the mean intensity of the image. Secondly, the RSA-IVDE algorithm [27] is introduced.

The RSA algorithm is a widely utilized public-key encryption method known for its high security and efficiency. It is commonly employed in applications requiring secure data transmission, including the encryption of image data. This article explores the application of the RSA algorithm for encrypting image data and outlines the steps involved in the encryption process (see Figure 1). The RSA algorithm employs two keys, a public key for encryption and a private key for decryption. The public key is distributed to anyone needing to send encrypted messages, while the private key remains confidential with the recipient.

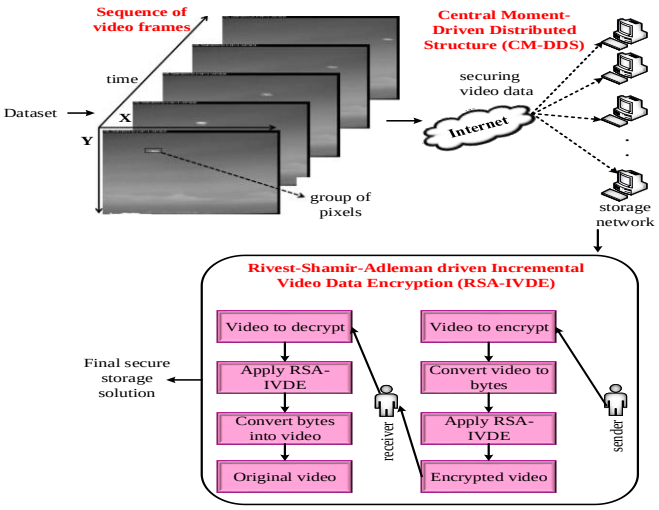


Figure 1. Proposed architecture.

To encrypt image data using RSA, the initial step involves converting the image data into a format compatible with the RSA algorithm. A common approach is to transform the image data into a series of binary values, where each pixel is represented by a binary value. This conversion is achieved through various techniques, like grayscale conversion or binary thresholding. Furthermore, the results obtained will be discussed in the upcoming section.

4. Results and Discussion

This proposed work develops algorithms for two primary purposes: data distribution and data cryptography. The UCF crime dataset [28] comprises a total of 1900 samples with 7247 average number of frames and 8 number of categories. Only 15 samples are listed for illustrative purposes. Subsequently, the formations of clusters and the read/write speed for each cluster are observed and presented in Table 1.

Data distribution involves the procedure of storing and transmitting data across multiple devices or locations to guarantee availability, reliability and performance. In a distributed system, data is replicated and dispersed among multiple nodes, enhancing fault tolerance, load balancing and scalability. Data distribution is crucial for various contemporary applications, including cloud computing, big data analytics and Internet of Things (IoT) devices.

Table 1. Data-distribution analysis.

Trial Seq.	Number of Clusters (#)	Read Speed (Mbps)	Write Speed (Mbps)
1	56	148.43	137.24
2	54	109.88	99.33
3	44	110.34	143.12
4	48	106.77	88.92
5	55	91.67	126.79
6	45	131.44	112.52
7	53	96.72	141.52
8	56	122.53	114.46
9	50	133.26	141.5
10	54	140.38	97.6
11	50	126.34	125.45
12	52	127.81	106.63
13	56	101.58	121.91
14	46	127.82	104.65
15	54	105.48	91.22

Distributing data across multiple nodes is an intricate process that involves addressing issues like data consistency, synchronization and security. To achieve efficient data distribution, various techniques like replication, sharding and partitioning are employed. Replication entails copying data to multiple nodes

to ensure redundancy and availability, while sharding and partitioning involve dividing data into smaller subsets and distributing them across different nodes. Efficient data distribution is essential for modern businesses and organizations that need to manage and process large volumes of data quickly and effectively. Proper data distribution enables organizations to achieve better performance, scalability and reliability, ensuring that data is available when and where needed.

The mean read speed for the entire dataset is 120 Mbps and the write speed is 110 Mbps. The results are also illustrated diagrammatically (see Figures 2 and 3). Additionally, the formed clusters undergo testing for data loss before and after distribution, with the recorded observations presented in Table 2.

Data Loss Prevention (DLP) is a set of strategies and technologies aimed at safeguarding sensitive data from loss, theft or misuse. DLP solutions typically involve identifying, classifying and monitoring sensitive data as it moves across the network, applying controls and policies to prevent unauthorized access or exfiltration.

Effective DLP helps organizations mitigate risks related to data breaches, compliance violations and reputational damage. By implementing DLP measures, organizations reduce the likelihood of data-loss incidents, minimize their impact and maintain customer trust and regulatory compliance. Common DLP strategies include encryption, access controls, activity monitoring, data classification and incident response planning. However, DLP is not a one-size-fits-all solution and organizations should tailor their approach to their specific needs and risk profile. A comprehensive DLP program requires collaboration across multiple stakeholders, including IT, security, legal and business teams.

Thirdly, the time taken for data distribution is recorded for further analysis (see Table 3). The mean data distribution time is 1.239 seconds for the entire dataset. Distributing data across multiple clusters is a crucial technique for storing and processing large volumes of data efficiently. By distributing data across multiple clusters, organizations improve performance, increase scalability and enhance data availability and reliability. The process of data distribution involves dividing large datasets into smaller subsets and distributing them across multiple clusters in a way that maximizes performance and minimizes data duplication. This is achieved through various methods like shredding, replication and partitioning.

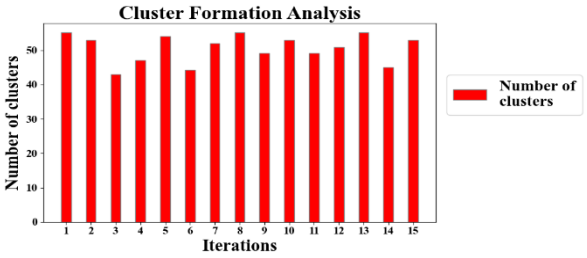


Figure 2. Cluster-formation analysis.

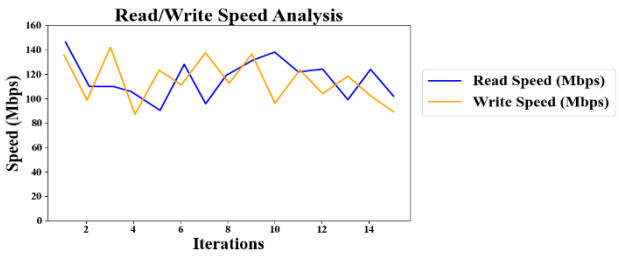


Figure 3. Read/write speed analysis.

Shredding entails breaking down datasets into smaller, more manageable subsets and distributing them across multiple clusters. Replication involves generating duplicates of datasets and storing them across multiple clusters to improve data availability and reliability. Partitioning involves segmenting datasets into smaller subsets and distributing them across multiple clusters based on specific criteria like data type, geography or user access.

Table 2. Data-distribution analysis.

Trial Seq.	Number of Clusters (#)	Data Loss (%)		Trial Seq.	Number of Clusters (#)	Data Loss (%)
1	56	0		9	50	0
2	54	0		10	54	0
3	44	0		11	50	0
4	48	0		12	52	0
5	55	0		13	56	0
6	45	0		14	46	0
7	53	0		15	54	0
8	56	0				

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 10, No. 01, March 2024.

In summary, distributing data over multiple clusters allows organizations to harness the advantages of distributed-computing and big-data technologies, facilitating high-performance data processing, analysis and storage. The results are also visually presented in Figure 4.

Furthermore, the results derived from the cryptographic process are observed and documented in the provided details, as presented in Table 3.

Table 3. Data cryptographic analysis.

Trial Seq.	Key Length	Encryption Time (Sec)	Decryption Time (Sec)	Data Distribution Time (Sec)
1	44	0.126	0.147	1.776
2	44	0.123	0.144	1.973
3	44	0.192	0.213	1.494
4	44	0.121	0.140	1.38
5	44	0.180	0.216	1.971
6	44	0.207	0.228	1.723
7	44	0.184	0.206	1.75
8	44	0.208	0.246	1.29
9	44	0.221	0.245	1.143
10	44	0.218	0.255	1.432
11	44	0.146	0.162	1.407
12	44	0.195	0.226	1.733
13	44	0.185	0.209	1.701
14	44	0.208	0.229	1.858
15	44	0.210	0.248	1.675

For the entire dataset, the mean encryption time is 0.182 seconds and the mean decryption time is 0.208 seconds. Data encryption and decryption are crucial processes employed to safeguard sensitive information from unauthorized access and theft. Encryption transforms plain text or other forms of data into a coded format, readable only by authorized users possessing the key or password needed for decryption. Decryption, on the other hand, reverses the encryption process, restoring the data to its original form for comprehension. Data encryption ensures the confidentiality, integrity and authenticity of information by rendering it unreadable to anyone lacking access to the decryption key. It is widely utilized in industries like finance, healthcare and government, where data privacy is of paramount importance.

Two primary categories of encryption exist: symmetric encryption and asymmetric encryption. Symmetric encryption employs a single key for both encryption and decryption, while asymmetric encryption uses distinct keys for these operations. While asymmetric encryption provides enhanced security, it also demands more computational resources compared to symmetric encryption. Decryption necessitates the use of the same key or password employed during encryption to decode the data. When a user inputs the correct password or key, the encrypted data is transformed back into its original form, allowing the user to comprehend the information. The result is observed graphically by Figure 5.

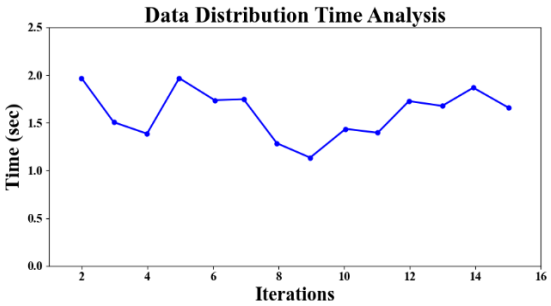


Figure 4. Data-distribution time analysis.

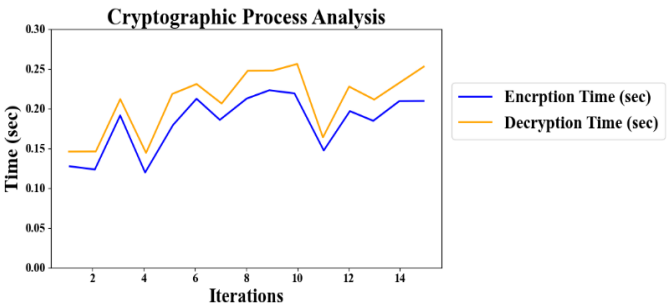


Figure 5. Encryption and decryption-time analysis.

Moreover, the proposed methodology has undergone rigorous testing, employing a diverse set of alternative datasets and the corresponding outcomes are meticulously detailed in Table 4.

Furthermore, a comparative analysis is conducted in the subsequent sub-section of this work, juxtaposing the acquired results with findings from parallel research endeavors.

Table 4. Test results on various datasets.

Dataset Name	Data Distribution Mean Time (Sec)	Encryption Mean Time (Sec)	Decryption Mean Time (Sec)
UMN [29]	1.776	0.192	0.213
UCSD [29]	1.75	0.121	0.140
Ped – 1 [29]	1.29	0.180	0.216
Ped – 2 [29]	1.143	0.207	0.228
BOSS [30]	1.432	0.184	0.206
Proposed (UCF crime)	1.407	0.208	0.246

4.1 Comparative Analysis

After conducting thorough analyses of the obtained results, this sub-section of the paper involves a comparison between the achieved outcomes and the corresponding benchmarked results, as illustrated in Table 5.

Table 5. Comparative analysis.

Ref.	Proposed Algorithm Name	No. of Segments (Mean) (#)	Mean Time for Distribution (Sec.)	Data Loss Percentage (%)	Mean Time for Encryption (Sec)	Mean Time for Decryption (Sec)
Yu et al. [11]	Centipede	34	6.11	7	3.293	4.734
Wu et al. [13]	HuaVideo	40	8.617	5	2.744	4.708
Mar et al. [14]	Secure personal cloud storage	33	6.998	5	2.936	2.339
Ambani et al. [16]	Data Mining Techniques	40	7.764	9	2.378	5.129
Besançon et al. [17]	Blockchain Interoperability	33	5.218	6	2.309	4.895
Jacob et al. [21]	Erasure	23	6.686	6	4.62	3.227
Proposed Method	Central Moment, Tree Storage Structure, RSA and Incremental backups	51	1.620	0	0.182	0.208

Thus, it becomes evident that the proposed method has surpassed a majority of the well-known benchmarked techniques in all aspects.

4.2 Case Study

Study 1: Data-distribution Analysis

In this comprehensive case study, the examination delves into the impact of different data distribution strategies—shredding, replication and partitioning—on the performance of a system managing large volumes of video data. The analysis involves six distinct scenarios: (1) lacking shredding, (2) devoid of replication, (3) without partitioning, (4) in the absence of both shredding and replication, (5) excluding both shredding and partitioning and (6) lacking both replication and partitioning. Each scenario explores how the absence of a specific distribution strategy affects data-management efficiency, fault tolerance and overall system performance. This study provides valuable insights into the nuanced interplay between these strategies and their role in optimizing data distribution for enhanced system robustness and reliability.

Scenario 1: Without Shredding

Intention: In this scenario, the CM-DDS framework is implemented without the use of shredding, which involves breaking down large datasets into smaller subsets.

Remark: While CM-DDS provides enhanced statistical representation and adaptability, the absence of

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 10, No. 01, March 2024.

shredding results in challenges related to efficient handling of extremely large video datasets. The framework experiences limitations in terms of processing speed and resource utilization.

Scenario 2: Without Replication

Intention: Replication, which involves creating copies of data, is excluded from the CM-DDS framework.

Remark: Without replication, the redundancy and fault tolerance of the system are compromised. In the event of node failures, the availability of video data could be at risk, leading to potential interruptions in data access and processing.

Scenario 3: Without Partitioning

Intention: Partitioning, the division of data into smaller subsets, is not employed in this scenario.

Remark: The absence of partitioning might lead to suboptimal distribution of video data across nodes. This could result in uneven workloads and might affect the overall efficiency of the system, especially when dealing with diverse datasets.

Scenario 4: Without Shredding and Replication

Intention: This scenario excludes both shredding and replication.
Remark: The absence of shredding and replication results in a less robust system. Without shredding, large datasets pose processing challenges and without replication, the system lacks the necessary redundancy to ensure continuous data availability.

Scenario 5: Without Shredding and Partitioning

Intention: This scenario excludes both shredding and partitioning.

Remark: The lack of shredding and partitioning leads to difficulties in handling large volumes of video data efficiently. The absence of partitioning results in suboptimal data distribution across nodes, impacting the overall performance of the CM-DDS framework.

Scenario 6: Without Replication and Partitioning

Intention: Replication and partitioning are both excluded from the CM-DDS framework in this scenario.

Remark: Without replication, the system lacks the necessary redundancy and without partitioning, the distribution of video data is not optimized. This could lead to challenges in fault tolerance, workload distribution and overall system performance. The data-distribution time for different scenarios are depicted in Table 6.

Table 6. Data-distribution time in different scenarios.

Trial seq.	Data-distribution time (s)					
	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Scenario 5	Scenario 6
1	2.012	2.776	2.899	2.825	2.812	2.839
2	2.189	2.973	2.075	2.001	2.003	2.018
3	2.805	2.494	2.612	2.535	2.525	2.554
4	2.634	2.38	2.507	2.424	2.418	2.446
5	2.175	2.971	2.059	2.996	2.997	2.017
6	2.962	2.723	2.828	2.779	2.776	2.792
7	2.989	2.75	2.857	2.806	2.804	2.819
8	2.511	2.29	2.401	2.349	2.347	2.365
9	2.364	2.143	2.255	2.202	2.198	2.215
10	2.653	2.432	2.544	2.485	2.472	2.493
11	2.628	2.407	2.519	2.461	2.448	2.467
12	2.964	2.733	2.855	2.796	2.782	2.803
13	2.932	2.701	2.823	2.746	2.751	2.772
14	2.089	2.858	2.98	2.992	2.908	2.932
15	2.906	2.675	12798	2.738	2.726	2.748

These scenarios represent variations in the data-distribution times based on the absence of specific strategies (shredding, replication, partitioning). The values are indicative and vary based on the specific characteristics of each scenario.

Study 2: Key-length Analysis

This comprehensive case study investigates the influence of varying key lengths in a trial sequence 15, on the encryption and decryption processes of a cryptographic system. The analysis encompasses fifteen distinct trials, each with a different key length within the specified range. Table 7 explains the encryption and decryption times with respect to key length.

Table 7. Encryption and decryption times.

Trial seq.	Key length	Encryption time (s)	Decryption time (s)
15	30	0.255	0.277
15	32	0.250	0.272
15	34	0.345	0.269
15	36	0.340	0.264
15	38	0.340	0.262
15	41	0.335	0.258
15	42	0.330	0.254
15	45	0.325	0.349
15	47	0.325	0.348
15	50	0.320	0.343

The encryption and decryption times corresponding to 15-trial sequence are meticulously recorded, shedding light on the performance characteristics and computational efficiency associated with different key lengths. This study aims to discern patterns, trends and trade-offs in the cryptographic processes as the key length undergoes variation, providing valuable insights into optimizing the balance between security and computational overhead in cryptographic applications.

5. CONCLUSION

The critical gaps in existing distributed-storage solutions for video data are addressed by this research. Innovative contributions addressed challenges, enhancing video-data management through CM-DDS integration for improved representation and adaptability. The introduction of RSA-IVDE ensured heightened security and efficiency with incremental encryption, adaptable to evolving data and supporting real-time encryption. The optimization of video-data distribution across nodes improved redundancy and fault tolerance, ensuring continuous data availability. The emphasis on simplicity and accessibility in the framework's design provided an efficient solution for managing large video datasets across diverse applications. Overall, a comprehensive and robust approach to secure distributed storage and transmission of video data was contributed by this research, addressing current limitations and meeting the varied needs of users in social networks, video games, surveillance and transportation networks. The proposed approach excels in speed and efficiency, with minimal decryption (0.208s), encryption (0.182s) and distribution (1.620s) times, marking a significant advancement compared to traditional methods. These results underscore its potential to outperform existing approaches in data processing and distribution. Future research focuses on collaborating with environmental experts and integrating eco-friendly practices to enhance the ecological impact of secure distributed-storage systems, ensuring their sustainability for long-term use. This includes exploring energy-efficient strategies, incorporating renewable-energy sources and implementing environmentally conscious design principles to minimize the environmental footprint of such systems.

REFERENCES

[1] D. Dhingra and M. Dua, "Medical Video Encryption Using Novel 2D Cosine-sine Map and Dynamic DNA Coding," Medical & Biological Engineering & Computing, vol.62, pp. 237-255, 2023.
[2] S. Das, A. Adhikary, A. A. Laghari and S. Mitra, "Eldo-care: EEG with Kinect Sensor Based Telehealthcare for the Disabled and the Elderly," Neuroscience Informatics, vol. 3, no. 2, p. 100130, 2023.
[3] N. Cecchinato, A. Toma, C. Drioli, G. Oliva, G. Sechi, G.L. Foresti, "Secure Real-time Multimedia Data Transmission from Low-cost UAVs with a Lightweight AES Encryption," IEEE Communications

Magazine, vol. 61, no. 5, pp. 160–165, DOI:10.1109/mcom.001.2200611, 2023.

[4] H. N. Khan, A. Das and A. Chaudhuri, "Unique Video Encryption Technique Intended for Smart City Application," IETE Journal of Research, vol. 69, no. 9, pp. 5830–5839, 2023.

[5] P. Saini and K. Kumar, "S-method: Secure Multimedia Encryption Technique in Cloud Environment," Multimedia Tools and Applications, DOI: 10.1007/s11042-023-15600-w, 2023.

[6] Y. Fouzar, A. Lakhssassi and M. Ramakrishna, "A Novel Hybrid Multikey Cryptography Technique for Video Communication," IEEE Access, vol. 11, pp. 15693–15700, 2023.

[7] S. Cheikh Youssef et al., "Video Labelling Robot-assisted Radical Prostatectomy and the Role of Artificial Intelligence (AI): Training a Novice," Journal of Robotic Surgery, vol. 17, no. 2, pp. 695–701, DOI:10.1007/s11701-022-01465-y, 2022.

[8] Z. Zhou and Y. Liu, "Political Teaching Application in High Vocational Care Courses Based on Machine Learning Systems," Soft Computing, vol. 27, no. 11, pp. 7657–7666, 2023.

[9] A. Ayub Khan et al., "BIoMT: A State-of-the-art Consortium Serverless Network Architecture for Healthcare System Using Blockchain Smart Contracts," IEEE Access, vol. 10, pp. 78887–78898, DOI:10.1109/access.2022.3194195, 2022.

[10] A. Ayub Khan et al., "Internet of Things (IoT) Security with Blockchain Technology: A State-of-the-art Review," IEEE Access, vol. 10, pp. 122679–122695, DOI:10.1109/access.2022.3223370, 2022.

[11] J. Yu, H. Chen, K. Wu, T. Zhou, Z. Cai and F. Liu, "Centipede: Leveraging the Distributed Camera Crowd for Cooperative Video Data Storage," IEEE Internet of Things Jour., vol. 8, no. 22, pp. 16498–16509, 2021.

[12] Y. Wu, Y. Zhang, C. Wang, C. Li and C. Xing, "HuaVideo: Towards a Secure, Scalable and Compatible HTML5 Video Providing System," Proc. of the 2014 11th IEEE Web Information System and Application Conf., DOI:10.1109/wisa.2014.23, Tianjin, China, 2014.

[13] K. K. Mar, C. Y. Law and V. Chin, "Secure Personal Cloud Storage," Proc. of the 2015 10th IEEE Int. Conf. for Internet Technology and Secured Transactions (ICITST), pp. 108-113, DOI:10.1109/icitst.2015.7412068, London, UK, 2015.

[14] R. Ma, Q. Wang, Y. Zhang, J. Chen and J. Zhou, "A Dynamic Video Switching Mechanism for Distributed Crane Control System," Proc. of the 2021 IEEE Intl. Conf. on Dependable, Autonomic and Secure Computing, Intl. Conf. on Pervasive Intelligence and Computing, Intl. Conf. on Cloud and Big Data Computing, Intl. Conf. on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/ CyberSciTech), AB, Canada, DOI:10.1109/dasc-picom-cbdcom-cyberscitech52372.2021.00050, 2021.

[15] D. Ambani and K. Atkotiya, "Secure Data Contribution and Retrieval in Social Networks Using Effective Privacy Preserving Data Mining Techniques," Proc. of the 2021 IEEE Int. Conf. on Computing, Communication and Green Engineering (CCGE), pp. 1-5, DOI:10.1109/ccge50943.2021.9776454, Pune, India, 2021.

[16] L. Besancon, C. F. Silva and P. Ghodous, "Towards Blockchain Interoperability: Improving Video Games Data Exchange," Proc. of the 2019 IEEE Int. Conf. on Blockchain and Cryptocurrency (ICBC), pp. 81-85, DOI:10.1109/bloc.2019.8751347, Seoul, S. Korea, 2019.

[17] K. Balint, "Modern, Decentralized Blockchain-based Solutions for Saving Video Footage," Proc. of the 2020 IEEE 18th International Symposium on Intelligent Systems and Informatics (SISY), pp. 11-14, DOI:10.1109/sisy50555.2020.9217100, Subotica, Serbia, 2020.

[18] L. Du and Y. Li, "Privacy Preserving for Human Object in Video Surveillance *via* Visual Cryptography," Proc. of the 2014 IEEE Int. Conf. on Security, Pattern Analysis and Cybernetics (SPAC), pp. 80-85, DOI:10.1109/spac.2014.6982661, Wuhan, China, 2014.

[19] V. Shukla and A. Mishra, "A New Sequential Coding Method for Secure Data Communication," Proc. of the 2020 IEEE Int. Conf. on Computing, Power and Communication Technologies (GUCON), pp. 529-533, DOI:10.1109/gucon48875.2020.9231252, Greater Noida, India, 2020.

[20] C. Jacob and V. R. Rekha, "Secured and Reliable File Sharing System with De-duplication Using Erasure Correction Code," Proc. of 2017 IEEE Int. Conf. on Networks & Advances in Computational Technologies (NetACT), pp. 221-228, DOI:10.1109/netact.2017.8076770, Thiruvananthapuram, India 2017.

[21] A. Fitwi, Y. Chen and S. Zhu, "A Lightweight Blockchain-based Privacy Protection for Smart Surveillance at the Edge," Proc. of the 2019 IEEE Int. Conf. on Blockchain (Blockchain), pp. 552-555, DOI:10.1109/blockchain.2019.00080, Atlanta, USA, 2019.

[22] Y.-T. Chen, C.-C. Yen, Y.-T. Lin and J.-S. Wang, "Cooperative Caching Plan of Popular Videos for Mobile Users by Grouping Preferences," Proc. of the 2018 IEEE 16th Intl. Conf. on Dependable, Autonomic and Secure Computing, 16th Intl. Conf. on Pervasive Intelligence and Computing, 4th Intl. Conf. on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech), pp. 762-769, DOI: 10.1109/dasc/picom/datacom/cyberscitec.2018.00132, Athens, Greece, 2018.

[23] C. W. Ling, A. Datta and J. Xu, "A Case for Distributed Multilevel Storage Infrastructure for Visual Surveillance in Intelligent Transportation Networks," IEEE Internet Computing, vol. 22, no. 1, pp. 42–51, DOI:10.1109/mic.2017.265102316, 2018.

[24] S. Harnal and R. K. Chauhan, "Comparison for Confidential Cryptography in Multimedia Cloud

Environment," Proc. of the 2018 5th IEEE Int. Conf. on Parallel, Distributed and Grid Computing (PDGC), pp. 148-152, DOI:10.1109/pdgc.2018.8745909, Solan, India, 2018.
[25] A. A. Khan et al., "A Blockchain Security Module for Brain-computer Interface (BCI) with Multimedia Life Cycle Framework (MLCF)," Neuroscience Informatics, vol. 2, no. 1, p. 100030, DOI: 10.1016/j.neuri.2021.100030, 2022.
[26] M. Yin, M. Wang, Y. Huo and Z. Rao, "Simulation of Solid-liquid Phase Change at Pore Scale Using Lattice Boltzmann Method with Central Moments in Thermal Energy Storage," Journal of Energy Storage, vol. 49, p. 104116, DOI: 10.1016/j.est.2022.104116, 2022.
[27] N. A. Ugochukwu et al., "An Innovative Blockchain-based Secured Logistics Management Architecture: Utilizing an RSA Asymmetric Encryption Method," Mathematics, vol. 10, no. 24, p. 4670, DOI: 10.3390/math10244670, 2022.
[28] J. Tang et al., "Sae-PPL: Self-guided Attention Encoder with Prior Knowledge-guided Pseudo Labels for Weakly Supervised Video Anomaly Detection," Journal of Visual Communication and Image Representation, vol. 97, p. 103967, DOI: 10.1016/j.jvcir.2023.103967, 2023.
[29] S. Wang, E. Zhu, J. Yin and F. Porikli, "Video Anomaly Detection and Localization by Local Motion Based Joint Video Representation and OCELM," Neurocomputing, vol. 277, pp. 161–175, DOI: 10.1016/j.neucom.2016.08.156, 2018.
[30] S.-A. Jin, T. Kämäräinen, P. Rinke, O. J. Rojas and M. Todorović, "Machine Learning as a Tool to Engineer Microstructures: Morphological Prediction of Tannin-based Colloids Using Bayesian Surrogate Models," MRS Bulletin, vol. 47, no. 1, pp. 29–37, DOI: 10.1557/s43577-021-00183-4, 2022.

ملخص البحث:

تُعالج الأتمتة المتعلّقة بالتّوزيع والتّخزين الآمنين لبيانات الفيديو أوجُه القصور في العمليات اليدوية. وينسجم التوزيع والتّخزين الآمن للبيانات مع الأهمّية المتزايدة لحماية بيانات الفيديو. وتواجه الطرق التقليدية في توزيع بيانات الفيديو وتخزينها جملةً من المحدِّدات، منها نقص القابلية للتّوسيع، وازدواجية البيانات، ومشاكل الخصوصية، والقضايا المتعلّقة بالزّمن (سرعة الكتابة والقراءة)، وفعالية التّخزين.

تحاول هذه الدّراسة معالجة تلك المحدِّدات وتقليل آثارها على أنظمة توزيع بيانات الفيديو وتخزينها من خلال تحقيق التّوزيع الأمثل لبيانات الفيديو، مع تحسين الأمور المتعلّقة بالتّكرار وسماحية الخطأ، وضمان التّوافر المستمّر للبيانات حتى في حالات الفشل. يُضاف الى ذلك، التّركيز على بساطة النظام وسهولة الوصول الى البيانات، خاصّة عند التّعامل مع مجموعات بيانات الفيديو الضخمة وما لها من تطبيقات متنوّعة بما فيها الشّبكات الاجتماعية وألعاب الفيديو وشبكات النّقل.

وقد تمّت مقارنة النّظام المقترح مع عدد من الأنظمة الأخرى للتوزيع والتخزين الآمنيْن لبيانات الفيديو، واتضح تفوّق النّظام المقترح على الكثير من الأنظمة التقليدية؛ فقد بلغ معدّل زمن إزالة التشفير للنظام المقترح (0.208) ثانية، ومعدل زمن التشفير (0.182) ثانية، ومعدل زمن التوزيع (1.620) ثانية، وكانت الأقّل مقارنة بأزمنة عمل غيرها من الأنظمة.

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