Computer Modeling in
Engineering & Sciences

Tech Science Press

DOI: 10.32604/cmes.2024.053430

ARTICLE

Practical Privacy-Preserving ROI Encryption System for Surveillance Videos Supporting Selective Decryption

Chan Hyeong Cho, Hyun Min Song* and Taek-Young Youn*

Department of Cyber Security, Dankook University, Yongin, Gyeonggi-Do, 16890, Republic of Korea
*Corresponding Authors: Hyun Min Song. Email: hyunminsong@dankook.ac.kr; Taek-Young Youn. Email: taekyoung@dankook.ac.kr

Received: 30 April 2024 Accepted: 20 August 2024 Published: 31 October 2024

ABSTRACT

With the advancement of video recording devices and network infrastructure, we use surveillance cameras to protect our valuable assets. This paper proposes a novel system for encrypting personal information within recorded surveillance videos to enhance efficiency and security. The proposed method leverages Dlib's CNN-based facial recognition technology to identify Regions of Interest (ROIs) within the video, linking these ROIs to generate unique IDs. These IDs are then combined with a master key to create entity-specific keys, which are used to encrypt the ROIs within the video. This system supports selective decryption, effectively protecting personal information using surveillance footage. Additionally, the system overcomes the limitations of existing ROI recognition technologies by predicting unrecognized frames through post-processing. This research validates the proposed technology through experimental evaluations of execution time and post-processing techniques, ensuring comprehensive personal information protection. Guidelines for setting the thresholds used in this process are also provided. Implementing the proposed method could serve as an effective solution to security vulnerabilities that traditional approaches fail to address.

KEYWORDS

Privacy de-identification; selective decryption; surveillance video

1 Introduction

Recent advancements in image and video processing technologies have significantly transformed various aspects of our lives. The progress made in digital storage technology is particularly noteworthy, making video recording devices such as cameras and camcorders more accessible. These devices are not only used for personal recording but also for protecting valuable assets and individuals. This progress has led to the expansion of video processing tools like closed-circuit television (CCTV) and internet protocol (IP) cameras employed for security and surveillance purposes. These technologies find applications in monitoring, recording, and analyzing videos by individuals and public institutions, becoming more incorporated into our lives.

This development brings numerous benefits, including enhanced security, more efficient resource management, and improved criminal tracking and investigation capabilities. Thus, the impact of advances in image and video processing technologies extends across various facets of our society, underscoring the growing significance of research and understanding in this field. However, alongside these advancements comes the risk of personal information exposure, either through external attacks on recorded videos or through misuse of the footage for unintended purposes. This threatens personal privacy, necessitating robust measures to protect individuals' recorded personal information from potential attackers. Understanding and implementing de-identification processes is not just a necessity, but also an empowering step towards protecting individuals' privacy. These processes ensure that the privacy of individuals captured in videos remains intact, even in the face of malicious attempts to compromise their data.

De-identification technologies strike a balance between security and privacy. They protect against the aforementioned attacks, employing technology that de-identifies the target, thereby preserving privacy. These technologies can be categorized into two main approaches: those utilizing encryption methods and those employing non-encryption techniques such as mosaic, blur, and masking. Encryption-based de-identification methods involve encrypting the video frames to protect the privacy contained within them [1,2]. This approach offers the advantage of allowing the decrypted information to be re-identified. However, it may incur relatively high overhead during the encryption process [3–5]. On the other hand, de-identification technology using non-encryption methods [6–8] involves directly modifying the video frames, such as applying blur or mosaic effects, to protect personal information. This approach is cost-effective as it typically requires less overhead than encryption-based methods [9–12]. However, since the video frames are directly modified, additional copies of the video may need to be stored for re-identification purposes, potentially introducing security concerns regarding the storage of the modified video.

De-identification technology is not a one-size-fits-all solution. Depending on the application range, it can be adapted to suit different needs. It can be categorized into two main types: full-frame de-identification and Region of Interest (ROI) de-identification, which focuses on de-identifying only specific regions known as ROIs. Full-frame de-identification involves encrypting the entire video frame, thereby ensuring that all information is protected from exposure [13–15]. However, when using the Full-frame de-identification method for video, it is necessary to utilize it through a re-identification process. Therefore, there is a risk of exposing personal information unrelated to the intended use during the process. On the other hand, ROI de-identification targets only specific ROIs within the frame, resulting in relatively lower overhead compared to full-frame de-identification [16–18]. The surrounding areas outside the ROI remain identifiable by de-identifying only the ROIs, such as through encryption. This approach allows for behavior analysis while protecting the privacy of the target.

Surveillance videos, including those captured by CCTV and IP cameras, are ubiquitous worldwide, monitoring various locations for security and investigative purposes. However, these cameras operate around the clock, often capturing individuals without their knowledge or consent. These videos can compromise individuals' privacy and potentially cause harm if misused. Therefore, it is crucial to employ de-identification technology to dispense the captured personal information in surveillance videos.

Surveillance videos are recorded with surveillance filming equipment in investigative scenarios such as criminal investigations or locating lost items and missing persons. To effectively de-identify such surveillance footage, the following requirements must be met:

- De-identification technology must be implemented to allow re-identification of original data when needed. Surveillance videos are recorded for long durations and occupy large storage capacities. Using technology that does not permit re-identification would require additional storage space for identification purposes. Therefore, a method that allows efficient use of storage space while enabling re-identification must be employed.

- Personal information should be de-identified, while non-personal information must remain identifiable. For surveillance videos to be used in criminal investigations or lost property detection, it is essential to identify subjects' actions. However, if all frames are de-identified, re-identifying frames before use may expose the personal information of unrelated individuals.
- The re-identification process must ensure that personal information unrelated to the intended purpose is not exposed. Even when using the video for its intended purpose, investigators or law enforcement officers should only re-identify the information of subjects relevant to the investigation. Otherwise, personal information of unrelated subjects may be exposed, leading to potential misuse of the video for unintended purposes.

Numerous studies have been undertaken to address the requirements above. In [15,19], a method was proposed to protect videos using encryption techniques that allow re-identification. However, these studies employ encryption methods that encrypt the entire frame, making viewing non-privacy content in the de-identified videos impossible. On the other hand, In [20], a method was proposed to protect privacy by encrypting only the ROIs recognized within the video. However, the ROI encryption method described above suffers from the problem that during the encryption process, the same key is used to encrypt all ROIs, allowing the recipient of the key to re-identify all ROIs of all subjects during the re-identification process of de-identified videos. Recognizing this limitation, Hosny et al. [21] highlighted the security issue arising when identical keys are utilized for encrypting the ROIs, advocating for a key generation approach for ROI encryption where distinct keys are assigned to each ROI. This method enhances the protection of personal information about unrelated entities during video usage. However, complete security needs to be ensured for surveillance video. ROI encryption technology relies heavily on ROI recognition technology for identifying ROIs and executing encryption. Consequently, failure to recognize an ROI can immediately result in personal information leakage. Despite recent advancements in artificial intelligence technology leading to improved ROI recognition rates, achieving 100% recognition still needs to be achieved, necessitating additional solutions to address this scenario. Furthermore, there is a pressing need to enhance efficiency concerning the cost of storing and managing encryption keys by employing distinct keys for each ROI.

In this paper, we propose a privacy-preserving de-identification system for surveillance videos supporting selective decryption to address the aforementioned challenges. The proposed technology offers the following contributions:

- **Practical ROI Encryption and Selective Decryption for Surveillance Video.** This paper proposes an efficient encryption method that is not just theoretical but also practical. It utilizes standard encryption techniques to encrypt ROIs while maintaining target identification and protecting personal information. The process involves tracking recognized entities' ROIs within the video based on inter-frame distances, generating a distinguishable ID for each entity, and using this to create entity-specific keys for encryption. This approach improves key management and storage space efficiency, making it a viable solution for real-world applications. Additionally, it provides selective decryption using the generated entity-specific keys, ensuring the protection of personal information unrelated to the intended purpose when using de-identified footage.
- **Enhanced Privacy Protection.** Existing ROI encryption techniques rely on ROI recognition technology, which may not provide 100% accuracy, leading to unencrypted user privacy information. The proposed technology in this paper addresses this issue by predicting the

position of unrecognized ROIs using the ROI information of recognized entities. By leveraging the characteristic of objects moving continuously within the video, the method predicts and protects the position of unrecognized ROIs for entities that disappear and reappear in the video. This approach ensures the protection of unrecognized ROIs and prevents privacy leakage due to the limitations of ROI recognition technology.

The proposed technique in this paper addresses the limitation of ROI recognition technology, where 100% recognition is not feasible due to errors in recognizing ROI entities in one out of four sample videos, resulting in 64 frames where ROIs are not identified. To overcome this, a post-processing method is employed to predict the positions of entities in the missed frames and perform encryption. Additionally, comparing execution times with other encryption methods confirms the suitability of this approach. Moreover, experiments are conducted to determine appropriate and safe threshold values used in this technique, providing guidelines for its secure implementation.

In Section 2, we provide a table detailing related technologies to our proposed technique and descriptions and characteristics of each. In Section 3, we outline the fundamental concepts of our proposed technique, its target systems, and the requirements for securing surveillance videos. We offer a detailed explanation of our approach and fundamental algorithms. Finally, Section 4 presents experiments and analyses on whether our proposed technique effectively protects subjects' personal information in surveillance videos. We also analyze the safety of post-processing methods we support for enhanced security and experimental results on anonymization efficiency.

2 Related Work

With increasing awareness of the importance of privacy, researchers have explored various methods to protect privacy in recorded videos. Table 1 summarizes the methods for protecting personal information within videos. Kadam et al. [15,22] proposed a technique to ensure privacy by employing complete encryption of the entire frame. Encrypting every pixel of information within a captured frame makes the video indiscernible, thereby protecting the subject's personal information. The approach outlined in this study leverages symmetric key encryption such as Advanced Encryption Standard (AES) and an encryption methodology based on chaos maps for comprehensive de-identification, offering a high level of security by effectively encrypting all aspects of the video data. The advantage of full video encryption lies in robust data protection and enhanced security. However, this heightened security entails drawbacks such as increased processing and bandwidth requirements and computationally intensive encryption and decryption processes.

Table 1: Comprehensive comparison of research on video privacy

Proposed in	De-identification method	Re-identification method	Range of de-identification	Data type	Number of keys	Key management
[15,22]	Encryption	Decryption	Whole frame	Video	One key for entire video	Easy
[23]	Non-encryption	Save another image for re-identification	Only ROI	Image	N/A	N/A

(Continued)

Table 1 (continued)

Proposed in	De-identification method	Re-identification method	Range of de-identification	Data type	Number of keys	Key management
[8]	Non-encryption	Save another video for re-identification	Only ROI	Video	N/A	N/A
[24]	Encryption	Decryption	Only ROI	Video	One key for all ROI's	Easy
[21]	Encryption	Decryption	Only ROI	Video	One key for one ROI	Hard
Proposed method	Encryption	Decryption	Only ROI	Video	One key for one entity	Easy

Gross et al. [23] proposed a technique for de-identifying ROIs in an image using blur processing and mosaics. Agrawal et al. [8] extended this de-identification technology to video applications. These techniques enable specific video parts to be altered or obscured, rendering the original information challenging to discern. This process protects privacy and hinders the recovery of the original image. Blurring and mosaic masking techniques obscure the original image, rendering it unrecognizable. Moreover, deletion and modification methods involve removing or altering pixel values in the original image, further complicating the re-identification of the original image. However, due to the nature of this de-identification process, separate video storage in a format where the ROIs of the captured entities can be identified is necessary for re-identification. Storing identifiable videos separately leads to storage space inefficiencies and incurs additional costs associated with protecting the identifiable videos.

Farajallah et al. [24] proposed an ROI encryption method that identifies and encrypts only the Region of Interest (ROI). Due to the relatively minor encryption scope compared to encrypting the entire video, ROI encryption requires less computational effort. Additionally, areas outside the ROI remain unencrypted when viewing the video, allowing for behavior identification. However, the encryption method mentioned above employs the same key across all ROIs. This approach may pose challenges, as it enables decryption of all ROIs if the key for one ROI is compromised, potentially compromising the security of the captured entity's ROI. To protect the ROIs of captured entities, Hosny et al. [21] proposed a method that encrypts all recognized ROIs using different keys generated through a key generation process based on block scrambling and a chaotic logistic map. In video utilization, to encrypt the ROIs of captured entities across all frames in the video and execute a re-identification process, all the numerous encryption keys generated during the encryption process must be retained. Given the characteristics of surveillance videos capturing numerous ROIs over extended durations, storing all encryption keys proves inefficient regarding storage space and key management. Although current advancements in ROI recognition technology demonstrate a high recognition rate, if even a single frame of an entity within the video cannot be recognized due to factors that do not ensure a 100% recognition rate, a scenario where ROI information is immediately leaked could arise, posing a security risk. Hence, measures are necessary to address these challenges.

3 Privacy-Preserving De-Identification System for Surveillance Videos

This chapter provides a comprehensive overview of the system model and methodology for implementing the Privacy-Preserving De-Identification System for Surveillance Videos.

3.1 System Models

This paper proposes a system model, as illustrated in Fig. 1. Initially, an owner captures videos through CCTV and IP cameras installed by individuals or public institutions. The owner securely stores the captured videos and protects the privacy of the individuals depicted in the footage by encrypting the videos using their own encryption key. Subsequently, a client utilizes the stored videos for specific purposes. Depending on the circumstances, the client could be law enforcement officers, judges, or other relevant parties. Initially, the client requests the videos for their intended use from the owner. Upon confirming the client's request, the owner sends the requested videos to the client. The client then reviews the received videos and identifies targets relevant to their intended purpose. Subsequently, the client requests the decryption keys necessary to decrypt the targets. The owner responds by sending the requested decryption keys to the client, who then decrypts the targets and utilizes them for their intended purposes. These purposes can range from criminal investigations to lost item detection, demonstrating the versatility and practicality of the proposed system.

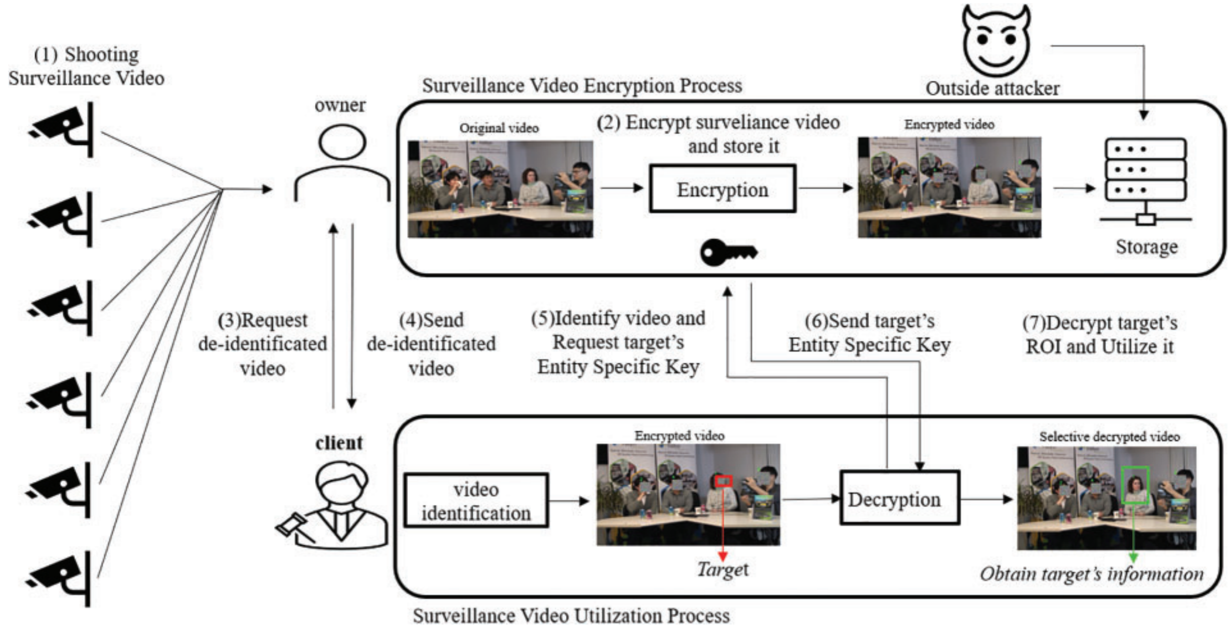
(1) Shooting Surveillance Video
owner
Surveillance Video Encryption Process
Outside attacker
Original video
(2) Encrypt surveliance video and store it
Encryption
Encrypted video
Storage
(3)Request de-identificated video
(4)Send de-identificated video
(5)Identify video and Request target's Entity Specific Key
(6)Send target's Entity Specific Key
(7)Decrypt target's ROI and Utilize it
client
video identification
Encrypted video
Target
Decryption
Selective decrypted video
Obtain target's information
Surveillance Video Utilization Process

Figure 1: System model diagram depicting the encryption/decryption process

One of the key challenges in the process of creating, managing, and utilizing surveillance videos is the risk of privacy breaches. Malicious attackers may compromise the privacy of individuals by attacking and leaking the stored videos. Additionally, if the captured video is shared with the client and used without adequate privacy protection measures, it may inadvertently include irrelevant information about individuals not related to the intended purpose of use. This unintended inclusion of unrelated information poses a significant threat to the privacy of the individuals depicted in the video, underscoring the urgency and importance of the proposed security models. This paper proposes two security models to address these threats, each offering varying levels of security strength according to the specific requirements.

- **Foundational Privacy for Surveillance Video:** Ensures unauthorized entities cannot access information about the individuals depicted in the captured video. An example scenario includes an attack on the storage system containing the video, resulting in the unauthorized acquisition and subsequent leakage of information about the individuals depicted.
- **Enhanced Privacy for Surveillance Video:** Ensures that during the utilization of captured video, authorized users cannot access information about the individuals depicted unrelated to the intended purpose of video usage. The client may represent a potential threat actor in the system model outlined.

In this security model, foundational privacy for surveillance videos strengthens the privacy protection measures for surveillance videos. It addresses situations where authorized users may inadvertently breach the privacy of individuals depicted in the video, unrelated to the intended usage purpose, during the utilization process. Therefore, this paper proposes a video de-identification solution that supports Enhanced Privacy for Surveillance Video.

3.2 Basic Idea

In this paper, we propose a method for recognizing ROIs in captured videos and distinguishing whether the ROIs belong to the same entity by utilizing the distances between recognized ROIs across frames. If determined to be ROIs of the same entity, they are assigned the same ID; otherwise, a new ID is assigned, a process referred to as ROI-link. We propose utilizing this ROI-link process to enhance strong security for surveillance videos. Generate entity-specific keys using the IDs assigned in the ROI-link process and encrypt the ROIs with these keys. This approach allows for assigning linked entity IDs to the ROIs of each recognized entity in every frame of the video. Utilizing the entity ID assigned through ROI-link, we generate entity-specific keys to differentiate between entities using the assigned IDs and a single primary key. These keys are then used to perform encryption on the ROIs.

Suppose the method proposed in this paper is employed for video encryption. In that case, only the entity-specific key of the requested entity is provided for re-identification during the video utilization process rather than the primary key. This allows for Practical Selective Decryption, which prevents the decryption of an entity's ROI while conveniently decrypting only the ROI of the intended entity. Practical selective decryption prevents personal information infringement by entities unrelated to the intended purpose during video usage.

ROI encryption technology relies on ROI recognition technology to perform de-identification processing on video ROIs. While current ROI recognition technology boasts a near-perfect recognition rate, achieving 100% recognition of entities in all frames is practically unattainable. If an entity remains unrecognized in a surveillance video, even for a single frame, it poses an immediate security threat, necessitating measures to address such scenarios. This paper introduces an ID assignment mechanism within the video to mitigate this issue. Additionally, a post-processing procedure is proposed to enhance security by predicting ROIs using the ROI position data from neighboring frames where the target entity has been recognized and frames where ROI recognition has not been successful.

Furthermore, considering cost efficiency, many entity-specific keys for encryption can be generated using the entity's ID and primary key. Thus, even if not all keys are stored, the entity-specific key for the desired entity can be dynamically generated through the primary key and ID values, ensuring security during identification. This approach offers benefits in terms of key management and storage space efficiency.

3.3 Proposed Method

In this chapter, we offer an in-depth elucidation of the video privacy de-identification process and the video privacy selective recovery process, which together constitute the practical de-identification system proposed in this paper.

Algorithm 1: Privacy de-identification system

```
procedure PRIVACY DE-IDENTIFICATION SYSTEM (input_video, primary_key, entity_info_file)
    detector ← Dlib.cnn_face_recognition
    previous_entity_location ← {}    ▷ Save entity's ID and ROI location using Python dictionary
    entity_id_counter ← 0
    frame_number ← 0
    writer ← info_file
    writer.writerow(['frame_num', 'ID', 'x', 'y', 'w', 'h'])
    while True do
        ret, frame ← read_frame(input_video)
        if not ret then
            break
        end if
        ROIs ← detect_ROI(detector, frame)
        for ROI in ROIs do
            ROI_ID ← ROI_Link(ROI, previous_entity_location(ROI, previous_entity_location))
            writer.putText(frame, str(ROI_ID), (x, y, w, h))
            entity_specific_key ← hash_algorithm(primary_key + str(ROI_ID))
            encrypt_ROI(frame, x, y, w, h, entity_specific_key)
        end for
        frame_number ← frame_number + 1
    end while
    video_capture.release()
end procedure
```

In order to securely protect the privacy of individuals captured in surveillance videos from potential security threats, a systematic approach is required to de-identify the privacy of the subjects involved. Fig. 2 outlines an efficient encryption protocol. This process entails an Entity Recognition phase, wherein the ROI of the subject within the surveillance video is identified. Subsequently, an ROI-link process establishes connectivity between entities across consecutive frames, assigning a unique ID to each entity identified through this process. The process culminates in Encryption using entity-specific keys, which are generated for each entity and used for Encryption. Following this encryption process, post-processing is applied to entities not recognized by ROI recognition technology. The method proposed in this paper enhances entity security through ROI-link and entity-specific keys, enabling convenient selective decryption and addressing security vulnerabilities arising from ROI recognition technology through post-processing.

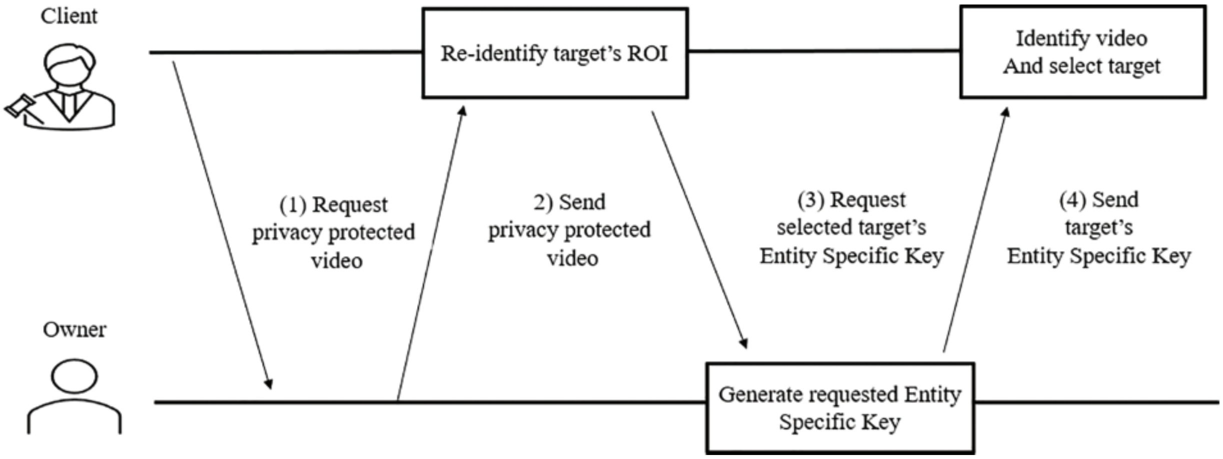

Figure 2: Process of privacy de-identification

Entity Recognition

In order to protect the privacy of entities captured in surveillance videos, it is essential to define the scope of protection clearly. This paper defines the scope of privacy by considering the face as a representative factor that can identify a person visually. For the recognition of facial areas of entities captured in the video, we utilized Dlib's CNN-based face detection module [25,26]. According to [27,28], Dlib's CNN-based face detection offers advantages such as being easy to implement, working effectively with odd angles, and being robust to different face occlusions compared to other face detection algorithms. Dlib's CNN-based face detection module utilizes deep neural networks to extract the positions of faces in frames and calculate confidence scores for each detected face [29–31]. As seen in Lines 7 and 16 of Algorithm 1, this module returns the coordinates (x, y, w, h) of bounding boxes that mark the locations of the recognized ROIs. This process enables the identification of areas within the protected scope and is essential for subsequent analysis and operations on facial data within computer vision applications.

ROI-Link

The ROI-link process utilizes the spatial information of ROIs present across consecutive frames to track entities' ROIs by measuring the distance between ROIs in successive frames and assigning unique IDs to each recognized ROI. The ROI-link process establishes connectivity among ROIs corresponding to each entity identified during Entity Recognition. Entities within the video typically exhibit continuous movement rather than instant disappearance, highlighting the need for a robust tracking process. Utilizing these characteristics, the ROI-link process begins with ROI recognition, assigning unique entity IDs to the ROIs within each frame. Subsequently, a tracking process based on ROI distances between frames is employed to identify entities. Consistently recognized entities across consecutive frames are assigned the same ID, while new entities are allocated new IDs. This method ensures efficient ROI differentiation. Algorithm 2 illustrates the ROI-link step. This step includes the "get ROI location" function in the second line of Algorithm 2, which retrieves the bounding box location information of the detected ROI from the ROI detector and fetches the ROI information for the recognized entity. The "find_center_of_ROI" function in the third line of Algorithm 2 determines the center of the input ROI. It then calculates the distance from the extracted center position to the center of each previously stored entity's ROI and compares it with a predefined threshold. Through this process, if the distance is less than the threshold, the ROIs are considered to belong to the same entity, and the same ID is assigned. Otherwise, a new ID is generated and assigned. This process is described in Lines 5 to 16 of Algorithm 2. The significance of the ROI-link process lies in its ability to enable encryption and selective decryption by associating a key with each entity present in the image. It also addresses the limitations of ROI recognition technology, which does not guarantee a 100 recognition rate. Furthermore, this process serves as a complementary processing step, adding value to the overall video analysis.

Algorithm 2: ROI link process

```
1: function ROI Link ($ROI$, $previous\_entity\_location$)
2:     ($x$, $y$, $w$, $h$) ← get_ROI_location($ROI$)
3:     $center$ ← find_center_of_ROI($x$, $y$, $w$, $h$)
4:     $match\_id$ ← None
5:     for entity_id, ($x_{prev}$, $y_{prev}$, $w_{prev}$, $h_{prev}$) in previous_entity_location.items() do
6:         $prev\_center$ ← find_center_of_ROI(($x_{prev}$, $y_{prev}$, $w_{prev}$, $h_{prev}$)
7:         $distance$ ← calculate_distance($center$, $prev\_center$)
8:         $threshold$ ← 220
9:         if distance $<$ threshold then
10:            $match\_id$ ← entity_id
11:            break
12:        end if
13:        if match_id is None then
14:            $match\_id$ ← entity_id_counter
15:            previous_entity_location[match_id] ← ($x$, $y$, $w$, $h$)
16:            entity_id_counter += 1
17:        end if
18:    end for
19:    return $match\_id$
20: end function
```

Encryption Using Entity-Specific Key

To efficiently encrypt objects captured in surveillance videos, each object is assigned an independent ID to generate object-specific keys. This process, described in Algorithm 1, Line 18, combines the object's assigned ID with the owner's master key. The combined information is input into a hashing algorithm to produce a fixed-size output, which serves as the object-specific key.

Using this method, even if the object-specific key for a particular object is provided during client video usage, the one-way nature of the hashing algorithm prevents inference of the master key. This ensures that the ROIs (Regions of Interest) of objects unrelated to the client's intended use cannot be deciphered. Additionally, the fixed output size of the hash algorithm makes it suitable as an encryption key regardless of the ID's size. An appropriate encryption algorithm tailored to video data must encrypt the recognized ROIs. In surveillance videos, ROIs vary in size and are not allocated in block units. Therefore, to effectively encrypt these diverse ROIs, the size of the ciphertext must match the size of the plaintext to prevent distortion. This paper adopts the Counter (CTR) mode to address this challenge. CTR mode is a block cipher mode that encrypts a constantly incrementing counter value to generate a counter stream and performs encryption through an XOR operation between the generated counter stream and the plaintext–due to this characteristic, allowing encryption without requiring block-size alignment. It maintains the same input and output sizes, preventing distortion caused by the generated ciphertext. The encryption process described in Lines 14 to 20 of Algorithm 1 includes storing information about the recognized ROIs, generating the entity-specific key, and using it for encryption. The encryption process in the algorithm follows these steps, as described in Algorithm 3: (1) In the ROI-link process for the given entity, the assigned ID value is combined with the owner's primary key and passed through a hashing algorithm to generate an entity-specific key. (2) Extract the bounding box region of the recognized ROI. (3) Separate the RGB channels of the extracted ROI. (4) Flatten the values of each channel into a one-dimensional array and convert them into a byte stream. (5) A flattened byte stream is encrypted using the generated entity-specific key through CTR mode. (6) Convert the encrypted channel values back into a one-dimensional array. (7) Reconstruct the one-dimensional array into a three-dimensional image. (8) Replace the bounding box in the original frame with the encrypted image. Using this method prevents alignment issues during the re-identification process in videos.

Algorithm 3: Encrypt ROI algorithm

```
function ENCRYPT_ROI (frame, x, y, w, h, key)
    $cipher \leftarrow$ AES.new($key$, AES.MODE_CTR)    ▷ Initialize AES cipher with key in CTR mode
    $face\_pixel \leftarrow frame[y : y + h,\ x : x + w]$    ▷ Extract the region of the face from the frame
    $r, g, b \leftarrow$ split($face\_pixel$)    ▷ Split the face pixels into R, G, B channels
    $r\_values \leftarrow$ array($r$).$flatten$()
    $g\_values \leftarrow$ array($g$).$flatten$()
    $b\_values \leftarrow$ array($b$).$flatten$()    ▷ Flatten each channel (R, G, B) into a one-dimensional array
    $r\_bytes \leftarrow r\_values.astype$(uint8).$tobytes$()
    $g\_bytes \leftarrow g\_values.astype$(uint8).$tobytes$()
    $b\_bytes \leftarrow b\_values.astype$(uint8).$tobytes$()    ▷ Convert flattened arrays into byte streams
    $r\_encrypted \leftarrow$ cipher.encrypt($r\_bytes$)
    $g\_encrypted \leftarrow$ cipher.encrypt($g\_bytes$)
    $b\_encrypted \leftarrow$ cipher.encrypt($b\_bytes$)    ▷ Encrypt each byte stream using the AES cipher
    $r\_encrypted\_array \leftarrow$ frombuffer($r\_encrypted$, dtype = uint8)
    $g\_encrypted\_array \leftarrow$ frombuffer($g\_encrypted$, dtype = uint8)
    $b\_encrypted\_array \leftarrow$ frombuffer($b\_encrypted$, dtype = uint8)    ▷ Convert each encrypted byte stream back into a one-dimensional array
    $encrypted\_image \leftarrow$ stack([$r\_encrypted\_array, g\_encrypted\_array, b\_encrypted\_array$], axis = 1)    ▷ Stack the encrypted arrays into a three-dimensional image
    $encrypted\_image \leftarrow encrypted\_image.astype$(uint8).$reshape$($face\_pixel.shape$)    ▷ Reshape the stacked array to match the shape of the original face pixels
    $frame[y : y + h, x : x + w] \leftarrow encrypted\_image$
end function
```

After encrypting the ROIs as described, the ID value of each recognized ROI is displayed at the top-left corner of the ROI in the video frame. This allows the client and the owner to recognize the decryption target during re-identification. Additionally, information such as the frame number, the location of the ROI, and its ID is recorded. This facilitates convenient retrieval during subsequent re-identification processes, enabling the decryption of only the ROIs with the desired ID within the video.

Post Process for Stronger Privacy

Given the inherent limitations of ROI recognition technology, which does not achieve 100% recognition of entities in all frames due to factors like obstacles and recognition errors, the necessity of encrypting the ROI of an unrecognized entity during the ROI recognition process becomes apparent. Existing ROI encryption technologies lack ROI connectivity between frames, making it challenging to predict and encrypt the ROI of an entity. However, in this paper, the ROI-link process establishes connectivity for each entity's ROI between frames, enabling prediction of the ROI location in an unrecognized frame. Entities identified in surveillance videos exhibit continuous movements rather than abrupt disappearances. Therefore, if an unrecognized frame lies between two recognized frames, the entity can be presumed to exist but remain unrecognized. Leveraging this approach, privacy protection for captured entities is achieved by encrypting the ROI in the unrecognized frame and replacing it with the nearest recognized frame's ROI position.

In Algorithm 4, a specific pseudocode is provided to perform these processes. During the encryption process, additional information files containing generated data and the primary key required to create the entity-specific key are requested. Moreover, access is made to the previously encrypted video to perform processing only on frames requiring post-processing. The initial step involves organizing information for each frame recognized by ID. For instance, if the frame numbers recognized for entity ID 1 are (219, 220, 225, 226), and the difference between 225 and 220 is greater than 1, frames between 225 and 220 are presumed to contain the entity but are unrecognized. Subsequently, the unrecognized frame numbers per ID are transformed into information regarding unrecognized IDs per frame number. This transformation process, detailed in Lines 15 to 27 of Algorithm 4, is a crucial step in the method. Encryption is applied to the unrecognized entities in the encrypted video. This process enables the protection of personal information for unrecognized entities through inference of unrecognized IDs to generate entity-specific keys and prediction of ROI locations using nearby frames, thereby protecting the privacy of unrecognized entities through ROI recognition technology.

Algorithm 4: Postprocess

```
 1: procedure POSTPROCESS (encrypted_video, info_file, primary_key)
 2:     recog_ids_in_framenum ← find_exist_ids(info_file)
 3:     miss_ids_in_framenum ← {}                ▷ dictionary of missed frames per ID
 4:     for ID in recog_ids_in_framenum.keys() do
 5:         miss_frames ← []
 6:         for i in range(len(info_file) − 1) do
 7:             if (recog_ids_in_framenum[ID][i + 1] − recog_ids_in_framenum[ID][i]) > 1 then
 8:                 for miss_frame in range(recog_ids_in_framenum[ID][i]+ 1, recog_ids_in_framenum[ID][i + 1]) do
 9:                     $miss_{f}rames$.append(miss_frame)
10:                 end for
11:             end if
12:         end for
```

(Continued)

Algorithm 4 (continued)

```
            $miss\_ids\_in\_framenum[ID] \leftarrow$ set($miss\_frames$)    ▷ add missed frame information to dictionary
    end for
    $post\_process\_info \leftarrow$ ‘frame_per_missed_ID_and_ROI_location’    ▷file for encrypted missed ROI
    with open($post\_process\_info$,‘$w$’) as $writer$:
        $writer$.writerow([‘$frame\_num$’, ‘$miss\_id$’, ‘$x_n$’, ‘$y_n$’, ‘$w_n$’, ‘$h_n$’])    ▷“n” means nearest
    for $ID$ in $recognized\_ids\_in\_framenum.keys()$ do
        for $missed\_set$ in $missed\_ids\_in\_framenum[ID]$ do
            for $target\_frame$ in $missed\_set$ do
                $n\_frame \leftarrow$ calculate_missing_frames_nearest_frame($target\_frame$)
                $x_n, y_y, w_n, h_n \leftarrow$ load_ROI_info($ID$, $n\_frame$, $info\_file$)
                $writer$.writerow([$target_f rame, ID, x_n, y_n, w_n, h_n$])
            end for
        end for
    end for
    $data\_frame \leftarrow$ read_info($post\_process\_info$)
    $frame\_num \leftarrow 0$
    while True do
        $ret, frame \leftarrow$ readFrame($encrypted\_video$)
        if not $ret$ then
            break
        end if
        if $frame\_num$ in $data\_frame$[‘$frame\_num$’].$values$ then
            $desired\_rows \leftarrow data\_frame[data\_frame[$‘$frame\_num$’$] == frame\_num]$
            for each $index, row$ in $desired\_rows$.iterrows() do
                $ID_t, x_t, y_t, w_t, h_t \leftarrow row[$‘$miss\_id$’$], row[$‘$x_n$’$], row[$‘$y_n$’$], row[$‘$w_n$’$], row[$‘$h_n$’$]$ ▷“t” means target
                $entity\_specific\_key \leftarrow$ hash_algorithm($primary\_key||$str($ID_t$))
                $frame[y_t : y_t + h_t, x_t : x_t + w_t] \leftarrow$ encrypted_face($frame, x_t, y_t, w_t, h_t, entity\_specific\_key$)
            end for
        end if
        $frame\_num \leftarrow frame\_num + 1$
    end while
    video_capture.release()
end procedure
```

Video Privacy Selective Re-Identification

To utilize the encrypted video as outlined in Fig. 3, the following procedure is followed. Initially, the client reviews the de-identified video and identifies the relevant entity for their intended use by referencing the ID displayed at the top left corner of the entity’s ROI within the video. The client then requests the owner for an entity-specific key to decrypt this identified entity. Upon receiving the request, the owner verifies the ID of the requested entity and combines it with the main key to generate the entity-specific key. Subsequently, the owner sends this key along with the frame numbers and ROI positions corresponding to each ID generated during encryption to the client.

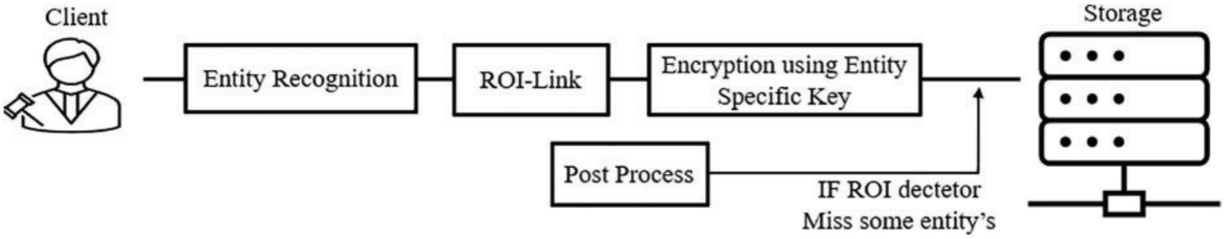
Client
Entity Recognition
ROI-Link
Encryption using Entity Specific Key
Storage
Post Process
IF ROI dectetor Miss some entity's

Figure 3: Process of video privacy selective re-identification

After receiving the entity-specific key and related information, the client decrypts the ROI corresponding to the identified entity. The decryption process is as follows: (1) Filter the information provided by the owner to extract details of the target entity with the desired ID. (2) Using the filtered target information and the entity-specific key, retrieve the pixel values of the ROI in the frames where the target entity appears. (3) Separate the pixel values into RGB channels and decrypt each channel individually. (4) Repeat this process for every frame where the target entity appears. This selective decryption allows the client to decrypt only the ROIs associated with the targeted entity, thereby protecting the privacy of unrelated entities depicted in the video.

4 Analysis

In this chapter, experiments related to the security and performance of the proposed "Practical De-Identification System for Surveillance Videos with Privacy-Preserving ROI Encryption Supporting Selective Decryption" are conducted. The experiments were performed using a Colab notebook with Python 3 runtime type and T4 GPU hardware accelerator. We conducted experiments on the proposed technique using a publicly available dataset widely used for video codec research and development. This collection includes a multitude of video files encompassing various formats and resolutions [32]. From this dataset, we selected four videos that resemble surveillance footage and have an appropriate number of people for our experiments. Table 2 provides information about the sample videos used for analysis. The encryption algorithm used for encryption is AES-CTR. Since all results are simulations, the accuracy of the ROI recognition and encryption algorithms heavily influences the findings. We used a widely recognized dataset and robust algorithms to mitigate this threat. While our experiments used specific video formats and resolutions, the methodology is applicable to a broader range of surveillance scenarios, although performance may vary with different datasets. The proposed method was tested in a controlled environment; hence, its performance in varied real-world conditions with dynamic lighting and occlusions remains to be further evaluated.

Table 2: Sample videos

Name	Frames	Resolution
Akiyo	300	QCIF (288 × 352)
Vidyo1	600	720 p (1280 × 720)

(Continued)

Table 2 (continued)

Name	Frames	Resolution
FourPeople	600	720 p (1280 × 720)
Vidyo4	600	720 p (1280 × 720)

In this chapter, experiments are conducted to evaluate the privacy and performance of the new surveillance video privacy de-identification system proposed in this paper.

4.1 Privacy

In this chapter, privacy experiments and analyses for the proposed method are conducted. The privacy experiments consist of two main experiments. The first experiment evaluates the suitability of the threshold value, which directly influences ID assignment in the ROI-link process. The second experiment assesses the post-processing performed on the video after going through these processes. In 4.1.1, an experiment is conducted to determine the appropriate threshold value that can be safely used for the proposed method, as it directly affects privacy. This experiment aims to provide a threshold that ensures the secure use of the proposed method. In 4.1.2, an experiment is conducted to evaluate whether the post-processing, which is a distinguishing feature of the proposed method compared to other encryption systems, is appropriately executed. This experiment assesses whether the proposed method effectively performs post-processing, setting it apart from other encryption systems. The experimental results derived in this privacy chapter are based on Dlib's CNN-based face detection and standardized AES-CTR mode encryption. Therefore, if the same face detection algorithm and encryption algorithm are used in different environments with the same hardware for the sample videos, the same results can be obtained.

4.1.1 How to Set Threshold in ROI-Link

The surveillance video de-identification system proposed in this paper assigns the same ID to ROIs recognized within frames if the distance between them falls below a specific threshold during the ROI-linking process. The ID serves as the most critical value in distinguishing entities during the generation of entity-specific keys. Therefore, the threshold value plays a crucial role in this ROI-linking process.

In this chapter, we aim to determine the appropriate threshold values used in the ROI-linking process. For this purpose, we analyze the threshold values within the range of [20, 400] for the four sample videos. We evaluate the ratio of the number of entities assigned IDs to the total number of entities in each video, represented as the number of entities divided by the number of IDs Rate. This analysis aims to identify the suitable threshold values for effective ROI linking.

To verify the appropriateness of the threshold value, we use the ratio of the number of entities to the number of recognized IDs within the video. This entity-to-ID ratio allows us to intuitively compare the number of entities present in the video with the number of IDs. This comparison serves as an indicator to assess the suitability of the chosen threshold value for applying the technique to the video.

The threshold range for secure ID assignment in the ROI-linking process can be categorized into three zones. Firstly, the unsecured zone refers to the range where two or more entities in the video are assigned the same ID. In such cases, during the entity-specific key generation process after ROI-linking, the same key may be assigned to multiple entities, leading to potential privacy breaches during selective re-identification. The unsecured zone corresponds to when the number of entities/the number of IDs Rate falls within the range of (100%). Next, the secure but impractical zone refers to the range where multiple IDs are assigned to a single entity in the video. Although using multiple entity-specific keys for a single entity makes selective recovery inefficient, it does not lead to security incidents where the same key is assigned to multiple unrelated entities. The secure but impractical zone corresponds to when the number of entities/the number of IDs Rate falls within the range of (0%, 100%). Finally, the secure and practical zone is where each entity in the video is assigned a single ID. In this scenario, no security issues arise, and performing post-processing and selective recovery is convenient. The secure and practical zone is achieved when the number of entities divided by the number of IDs Rate equals 100%.

Fig. 4 depicts a graph showing the number of entities divided by the number of ID rates for each video sample. For the "akiyo" video, where only one entity appears and there is minimal movement, the number of entities and the number of IDs assigned match across all thresholds, resulting in a 100% rate in all ranges. In the case of "vidyo4," despite having only one entity, rapidly moving ROIs cause multiple IDs to be recognized at lower threshold values. However, as the threshold value increases, the number of entities and IDs align. For videos like "FourPeople" and "vidyo1," where multiple individuals appear, lower threshold values result in more IDs recognized than the number of entities present. Conversely, as the threshold increases, fewer IDs are assigned than the number of entities, leading to the same ID being assigned to different entities. Overall, after considering these results, the threshold value falls within the range of [200, 240], where the number of entities divided by the number of IDs Rate is consistently 100% for all videos, indicating the secure and practical zone. However, it's crucial to assign appropriate thresholds based on video characteristics. The process involves evaluating if the number of IDs is fewer than the number of ROIs recognized in the frame for a given threshold. If so, the threshold is deemed insecure, and reducing it leads to identifying a secure threshold value.

4.1.2 Analysis of Post Process

ROI recognition technology is employed to recognize the ROIs within a video. Despite significant advancements, these recognition methods exhibit near-perfect recognition rates but are unable to achieve 100% accuracy. The remarkable progress in ROI recognition technology is notable in protecting privacy in surveillance videos. However, issues can be encountered due to unique events within frames, such as facial obstruction by obstacles. Even with such recognition technology, a security gap exists as it may fail to recognize all ROIs within the video. For instance, during the de-identification process of surveillance videos to protect privacy, failing to recognize even a single frame could lead to the immediate risk of privacy breach for entities. Therefore, this paper's proposed approach aims to ensure stronger privacy for surveillance videos. It involves encrypting the four specified videos and subsequently identifying frames where entities exist but are not recognized by ROI recognition technology. The proposed method for entity ROI protection is then applied through post-processing.

Among the four sample videos used in the experiments of this paper, namely akiyo, vidyo1, and vidyo4, no unusual events occurred, resulting in perfect ROI recognition for all frames. However, in the case of the FourPeople video, a portion of the face of one of the four entities was obscured by a magazine held in hand, leading to incomplete ROI recognition. Although encryption was applied to address this issue, it was observed that perfect protection was not achieved. Table 3 illustrates the number of frames not recognized for each video. Post-processing experiments for entity protection were conducted to address this challenge, focusing on the FourPeople video as the target. Through post-processing, efforts were made to enhance entity ROI protection.

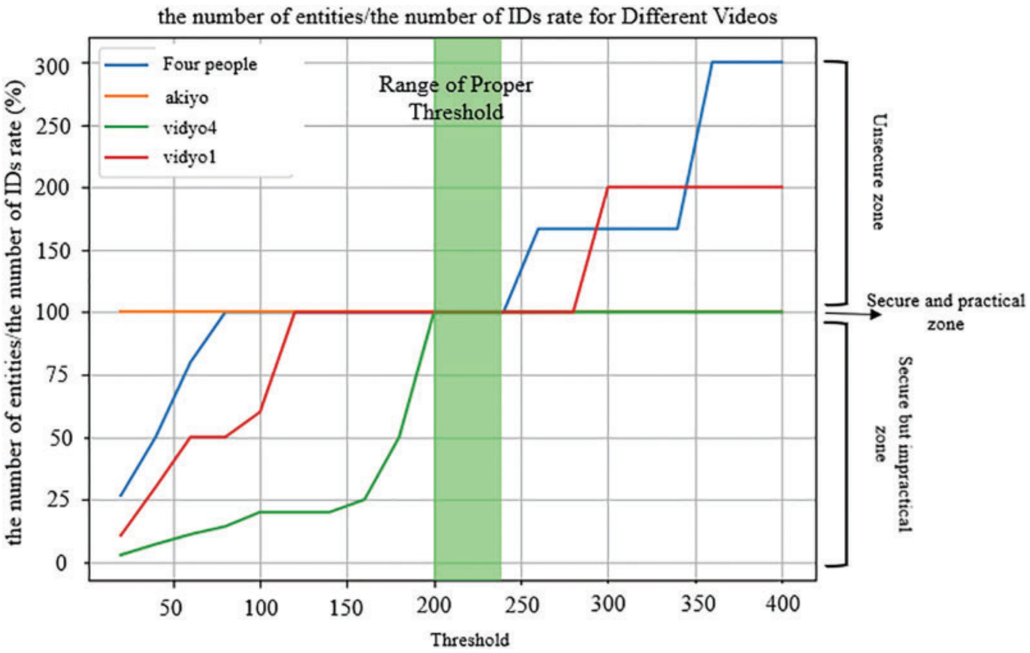

Figure 4: The ratio of the number of entities and generated IDs is according to the threshold. When the threshold is between 200 and 240, it shows complete matching without misidentification

Table 3: Information of ROI-recognition frame number

	Number of frames	Number of ROI-recognition success frames (success rate)	Number of ROI-recognition error frames (error rate)
Akiyo	300	300 (100%)	0 (0%)
Vidyo1	600	600 (100%)	0 (0%)
FourPeople	600	536 (89.3333%)	64 (10.6666%)
Vidyo4	600	600 (100%)	0 (0%)

In the case of the FourPeople video, out of a total of 600 frames, it was observed that 64 frames were not recognized for the entity assigned ID 0 out of the four recognized entities. These frames underwent the proposed post-processing procedure to protect the entities' privacy. The post-processing procedure successfully predicted and protected the ROI (Region of Interest) for all 64 frames where recognition failed. Fig. 5 presents the information before and after the post-processing for a selected frame among the 64 unrecognized frames. As shown in Fig. 5, it can be confirmed that this paper's proposed post-processing procedure effectively covers the security gap caused by the limitations of ROI recognition.

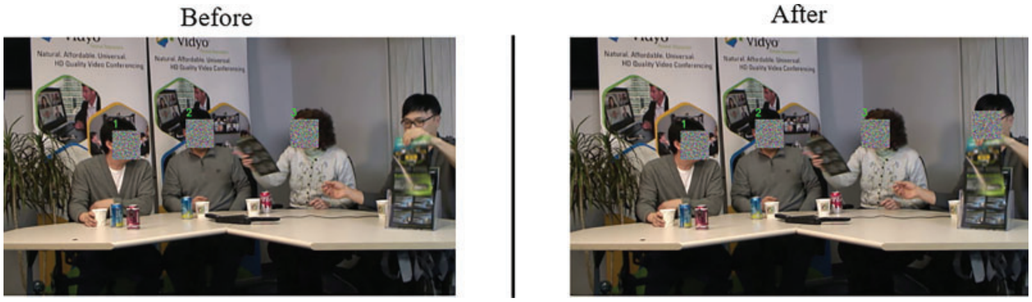

Figure 5: Effection of post-process (frame: 164)

4.2 Performance

This chapter presents the experimental results regarding the performance of the proposed approach in this paper. To measure the performance of the proposed approach, the total execution time for three encryption methods, entire frame encryption using one key, ROI encryption using one key, and the proposed method, was measured for four sample videos. To address potential numerical errors, all calculations were performed using double-precision arithmetic. The encryption methods were performed 100 times for each video, and the average execution time per video was presented. Additionally, we provide the error range for the average time. The error range was calculated by first finding the absolute differences between each processing time and the average time for all results to ensure a simple and intuitive understanding of the range. These values were then summed and divided by the total number of elements, and the resulting value was multiplied by 2.

As shown in Table 4, the entire frame encryption method exhibited the shortest average execution time among the four sample videos. At the same time, it is worth noting that the approach proposed in this paper had the most extended runtime. The primary reason for the shorter runtime of the entire frame encryption method lies in its omission of the target recognition process during encryption, resulting in time savings. Conversely, the proposed method had the most extended runtime due to this experiment's CNN-based ROI recognition method. It is known for its robustness to changes in posture and angle but is characterized by slower execution times. It is anticipated that a slightly reduced runtime could be achieved by employing real-time processing-capable and lightweight recognition techniques for ROI recognition.

Table 4: Average execution time table

	Entire frame encryption	ROI encryption using one key	Proposed method
Akiyo	0.76 s ± 0.29	1.16 s ± 0.37	1.35 s ± 0.33
Vidyo1	14.77 s ± 0.86	17.76 s ± 0.57	22.50 s ± 0.55
FourPeople	14.03 s ± 0.61	16.76 s ± 0.55	21.11 s ± 0.59
Vidyo4	13.87 s ± 0.50	16.79 s ± 0.49	20.45 s ± 0.63

Upon comparing the runtime of the proposed method to ROI encryption using a one-key method, it is evident that the runtime of ROI encryption using a single-key method is shorter. This difference stems from the time required for the ROI-link process to differentiate entities, assign IDs, and generate entity-specific keys based on these IDs. Consequently, while the method proposed in this paper incurs overhead compared to other image de-identification techniques, it is not critical given that it primarily processes stored images rather than engaging in real-time image processing.

The longer execution time observed in the above experiment compared to encrypting a general string with AES is mainly due to the relatively complex process of encrypting pixel information composed of three channels (R, G, B) separately for each channel. This contrasts with encrypting a general string of equivalent size, which comprises a single channel and thus involves a less intricate process.

5 Conclusion

This paper presents a system for de-identifying personal information within videos, leveraging Dlib's CNN-based facial recognition technology to identify Regions of Interest (ROIs) and generate unique IDs through the ROI linking process. These IDs are combined with a master key to create entity-specific keys, enabling selective decryption and enhancing privacy protection for surveillance videos. The proposed method addresses the limitations of ROI technology, particularly in recognizing all objects with 100% accuracy, by utilizing positional correlation of linked ROIs. The solution was rigorously validated through post-processing, effectively covering errors, including 64 frames where objects were not recognized. We also identified an appropriate threshold range for this process, suggesting that our method could be applied to CCTV and IP cameras to prevent inadvertent leakage of personal information during criminal investigations.

However, several limitations must be acknowledged. First, post-processing the entire recorded video poses challenges for real-time applications, as it may introduce delays that limit the system's responsiveness in time-sensitive situations. Second, the heavy reliance on ROI recognition technology means that some objects within the video might not be recognized, particularly in cases where a person is viewed from behind or is wearing a hat, potentially reducing the recognition rate. Third, while enhancing security, the system's use of entity-specific keys for encryption results in higher computational costs than other encryption methods.

In future research, improving the efficiency of the encryption method could help overcome these limitations. Specifically, exploring ways to minimize delays in real-time processing and reduce computational costs would enhance the system's responsiveness. Additionally, optimizing the process of generating and managing entity-specific keys could enable faster and more efficient processing while maintaining secure privacy protection for all objects within the video. If more accurate methods, such as facial recognition, are applied in the process of assigning IDs to objects, it would further improve the effectiveness of object identification and differentiation. These improvements would increase the proposed system's practicality and maximize its efficiency and effectiveness across various applications.

Acknowledgement: This paper is a supplementary and extended version of the paper presented at the 7th International Symposium on Mobile Internet Security (MobiSec23) Conference.

Funding Statement: This work was supported by the Institute of Information and Communications Technology Planning and Evaluation (IITP) funded by the Korea Government (MIST), Development of Collection and Integrated Analysis Methods of Automotive Inter and Intra System Artifacts through Construction of Event-Based Experimental System, under RS-2022-II221022.

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design: Chan Hyeong Cho, Taek-Young Youn; coding: Chan Hyeong Cho; analysis and interpretation of results: Chan Hyeong Cho, Taek-Young Youn, Hyun Min Song; draft manuscript preparation: Chan Hyeong Cho, Taek-Young Youn, Hyun Min Song. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The sample videos and tools used in the experiment are listed in the references section of the paper.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

1. Shi C, Bhargava B. An efficient MPEG video encryption algorithm. In: Proceedings of the Seventeenth IEEE Symposium on Reliable Distributed Systems (Cat. No. 98CB36281), 1998; West Lafayette, IN, USA, IEEE.
2. Lian S. Efficient image or video encryption based on spatiotemporal chaos system. Chaos Soliton Fract. 2009;40(5):2509–19. doi:10.1016/j.chaos.2007.10.054.
3. Li S, Chen G, Cheung A, Bhargava B, Lo KT. On the design of perceptual MPEG-video encryption algorithms. IEEE Trans Circuits Syst Video Technol. 2007;17(2):214–23. doi:10.1109/TCSVT.2006.888840.
4. Kim DH, Kim YG. A method for de-identification analysis of encrypted video. In: 2024 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI DAMT & NCON), 2024; Chiang Mai, Thailand, IEEE.
5. Li X, Yu H, Zhang H, Jin X, Sun H, Liu J. Video encryption based on hyperchaotic system. Multimed Tools Appl. 2020;79:23995–4011. doi:10.1007/s11042-020-09200-1.
6. Gross R, Sweeney L, Cohn J, De la Torre F, Baker S. Face de-identification. In: Protecting privacy in video surveillance. London, UK: Springer, 2009. p. 129–46.
7. Letournel G, Bugeau A, Ta VT, Domenger JP. Face de-identification with expressions preservation. In: 2015 IEEE International Conference on Image Processing (ICIP), 2015; Quebec City, QC, Canada, IEEE.
8. Agrawal P, Narayanan P. Person de-identification in videos. IEEE Trans Circuits Syst Video Technol. 2011;21(3):299–310. doi:10.1109/TCSVT.2011.2105551.
9. Park S, Na H, Choi D. Verifiable facial de-identification in video surveillance. IEEE Access. 2024;12:67758–71. doi:10.1109/ACCESS.2024.3399230.
10. Jeremiah SR, Castro OEL, Sharma PK, Park JH. CCTV footage de-identification for privacy protection: a comprehensive survey. J Internet Technol. 2024;25(3):379–86. doi:10.53106/160792642024052503004.
11. Feng G, Yan T, Yang J. Attribute-consistency reversible pedestrian de-identification in intelligent transportation. IEEE Trans Veh Technol. 2024;1–11. doi:10.1109/TVT.2024.3391834.
12. Cao Y, Zhang Y, Wu J, Fang Y. Multi-channel attribute preservation for face de-identification. Multimed Tools Appl. 2024;10:1–25. doi:10.1007/s11042-024-19308-3.
13. Chiaraluce F, Ciccarelli L, Gambi E, Pierleoni P, Reginelli M. A new chaotic algorithm for video encryption. IEEE Trans Consum Electron. 2002;48(4):838–44. doi:10.1109/TCE.2003.1196410.
14. Zhu ZL, Zhang W, Yu H. MPEG video encryption algorithm based on lorenz chaotic system. J Comput Appl. 2009;28(12):3003–6. doi:10.3724/SP.J.1087.2008.03003.

15. Kadam KS, Deshmukh A. Video frame encryption algorithm using AES. Int J Eng Res And. 2016;5(6):588–91. doi:10.17577/ijertv5is060670.
16. Yu JY, Kim YG. Coding unit-based region of interest encryption in HEVC/H.265 video. IEEE Access. 2023;11:47967–78. doi:10.1109/ACCESS.2023.3276243.
17. Lopez J, Hinojosa C, Arguello H, Ghanem B. Privacy-preserving optics for enhancing protection in face de-identification. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024; Seattle, WA, USA; p. 12120–9.
18. Massoudi A, Lefebvre F, De Vleeschouwer C, Macq B, Quisquater JJ. Overview on selective encryption of image and video: challenges and perspectives. Eurasip J Inf Secur. 2008;2008(1):179290. doi:10.1155/2008/179290.
19. Saleh MA, Tahir NM, Hisham E, Hashim H. An analysis and comparison for popular video encryption algorithms. In: 2015 IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE), 2015; Langkawi, Malaysia, IEEE.
20. Kim Y, Jin SH, Bae TM, Ro YM. A selective video encryption for the region of interest in scalable video coding. In: TENCON 2007–2007 IEEE Region 10 Conference, 2007; Taipei, Taiwan, IEEE.
21. Hosny KM, Zaki MA, Hamza HM, Fouda MM, Lashin NA. Privacy protection in surveillance videos using block scrambling-based encryption and DCNN-based face detection. IEEE Access. 2022;10:106750–69. doi:10.1109/ACCESS.2022.3211657.
22. Dumbere DM, Janwe NJ. Video encryption using AES algorithm. In: Second International Conference on Current Trends in Engineering and Technology-ICCTET 2014, 2014; Coimbatore, India, IEEE; p. 332–7.
23. Gross R, Sweeney L, De la Torre F, Baker S. Model-based face de-identification. In: 2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'06), 2006; New York, NY, USA, IEEE; p. 161–1.
24. Farajallah M, Hamidouche W, Déforges O, El Assad S. Roi encryption for the hevc coded video contents. In: 2015 IEEE International Conference on Image Processing (ICIP), 2015; Quebec City, QC, Canada, IEEE; p. 3096–100.
25. Dlib Python API Tutorials. Electronic resource. Available from: http://dlib.net/python/index.html. [Accessed 2024].
26. Boyko N, Basystiuk O, Shakhovska N. Performance evaluation and comparison of software for face recognition, based on dlib and opencv library. In: 2018 IEEE Second International Conference on Data Stream Mining & Processing (DSMP), 2018; Lviv, Ukraine, p. 478–82.
27. Jadhav A, Lone S, Matey S, Madamwar T, Jakhete S. Survey on face detection algorithms. Int J Innov Sci Res Technol. 2021;6(2):291–7. doi:10.1109/ICISC.2017.8068607.
28. Song J, Kim J, Nang J. Face de-identification using convolutional neural network (CNN) models for visual-copy detection. Appl Sci. 2024;14(5):1771. doi:10.3390/app14051771.
29. Divvala SK, Hoiem D, Hays JH, Efros AA, Hebert M. An empirical study of context in object detection. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009; Miami, FL, USA, IEEE; p. 1271–8.
30. Zou Z, Chen K, Shi Z, Guo Y, Ye J. Object detection in 20 years: a survey. Proc IEEE. 2023;111(3):257–76. doi:10.1109/JPROC.2023.3238524.
31. Sharma S, Shanmugasundaram K, Ramasamy SK. FARECâCNN based efficient face recognition technique using dlib. In: 2016 International Conference on Advanced Communication Control and Computing Technologies (ICACCCT), 2016; Ramanathapuram, India, IEEE; p. 192–5.
32. XiphOrg. Xiph.Org: Derfâs test media collection; 2022. Available from: https://media.xiph.org/video/derf/. [Accessed 2024].