Privacy-Preserving Selective Video Surveillance

Alem Fitwi, Yu Chen
Dept. of Electrical and Computer Engineering, Binghamton University, Binghamton, NY 13902, USA
Emails: {afitwi1, ychen}@binghamton.edu

***Abstract*—The pervasive and intrusive surveillance practices have raised a widespread concern amongst zillions of people about the invasion of their privacy. The privacy breaches in the existing mass-surveillance system are mainly attributed to the exploits of vulnerabilities by adversaries and abuse of cameras by people in charge of them. As a result, there has been a tremendously pressing demand from the public to make the surveillance system privacy-conscious. In this paper, we propose PriSev, a privacy-preserving selective video surveillance method, which enables selective-surveillance where only video frames containing aggressive and suspicious behavioral patterns, like gun brandishing or/and fist-raising, are made available for view by security personnel in the surveillance operation center and for storage. By introducing a lightweight dynamic chaotic image enciphering (DyCIE) scheme, the proposed PriSev method enables onsite object detection and frame encryption at the network edge where the video is created. At the fog/cloud layer, frame decryption is efficiently performed followed by deep-neural-network (DNN) based frame-filtering and selective storage that runs on a surveillance server. In addition, a multi-agent system is introduced for the exchange of deciphering keys between the sending and receiving agents. Extensive experimental study and performance analyses corroborate that the proposed PriSev method is able to efficiently perform a privacy-preserving selective surveillance in real-time.**

***Keywords*-Privacy-preserving, Selective Surveillance, Deep Neural Networks, Frame Enciphering.**

I. Introduction

With the overt goals of reducing crime and ensuring physical security and public safety, today there are about 770 million surveillance cameras deployed around the world. By the year 2021, the number of cameras perched on streets, corners, buildings, border checkpoints, and lamp posts is projected to rise to one billion [24]. The increasing deployment of surveillance cameras enable the law enforcers to garner a great deal of visual information about individuals indiscriminately without their knowledge and consent. As a result, there is a mixed public feeling in relation to the practice of mass-surveillance. On the one hand, a number of people have a good view of it because they believe it has the potential to deter and reduce crimes, and help monitor traffics on top of providing footage of crime scenes as evidence against criminals in courts of law. On the other hand, people are gravely concerned about the invasion of the privacy of individuals through the practice of non-selective surveillance. Hence, the public want the surveillance systems have some intelligence and the ability to protect and anonymize privacy-sensitive attributes of individuals. Besides, there has been a quest for the surveillance systems to have the capability to selectively store only those frames containing criminal acts or suspicious behavioral patterns vital for future reference in lieu of mass-storage to cut off the risk of privacy breaches through misuse and leaks.

The privacy invasion is mainly attributed to the possible abuse of these zillions of surveillance cameras in operation today in many urban areas across the world and network vulnerabilities [16], [19], [21]. Most of the preexisting surveillance systems rely on security personnel who sit in surveillance operation centers to observe any illicit acts and to perform video analytics. With the expanded sighting scope, authorized security personnel in charge of the cameras might abuse them for voyeurism, prowling through windows, blackmailing, and unauthorized collection of data on activities or behaviors of individuals [37], [42], [46]. This necessitates the incorporation of privacy-preserving mechanisms in the mechanical surveillance systems by design that only require minimal and accountable human interventions [5], [16].

Most of today's surveillance systems are deployed based on either the fog or cloud computing architectures. In a cloud based deployment, the video analytics is performed in distant powerful cloud machines. However, there is a catch in such setting in that plain videos may be intercepted by adversaries exploiting network vulnerabilities while on transit from the network edge to the cloud centers. This increases the likelihood of privacy breaches. On the other hand, the edge computing paradigm pushes some intelligence and processing power to the smart cameras allowing the enforcement of privacy-preserving mechanisms like video-frames enciphering to be performed at the point of video creation and collection [13], [14], [15]. This decreases the probability of the spilling of sensitive private data about individuals into the wider cyber space. However, edge computing is a resource-constrained environment that can only support lightweight methods [8], [45]. Hence, an edge-device friendly end-to-end enciphering technique is vital to preserve the privacy of frame contents while on their way to the video analytics center.

Apart from some works that try to address the privacy issues of sharing images in social media networks [37], [46] and some cloud-based surveillance solutions that focus on specific privacy attributes like face and mediation [42], there are no viable solutions to the pressing privacy problems of mass-surveillance yet. In this paper, a comprehensive scheme for Privacy-preserving Selective Video surveillance (PriSev) is proposed considering the computational resource requirements at both cloud/fog layer and the edge of the network where cameras are deployed. The contribution of this paper can be summarized as ensues:

- A novel privacy-preserving selective video surveillance (PriSev) scheme is proposed. In PriSev, video frames that contain some objects are selectively enciphered at

978-1-7281-6607-0/20/$31.00 ©2020 IEEE

the edge to protect any disclosure during transit to the fog/cloud server for analytics. At the fog/cloud layer, frames are deciphered and their contents are classified into innocuous and aggressive behavioral patterns. Only frames with aggressive behavioral patterns are viewed and stored thereby protecting the privacy of innocent individuals. To the best of our knowledge, the PriSev is the first simplified but efficient solution to enable cameras to perform selective surveillance.

- A lightweight dynamic chaotic image enciphering (DyCIE) algorithm at the edge and a deep neural networks (DNN)-based model for frame filtering at a fog/cloud machine are introduced. The DyCIE method can robustly encipher frames at an edge-device faster than preexisting cryptographic schemes. It is preceded by a lighter process of frame scanning to check the presence of any object for selective enciphering. This improves the performance of the scheme in that only object-containing frame are enciphered. At a fog/cloud server, the compute-intensive DNN-based model identifies frames that contain aggressive behavioral patterns like raising fist and gun-brandishing ensuing the deciphering process.
- A thorough security, robustness, and performance analyses of the pragmatic implementation of the proposed scheme is presented. The results corroborate that the proposed scheme is efficient, secure, and robust.

The remainder of this paper is organized as ensues: Section II presents related works in the areas of privacy in video surveillance, enciphering techniques, and privacy-sensitive attributes detection. The PriSev system architecture and the designs of key components are discussed in Section III. The details of the experimental study and performance analysis results are elucidated in Section IV. At last, our concluding remarks are presented in Section V.

II. Related Works

A. Privacy Challenges and Controversies

Contrary to the brazen breach of privacy by the contemporary practice of video surveillance, a number of literature and legislation or constitutions clearly indicate that privacy is a fundamental right and the state of being free from being disturbed or observed by other people without one's consent. [3], [4], [5], [23], [31]. It is the protection of information that says who we are, what we do, what we think, and what we believe in. However, the quite intrusive nature of the existing practice of video surveillance have stripped individuals of their rights to control what information about themselves are collected and shared. The closed-circuit television (CCTV) cameras glean a great deal of information about individuals' daily activities and behavioral patterns without their consents. The controversy does not stop here. Even governments and some pro mass-surveillance people double down by stating that people who have nothing to hide should not be afraid of being spied by CCTV cameras audaciously disregarding the rights of individuals to privacy [23]. As a result, millions of people are very much concerned about the perpetual invasion of their privacy by the intrusive surveillance system in use today [11], [37], [42], [46].

Many reports that confirm the fear of the public about privacy invasion by the practice of invasive mass-surveillance have been published. They explain the many ways the surveillance system is abused. For example, the American Civil Liberties Union (ACLU) has identified five abuses of CCTV cameras, which are criminal abuse, institutional abuse, abuse for personal purposes, discriminatory targeting, and voyeurism [35]. Besides, maneuverable cameras, like pan-tilt-zoom (PTZ) camera, could be abused and directed to intrusively spy on other people in their apartments. One clear case is the spying on the private apartment of the German Chancellor Angela Merkel by a security guard using a nearby museum's CCTV camera [5], [21]. Surveillance systems comprise many communication means, embedded hardware and software. By this virtue, they have been targets of many cyber attacks that could cause the spill of videos to the wider cyber space [10]. Hence, the public recognize the benefits that surveillance cameras can offer in terms of physical security, crime deterrence, traffic control, increased safety, and provision of evidence against offenders in courts of law. They, however, want the surveillance system to have some intelligence so as to be selective.

Almost all of the efforts that have been made so far to preserve privacy by embedding privacy curtailments in smart cameras [11], [37], [42], [46] focus either on masking some privacy attributes like faces or cloud-based protections. However, transferring raw video streams to cloud centers over insecure communication channels increases the risk of privacy breaches. A better approach is migrating some intelligence and processing power to the smart cameras that allows the processing of videos and images at the point where they are created [41]. But the main challenge is that edge devices are unable to handle compute-intensive algorithms because of resource-constraints. Therefore, an approach that makes use of the best of both paradigms, edge and cloud computing, is essential to ensure end-to-end privacy.

On the other hand, focusing on specific privacy attributes like the faces might not be sufficient to preserve privacy. Individuals can still be identified by their gait, clothing, and other distinguishing visual marks on other parts of their body. Hence, one of our design goals is to support full frame enciphering at the edge to prevent identification by attributes other than the face when intercepted on transit.

Another controversy is the trade-off between privacy and usability. Privacy is a human right concept which is very difficult to define with clear boundaries [3], [31]. Perfect privacy implies zero usability which nullifies the very purpose of the existence of surveillance systems, and vice versa. To date, there is not a clearly drawn hard line between them. Hence, the trade-off made in our PriSev is that the privacy of people who do not engage in illegal activities should be fully protected. But the privacy of people who happen to engage in illegal acts cannot be fully protected for their videos must be observed and analyzed by people in charge of the surveillance system for preventive measures and law enforcement purpose. However, their identities will not be revealed through network interception or man-in-the-middle attack.

B. Video Privacy Protection

Any privacy protection scheme for video streams must attain a balance between privacy, clarity, reversibility, security, and robustness [29]. Privacy is the condition of not being identified by human observers in videos. Clarity is a requirement that any privacy protection means should permit the identification of suspicious behavior and the collection of non-sensitive information. Privacy-protection schemes must have a reversibility attribute in order for privacy-protected frames that contain crime scenes to be reversible. Besides, the scheme must be secure and robust. The security ensures that the reversion is done only by authorized party and the robustness guarantees that the scheme does not fail to detect sensitive areas. Generally, image privacy protection schemes can be put into four classes, namely editing, face regions, false color, and JPEG [31]. The editing schemes like blurring, black box, pixelation, and masking are simple but unable to fully hide the private things and they are prone to reconstruction [28], [44]. Most importantly, information is not recoverable. The face regions approach [22], [27], [32] also suffers from not being suitable to real-time processing and reversibility. There exist the same problems in the false color [9], [34] and JPEG [2], [47] methods as well.

Encryption is the most secure member of the editing class privacy protection schemes. It protects private information and sensitive data from unauthorized accesses, and enhances the security of communication between two communicating parties. However, given the real-time nature of videos, the traditional symmetric key cryptographic mechanisms like triple Data Encryption Standard (3DES), International Data Encryption Algorithm (IDEA), and Advanced Data Encryption (AES), and asymmetric key cryptographic mechanisms like Rivest, Shamir and Adleman (RSA), and Elliptical Curve Cryptography (ECC) are not suitable for image encryption. The AES cipher is considered to be one of the most secure ciphers commonly used in secure sockets layer (SSL) or transport layer security (TLS) across the Internet today. However, it is too slow to be employed for real-time video encryption at the edge, where there are limited resources.

For image encryption, chaotic-based encryption mechanism offers better solutions. Chaos based encryption schemes are mostly employed for image enciphering because of their better performance, high degree of sensitivity to slight changes in initial conditions, higher degree of randomness, enormous key space, aperiodicity, and high security. They can be designed by taking advantage of the more complex behavior of chaotic signals. There are chaotic maps of different dimensions including one dimensional, multidimensional and cascade chaotic system. The performance of each chaotic system can be determined through the use of statistical and security analysis [17], [49], [48]. But they don't fit into a resource-constrained environment, like the edge of a network. Hence, we designed a lightweight video-frame enciphering method based on a chaotic map and validated its security and performance.

C. Machine Learning in Preserving Privacy

Object detection plays very pronounced role in the process of creating a privacy-preserving surveillance system. Privacy attributes or some special behavioral patters on videos or images are detected by DNNs. The ImageNet challenge [33] has revolutionized the development of convoluted neural networks (CNN). Since then, a number of more accurate convoluted networks like VGG-Net [36], Res-Net [20], and many others [7], [25], [38], [39] were developed based on deeply involved networks. These types of networks are convenient for cloud computing or server-based deployment.

Hence, there are efforts to solve the privacy problems in surveillance systems using efficient object-detectors. Researches have attempted to leverage the advancement in machine learning and object-detection technologies to localize privacy-sensitive objects and provide privacy-setting recommendations to social media users [1], [6], [42], [46]. Others like [37] tried to create a platform for offloading tasks from mobiles to nearby cloud servers for private objects detection. These works give insight about the privacy attributes deemed private by many people and some methods pertinent to video analytics. However, they do not provide solutions for the privacy problems in surveillance systems.

III. PriSev System Architecture

As portrayed in Fig. 1, the proposed PriSev system mainly comprises sensory, network, and filtering fog-/cloud-server layers. Besides, it includes a storage area and viewing terminals. The sensory layer refers to the edge where surveillance cameras are placed and video frames are created. It checks whether the frames contain objects of interest, and enciphers those frames that contain some objects. The network layer is the channel over which the frames and control messages are exchanged in encrypted form between the sensory and the filtering fog/cloud servers layers or between the storage and remote users of stored videos. The filtering layer is the heart of the PriSev system where a policy-based frame discrimination is performed by the help of a model built based on DNNs. Frames that contain suspicious behaviors like brandishing a gun or throwing a fist are forwarded to the storage disks and live viewing surveillance operation centers (SOC). Only video frames that are deemed important for later use are stored.

A. Lightweight Video Frame Enciphering Mechanism

The traditional cryptographic mechanisms like AES and RSA are not quite suitable for real-time video frames/images encryption at the edge. In the PriSev scheme, a lightweight dynamic chaotic image enciphering (DyCIE) scheme is introduced based on a discrete chaotic dynamic system [12], [26], [30], [43] that can run at the edge in real time. It is lighter and sensitive to any slight variation in the values of a key. As stated in Eq. (1), the key is defined as a list data type and its elements constitute the coefficients and initial value of the chaotic dynamic system. This chaotic encryption scheme

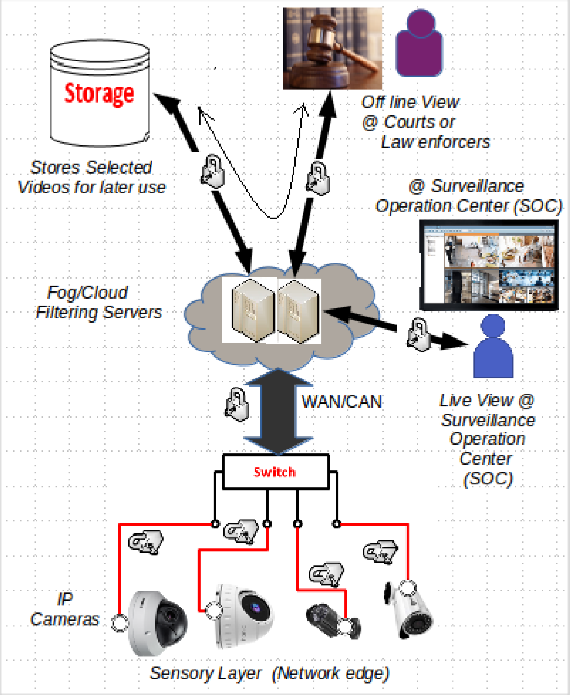

Fig. 1. PriSev System Architecture.

Algorithm 1 DyCIE

```
@ Edge Camera: a sending end
$f_c \leftarrow$ cam.video()
$C \leftarrow []$
$w,\ h \leftarrow$ $\mathrm{f_c}$.size
procedure KEYGEN
    K ← secRand([$k_0$, $k_1$, $k_2$, $k_3$, $k_4$])
    return K
$K \leftarrow keyGen()$
procedure GENCHAOS(K)
    $i \leftarrow 0$
    $C_0 \leftarrow$ K[0]
top:
    $temp \leftarrow$ K[1]*$C_0$(1-K[2]*$C_0$)
    $C[i] \leftarrow$ ceil(temp*K[3])
    $C_0 \leftarrow$ temp*K[4]
    $i$+=$1$
    if $i$ == $(h*w)$ then $break$
goto top:
    return C
$C \leftarrow genChaos(K)$
procedure ENCFRAME(C, $\mathrm{f_c}$)
    $f_{enc} \leftarrow f_c \oplus C$
    return $\mathrm{f_{enc}}$
@ Filtering Server: a receiving end
$K,\ f_{enc} \leftarrow$ received from an edge camera
$C \leftarrow genChaos(K)$
procedure DECFRAME(C, $\mathrm{f_{enc}}$)
    $f_c \leftarrow f_{enc} \oplus C$
    return $\mathrm{f_c}$
```

is much more efficient for video encryption.

$$\begin{aligned} K &= [k_0,\ k_1,\ k_2,\ k_3,\ k_4] \\ C_0 &= k_0 \\ tmp &= k_1 * C_0(1 - k_2 * C_0) \\ C &= tmp * k_3\ (scaling) \\ C_0 &= tmp * k_4\ (update) \end{aligned} \tag{1}$$

Eq. (1) is recursive in that the new chaotic values of the dynamic system in the equation are generated based on previous values multiplied by some key parameters. It starts with an initial value C_0 and coefficients k_1, and k_2. Then, it is iterated until a maximum number of iterations equal to the product of the height and width of an input frame is reached. Extensive experiments and analyses have been carried out to obtain the secure ranges of the initial value k_0, and the coefficients k_1, k_2, k_3, and k_4, which are generated in the form of encryption key. Based on our experimental study, k_0 can take any double-precision floating value between (0.2, 0.9). For k_1, the secure range that generates secure random chaos is found out to be between (3.89, 4.0). k_2 and k_4 can assume floating values in the range of (0.99, 1.0) and k_3 must be within the range of (254, 255).

As outlined by Algorithm 1, the DyCIE scheme comprises three major components: key generation, chaos generation, and frame encryption. The key is defined as a list data type whose elements are the initial value and the coefficients of the enciphering equations. Their values fall within the defined secure ranges and are generated using a secure cryptographic random generator. The chaos generator, which takes the key and the current video frame dimensions as inputs, produces a random chaos that in turn is used as a key to encipher the video frames. To enhance the security of the scheme and to cut down on its computational complexity, the enciphering process is performed channel-wise. In other words, as portrayed in Fig. 2, the frame to be encrypted is efficiently split into the three color channels, namely red (R), green (G), and blue (B). Then, enciphering of the three color channels is performed in parallel and the results are combined together before transmission. The three keys are also appended into a single list before transmission in the same order as the color channels (RGB).

The enciphering process is performed at the edge and the inverse process is performed at the filtering server in the fog/cloud layer. Communications between storage and server, and between viewers and server are also enciphered. To uncover the encrypted frames, the same chaotic images are generated at the receiving side using the corresponding keys.

B. Foreground Object Detection At The Edge

The sensory layer of the PriSev system comprises smart CCTV cameras for capturing the footage of the area under surveillance. It is ideal to deploy privacy-preserving measures at the point where the videos are created. However, due to resource constraints, the cameras cannot afford DNN-based compute-intensive tasks like object classification and localization, and image encryption in real-time. To optimize the performance, in the PriSev scheme, the edge cameras are only in charge of frame enciphering task to ensure end-to-end

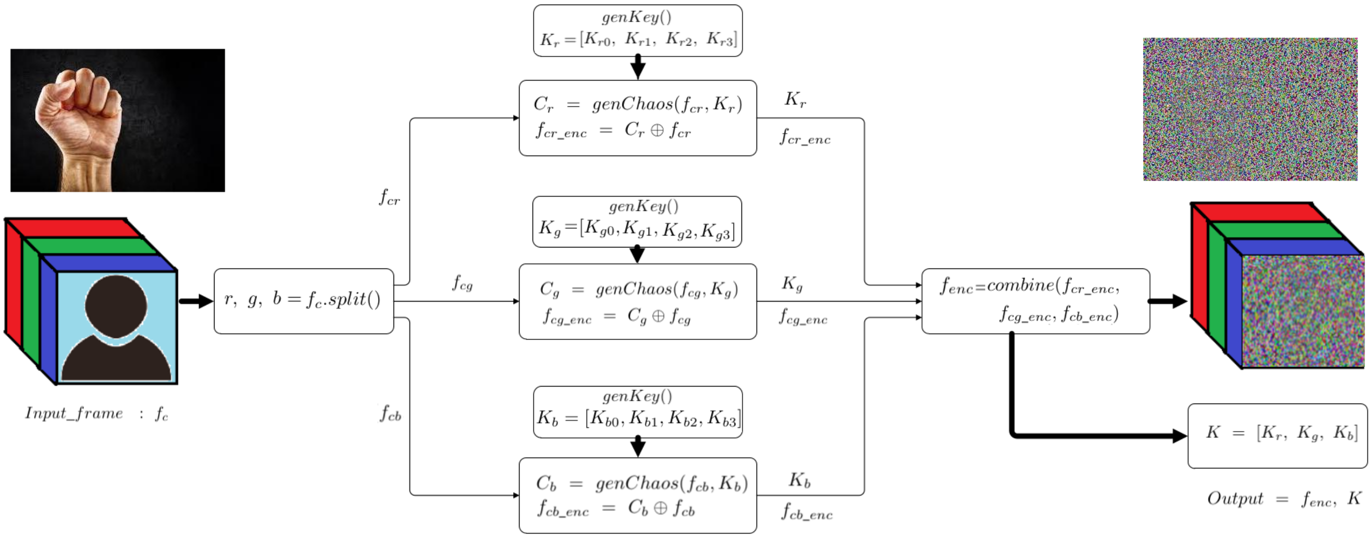
genKey()
$K_r = [K_{r0}, K_{r1}, K_{r2}, K_{r3}]$
$C_r = genChaos(f_{cr}, K_r)$
$f_{cr\_enc} = C_r \oplus f_{cr}$
K_r
f_{cr_enc}
f_{cr}
genKey()
$K_g = [K_{g0}, K_{g1}, K_{g2}, K_{g3}]$
$r, g, b = f_c.split()$
f_{cg}
$C_g = genChaos(f_{cg}, K_g)$
$f_{cg\_enc} = C_g \oplus f_{cg}$
K_g
f_{cg_enc}
$f_{enc} = combine(f_{cr\_enc}, f_{cg\_enc}, f_{cb\_enc})$
$Input\_frame : f_c$
f_{cb}
genKey()
$K_b = [K_{b0}, K_{b1}, K_{b2}, K_{b3}]$
$C_b = genChaos(f_{cb}, K_b)$
$f_{cb\_enc} = C_b \oplus f_{cb}$
K_b
f_{cb_enc}
$K = [K_r, K_g, K_b]$
$Output = f_{enc}, K$

Fig. 2. Channel-wise Frame Enciphering: Color channels R, G, and B are enciphered in parallel.

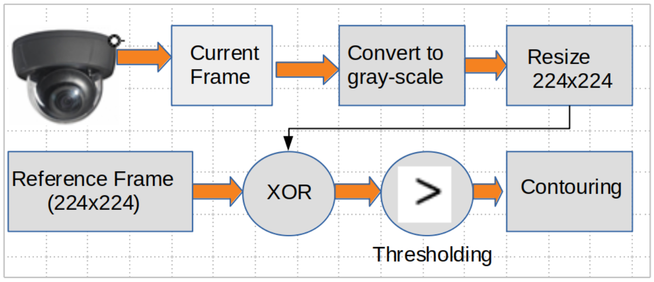

Fig. 3. Checking the presence of object in a frame

privacy, and other compute-intensive tasks like DNN-based frame discrimination are assigned to the fog/cloud server.

Yet enciphering every frame created at the edge is so costly. Image encryption is time consuming and it degrades the performance of the system. In terms of privacy, there is no benefit to scramble frames containing no objects. Hence, a preliminary lightweight foreground object detection mechanism is enforced at the edge for selective frame scrambling.

$$f_D = \{(x, y) | f_c(x, y) \ xor \ f_r(x, y)\} \tag{2}$$

In Eq. (2), f_r is the reference frame, f_c is the current frame, and (x, y) refers to the pixel values of a two-dimensional (2D) image. The reference/background frame f_r is the first frame shot by the camera. Then, every frame captured by the camera is compared with f_r to check if there is any difference or relative motion. To make the process computationally efficient, the f_r is resized to a 224×224 gray-scale image at the edge and every current frame f_c as well is resized and converted to gray-scale. Figure 3 illustrates the steps in the process of generic foreground object detection. As described by Algorithm 2, only frames containing objects, irrespective of their types, are scrambled. That is, a frame is encrypted only if there is a pixel difference of at least 0.35% between the reference and current frame. The 0.35% was set after a thorough study of the change of pixels due to object(s) motion and environmental factors.

Algorithm 2 Foreground Object Detection

```
@ Edge Camera
$f_r$ ← Reference frame
$w, h$ ← $f_r$.size
$f_c$ ← cam.video()
$f_{resized}$ ← $f_c$.resize((h, w))
procedure CHECKOBJECT($f_r$, $f_c$, $f_{resized}$)
    $f_D$ ← $f_{resized}$(x, y) $xor$ $f_r$(x, y)
    diff ← percentage of pixel change
    if diff $\geq 0.35\%$ then
        result ← encFrame(C, $f_c$)
    else
        Continue to next frame
    return result
```

C. Frame Discrimination

One glaring inefficiency of the current surveillance system is the lack of intelligence to differentiate between innocent and illicit acts of individuals. This type of non-selective practice increases the likelihood of privacy breaches. Our PriSev scheme focuses on identifying suspicious or aggressive behavioral patters and filtering out those video frames containing non-aggressive gestures or acts in lieu of detecting privacy-sensitive attributes and masking them. This paper primarily focuses on detecting aggressive gestures like gun-brandishing and raising a hand with a formed fist vertically or horizontally.

Figure 4 portrays the proposed DNN model trained using image datasets collected from various sources like Kaggle, soft computing, Edgecase.ai, and EgoGesture Dataset. When frames arrive at the fog/cloud server from the edge cameras, they are first unscrambled and fed to the model. The model detects guns held by a person caught on a camera and any

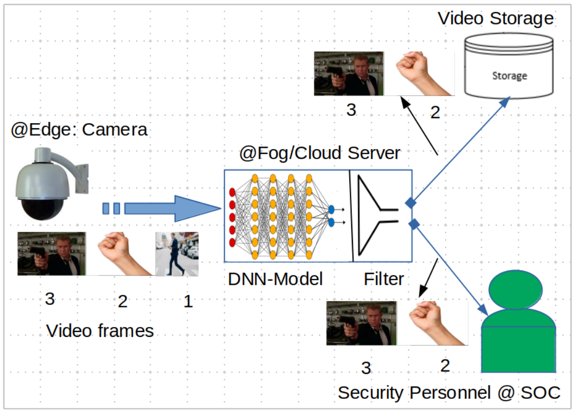

Fig. 4. Object Detection and Frame Filtering

horizontal or vertical movement of a hand with a formed fist. As described in Algorithm 3, every frame is checked whether it contains the objects of interest or not. Those innocuous frames that don't contain offensive behavioral patterns are filtered out. They are not sent to the security personnel for viewing and analytics. Only those that contain offensive patterns are viewed and stored for later use. This improves the privacy of the surveillance practices by filtering out frames that contain innocuous individuals.

Algorithm 3 Frame filtering @ Fog/Cloud Server

```
vid ← videoCapture()
out ← videoWriter(name, compressionFormat, (w,h))
while true do
    status, frame ← vid.read()
    if status then
        objects ← dnnModel(frame)
        if len(objects) == 0 then
            continue
        else
            show(frame) @ SOC
            out.write(frame) @ storage
    else
        break
```

D. Agent-Based Key Management

The use of a scrambling process in the IP CCTV surveillance system requires an efficient key exchange management. Hence, a client-server architecture based lightweight agents for key exchange are introduced. As illustrated in Fig. 5, there are four types of agents namely camera, server, live terminal, offline terminal, and storage agents. The camera agent stores the public key of the server agent, and generates a session key for every frame scrambling. It also makes periodic security authentication of the camera to detect unauthorized changes. It informs the server if it detects any mismatch between the reference and new hashes. The server agent keeps record of the public keys of all clients for the purpose of secure session key exchange. The key exchange between the server and storage is handled by the storage agent. The live terminal agent unscrambles frames to be viewed live by security guards sitting in the surveillance operation. It keeps the public key of the server for recovering a session key that is used to decipher encrypted frames if the terminal is not directly connected to the server. The offline terminal agent is manually triggered whenever authorized users want to access stored videos. Hence, all agents but the offline terminal agent operate automatically.

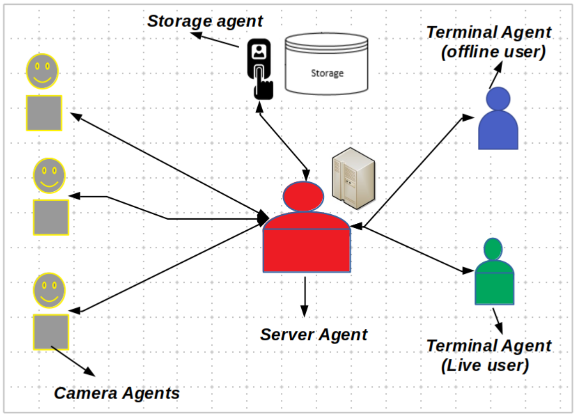

Fig. 5. Simplified Diagram of Agents Interaction.

In a scenario where there are a large number of connected devices, there is key proliferation and often creates a problem in the key-distribution management. However, it creates no problem for the agents are designed to leverage the modular deployment of IP CCTV cameras. In the pragmatic deployment of CCTV cameras, a digital server known as Network Video Recorder (NVR) sits on the surveillance network to receive live image/video-frame streams and to record them digitally to a hard disk. It often comprises a video surveillance management software, and limited number of dedicated channels through which every IP camera is connected on a one-to-one basis. Hence, large number of IP-cameras are not stuffed to a single NVR, rather they are divided into multiple groups where each group is connected to an NVR server. For further video analytics, powerful virtual or physical cloud machines capable of performing hundreds of frames per second (fps) are connected and configured to the NVRs.

IV. Performance Analysis

This section presents the functionality test, security analysis, performance analysis, computational overheads analysis, and comparison with counterpart methods. The performance of the scrambling process was carried out on an edge device, Raspberry PI 4. The DNN model was tested on a powerful Predator Triton 700-A laptop connected to the edge device.

A. Dynamic Chaotic Image Enciphering Scheme

To analyze the performance and security of the proposed DyCIE algorithm, a number of standard metrics are considered, including time complexity, key space, key sensitivity, Peak Signal to Noise Ratio (PSNR), Number of Pixels Change Rate (NPCR), Unified Average Changing Intensity (UACI), correlation, and Entropy. Figure 6 shows

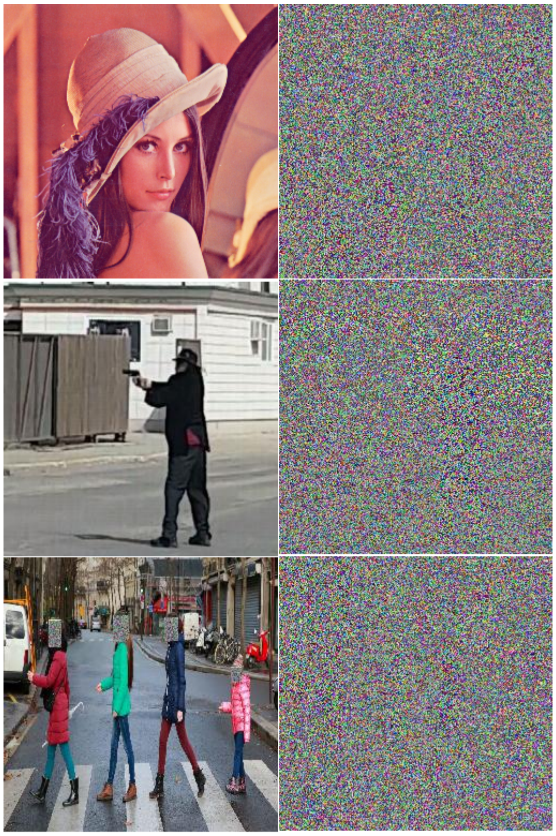

Fig. 6. Visual Assessment of Three Scrambled Images/Frames.

the functional test results of the scrambling scheme. All input images are successfully encrypted.

1) Computational Speed Analysis: For the time complexity analysis, the input frame/image employed is a 480P RGB color. The speed analysis refers to the amount of time consumed for encrypting a frame or image. It includes time required for preliminary scanning for the presence of an object, splitting the input frame into the three colors, encrypting each channel in parallel, and combining them together. Our scheme can process 10.274 fps on average.

2) Visual Analysis: The output of a good scrambling scheme should be highly random that gives no visual information to an adversary. As illustrated by the three input frames along with their scrambled versions in Fig. 6, the scrambled forms give no visual clue about their corresponding plain input images. Hence, the output of the proposed scheme reveals no visual information. In fact, we enciphered each of the three input images in Fig. 6 more than 100,000 times using different keys in an automated manner and all passed the visual and other assessments.

3) Key Space Analysis: The image scrambling is performed color channel-wise in parallel. As shown in Fig. 2, three keys are generated concurrently. Eq. (3) depicts that each channel's session key comprises five double-precision floating point values. Roughly speaking, a single-precision number requires 32 bits and its double-precision counterpart is a 64 bits long number. Hence, the channel key size is 320 (60×5) bits long. The DyCIE algorithm is highly sensitive even to a single bit change and partial guessing of the key cannot help at all.

$$\begin{aligned} K &= [K_R,\ K_G,\ K_B] \\ K_R &= [k_0,\ k_1,\ k_2,\ k_3,\ k_4] \ = \ 320\ bits \\ K_G &= [k_0,\ k_1,\ k_2,\ k_3,\ k_4] \ = \ 320\ bits \\ K_B &= [k_0,\ k_1,\ k_2,\ k_3,\ k_4] \ = \ 320\ bits \end{aligned} \tag{3}$$

At present, more powerful and technologically superior exascale supercomputers are on the make. If we assume that the exaFLOPS computer can perform 10^{15} decryptions per μs. Then, a 320-bit long key can be exhaustively searched in 7.6×10^{67} years with this machine, as computed in Eq. (4). The 128 bit key space is often considered as the lower boundary of a secure key space in the practice of symmetrical cryptography.

$$\begin{aligned} KeySpace &= 2^{320}\ bits \\ Time\ required &= \frac{2^{320} \times 10^{-15} \times 10^{-6}}{365 \times 24 \times 3600} \\ &= 7.6 \times 10^{67}\ years \end{aligned} \tag{4}$$

4) Key Sensitivity Analysis: It is the measure of how much different cipher a slightly different key can produce. We repeatedly tested the proposed scheme using a key and another one differing by a single bit. The DyCIE algorithm is highly sensitive to a slight key change and the average number of pixels change rate in the ciphers is about 99.7% corroborating that it is resistant to differential attacks.

5) Peak Signal to Noise Ratio Analysis: PSNR is defined via the mean squared error (MSE). Given a plain $W \times H \times 3$ RGB image, I, and its noisy/scrambled version K, the MSE and PSNR are defined by Eq. (5.

$$\begin{aligned} MSE &= \frac{1}{L} \sum_{i=0}^{W-1} \sum_{j=0}^{H-1} \sum_{k=0}^{2} [I(i,j,k) - K(i,j,k)]^2 \\ L &= W \times H \times 3 \\ PSRN &= 20 \times log_{10}(\frac{MAX_I}{\sqrt{MSE}}) \end{aligned} \tag{5}$$

Here, MAX_I is the maximum possible pixel value of the plain image. When the pixels are represented using eight bits per sample, this is 255. Unlike the measure of the quality of reconstruction of lossy compression, the PSNR of a good scrambling scheme is expected to be low because the MSE of the plain and scrambled images is expected to be higher. The average PSNR of the proposed scheme is 10.45dB.

6) Histogram Analysis: It is the frequency description of each unique pixel values of an image. Figure 7 shows the histograms of the cipher of the famous Lenna picture shown on right top of Fig. 6. It verifies that the scrambling scheme is robust against any statistical histogram attack in that the scrambled images have uniform histogram.

7) Correlation Analysis: It measures the correlation among the adjacent pixels of an image. A good scrambling technique is expected to result in a scrambled image/frame with no correlation between adjacent pixels. The horizontal, vertical, and diagonal correlation analysis of our proposed model produces nearly zero correlation values for all test images substantiating that pixels that constitute enciphered frames are uncorrelated.

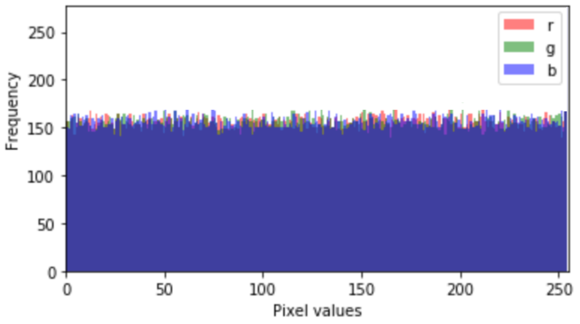

Fig. 7. Histogram Analysis.

8) Information Entropy: The information entropy $H(I)$ measures the amount of randomness in the information content of the scrambled image I containing N pixels, computed by Eq.(6). The ideal value of H depends on the number of bits used to represent the image picture elements. For an 8-bit pixel, it is $H(I) = 8$, and $H(I) = 16$ for a 16-bit pixel representation. Our scheme produces entropy values ranging from 7.9851 to 7.9984, signifying its security against entropy analysis attack.

$$H(I) = -\sum_{i=0}^{N-1} P(I_i) log_2(I_i) \tag{6}$$

The performance and security analyses demonstrate that the proposed DyCIE scheme is fast, robust, and secure. In addition, we have conducted comparative analysis with other popular cryptographic schemes like AES Chaining Block Cipher (AES-CBC) and recently proposed chaotic image enciphering mechanisms like [18], [40]. The AES-CBC is said to be one of the most secure and widely used data encryption standard. Our scheme has comparable results with AES-CBC in most cases except AES has a slight edge in terms of PSNR. However, the AES-CBC has lower pixel sensitivity and much longer computational time. In our efficient implementation for the analysis, the AES takes about 3.245 seconds to encrypt 480P color image whereas our DyCIE scheme takes about 97.33 milliseconds. The key sensitivity of AES is excellent but the key sensitivity of our scheme is better by about 0.00673. The double spiral scans and chaotic maps based image encryption [40] has very good security but it is not feasible for edge deployment because of its slow speed. Its efficient implementation exploiting available CPU resources can scramble only one 480P color frame in 1.95 seconds.

Furthermore, our scheme is coupled with simple preliminary object presence scanner at the edge. Scrambling every frame irrespective of whether it contains some objects or not wastes computational resources and degrades the performance. Hence, frames are passed through preliminary object-scanning which includes comparison with a reference frame of a camera to compute the difference. Then, only frames that contain any object are scrambled. Figure 8(a) shows that the incoming frame is the same as the reference frame and is transmitted in clear form. Figure 8(b) shows a frame that contains an object and is different from the reference frame. As a result, it is scrambled. Figure 8(c) demonstrates the process of preliminary check for the presence of an object. Figure 9 shows a variation in the number of frames performed every second. On average, the proposed DyCIE scheme can scramble 10.274 fps.

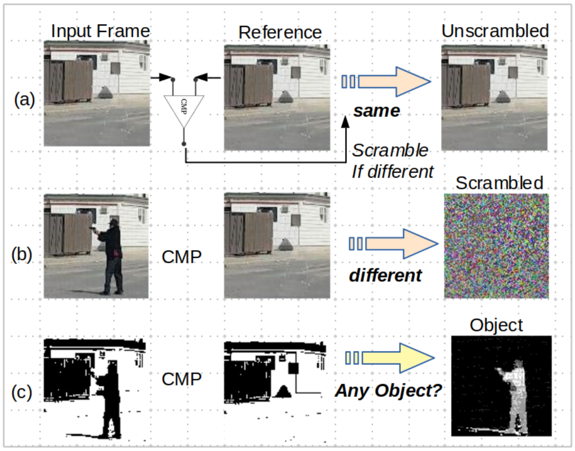

Fig. 8. Preliminary Object Scanning and enciphering.

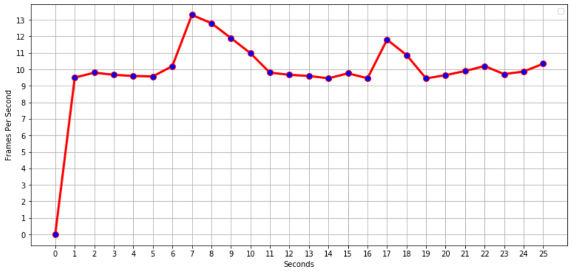

Fig. 9. The number of Frames processed every second (fps) at the edge.

B. Detection of Aggressive Behavioral Patterns

In this work, we have created a deep learning model based on a dataset containing thousands of gun-brandishing and fist throwing images garnered from a number of sources. Stitched samples are portrayed in Fig. 10. It is capable of classifying frames/images based on the behavioral pattern they contain. Those frames containing aggressive behaviors like gun-wielding, punching or fist-throwing are classified as "aggressive" frames, which are forwarded to the viewing center and recorded on disk for later use. Those frames classified as "innocuous" are discarded or skipped. The proposed PriSev scheme maximizes the privacy and usability by making an efficient trade-off between the two behaviors.

In the real world, video streams created by a number of IP cameras are sent to digital network video recorder (NVR) servers. One NVR alone can support multiple cameras connected one per channel. It is powerful enough to play back the video streams coming via every channel at least 30fps. In our experiment on a *Predator Triton 700-A laptop*, the DNN-model is able to run at 43 fps on average. The test results show that our model successfully classifies the frames with an average accuracy of 98.97%. The diversity of the dataset used for training the model contributed to the higher accuracy in addition to the careful design of the various layers. Hence, we are able to classify and filter incoming frames into aggressive

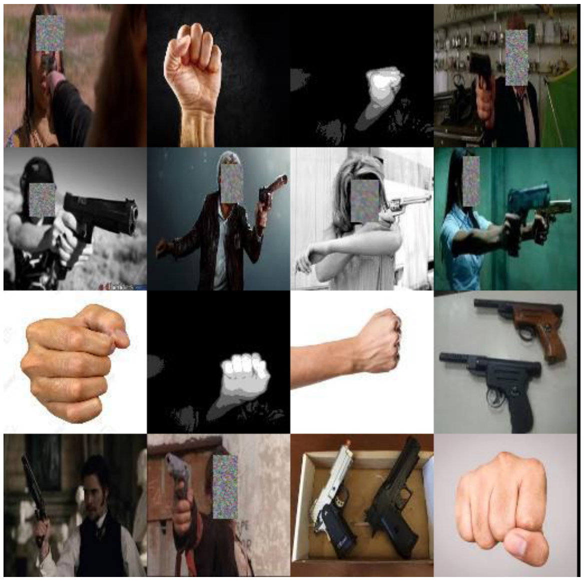

Fig. 10. Sample Dataset (Faces of individuals are masked for privacy.)

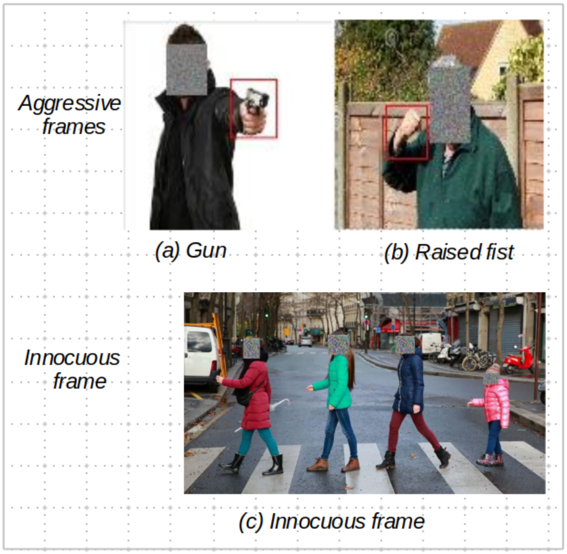

Fig. 11. Frame Classification: (a) and (b) Aggressive, and (c) Innocuous.

and innocuous. Figure 11 shows some sample illustrations.

C. Overhead Analysis and Scalability

In our PriSev scheme, the DNN-based frame filtering is performed on a fog/cloud server, and the preliminary object detection and enciphering process are performed at the edge. The edge is the performance bottleneck. At the server end hundreds of frames could be processed every second but at the edge only fewer frames are processed due to resource constraints. With our scheme enforced, the average throughput at the camera edge is 10.274 fps. That is, 97.33ms is required to process every 480P frame at the edge. Our scheme is more efficient than existing schemes and it ensures end-to-end privacy.

In terms of bandwidth, the enciphering scheme does not have much effect because the ciphered frame has still the same size as the original plain frame. The only overhead is the deciphering key, which adds 3 × 320 bits for every frame. The approximate bandwidth of a video streaming can be calculated by taking the frame size, and the frame rate into consideration. It mainly depends on the frame resolution, color depth, and fps. The bandwidth requirements of a clear frame, an enciphered frame, and the overhead are computed by using Eqs. (7), (8), and (9). Hence, the impact of the key on the bandwidth is insignificant (only 0.31%).

$$\begin{aligned} F_{clear}\ BW =&\ 640 \times 480 \times 3 \times 10.274 fps \\ =&\ 9.0298828 Mbps \end{aligned} \tag{7}$$

$$\begin{aligned} F_{cipher}\ BW =&\ (640 \times 480 \times 3 + 3 \times 320) \times 10.274 fps \\ =&\ 9.0392889 Mbps \end{aligned} \tag{8}$$

$$\begin{aligned} BW\ overhead =&\ (\frac{9.03928894 - 9.0298828}{9.0298828}) \times 100 \\ =&\ 0.104\% \end{aligned} \tag{9}$$

The PriSev scheme does not affect the scalability of the CCTV system. It can support as many cameras as needed with small overheads. The performance issue is confined to individual cameras at the edge of the network. In the real world, the deployment of the surveillance system is modular. An NVR can have 8 to 512 channels that enable it to support 8 to 512 IP cameras. Hence, our scheme has no impact on scalability when the number of cameras is increased from 8 to 512.

D. Limitations

The limitation of the PriSev is some penalty in processing speed at the edge of the network due to limited computing resources. The experimental analysis and implementation were carried out on a relatively slower edge device, the Raspberry PI 4, and the average frame processing capability of the PriSev is about 10 fps. However, the speed could be drastically improved if much faster edge devices like NVIDIA Jetson Nano are employed.

V. Conclusions

In this paper, we proposed PriSev, a privacy-preserving selective video surveillance scheme, which comprises a lightweight dynamic chaotic image enciphering (DyCIE) algorithm integrated with a preliminary object scanner, a DNN based frame classification model, and a very light agent-based key and control messages exchange management. The DyCIE algorithm provides an end-to-end enciphering of video frames that contain objects to prevent privacy breaches due to interceptions. The DNN-based model classifies frames as "aggressive" and "innocuous" creating an auspicious condition for filtering out frames that contain no criminal or aggressive behavioral patterns, which prevents privacy breaches due to possible abuses or leaks by authorized people in charge of the surveillance system. All security and experimental analyses corroborate that the proposed PriSev scheme attains very good efficiency and security. Specifically, the DyCIE enciphering is much faster than existing popular methods at the edge.

REFERENCES

[1] M. Al-Rubaie and J. M. Chang, “Privacy-preserving machine learning: Threats and solutions,” *IEEE Security & Privacy*, vol. 17, no. 2, pp. 49–58, 2019.

[2] A. Artusi, R. K. Mantiuk, T. Richter, P. Hanhart, P. Korshunov, M. Agostinelli, A. Ten, and T. Ebrahimi, “Overview and evaluation of the jpeg xt hdr image compression standard,” *Journal of Real-Time Image Processing*, vol. 16, no. 2, pp. 413–428, 2019.

[3] D. Banisar, *Privacy and Human Rights...: An International Survey of Privacy Laws and Developments*. Electronic Privacy Information Center, 1999.

[4] W. C. Bennett, *Civilian drones, privacy, and the federal-state balance*. Center for Technology Innovation at Brookings Washington DC, 2014.

[5] A. Cavallaro, “Privacy in video surveillance [in the spotlight],” *IEEE Signal Processing Magazine*, vol. 2, no. 24, pp. 168–166, 2007.

[6] K. Chaudhuri and C. Monteleoni, “Privacy-preserving logistic regression,” in *Advances in neural information processing systems*, 2009, pp. 289–296.

[7] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, “Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp. 834–848, 2017.

[8] N. Chen, Y. Chen, E. Blasch, H. Ling, Y. You, and X. Ye, “Enabling smart urban surveillance at the edge,” in *2017 IEEE International Conference on Smart Cloud (SmartCloud)*. IEEE, 2017, pp. 109–119.

[9] S. Çiftçi, P. Korshunov, A. O. Akyüz, and T. Ebrahimi, “Using false colors to protect visual privacy of sensitive content,” in *Human Vision and Electronic Imaging Xx*, vol. 9394. International Society for Optics and Photonics, 2015, p. 93941L.

[10] A. Costin, “Security of cctv and video surveillance systems: Threats, vulnerabilities, attacks, and mitigations,” in *Proceedings of the 6th international workshop on trustworthy embedded devices*, 2016, pp. 45–54.

[11] F. Dufaux, “Video scrambling for privacy protection in video surveillance: recent results and validation framework,” in *Mobile Multimedia/Image Processing, Security, and Applications 2011*, vol. 8063. International Society for Optics and Photonics, 2011, p. 806302.

[12] M. J. Feigenbaum, “The onset spectrum of turbulence,” *Physics Letters A*, vol. 74, no. 6, pp. 375–378, 1979.

[13] A. Fitwi, Y. Chen, and N. Zhou, “An agent-administrator-based security mechanism for distributed sensors and drones for smart grid monitoring,” in *Signal Processing, Sensor/Information Fusion, and Target Recognition XXVIII*, vol. 11018. International Society for Optics and Photonics, 2019, p. 110180L.

[14] A. Fitwi, Y. Chen, and S. Zhu, “A lightweight blockchain-based privacy protection for smart surveillance at the edge,” in *2019 IEEE International Conference on Blockchain (Blockchain)*. IEEE, 2019, pp. 552–555.

[15] ——, “No peeking through my windows: Conserving privacy in personal drones,” *arXiv preprint arXiv:1908.09935*, 2019.

[16] A. H. Fitwi, D. Nagothu, Y. Chen, and E. Blasch, “A distributed agent-based framework for a constellation of drones in a military operation,” in *2019 Winter Simulation Conference (WSC)*. IEEE, 2019, pp. 2548–2559.

[17] A. H. Fitwi and S. Nouh, “Performance analysis of chaotic encryption using a shared image as a key,” *Zede Journal*, vol. 28, pp. 17–29, 2011.

[18] G. Hanchinamani and L. Kulkarni, “An efficient image encryption scheme based on a peter de jong chaotic map and a rc4 stream cipher,” *3D Research*, vol. 6, no. 3, p. 30, 2015.

[19] B. Harris and R. Hunt, “Tcp/ip security threats and attack methods,” *Computer communications*, vol. 22, no. 10, pp. 885–897, 1999.

[20] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[21] U. Hessler, “Museum camera films merkel’s apartment in security breach,” *https://www.dw.com/en/museum-camera-films-merkels-apartment-in-security-breach/a-1945643*, 2006 (accessed on April 2, 2019).

[22] P. Korshunov and T. Ebrahimi, “Using face morphing to protect privacy,” in *2013 10th IEEE International Conference on Advanced Video and Signal Based Surveillance*. IEEE, 2013, pp. 208–213.

[23] V. Kumar and J. Svensson, *Promoting social change and democracy through information technology*. IGI Global, 2015.

[24] L. Lin and N. Purnell, “A world with a billion cameras watching you is just around the corner,” *The Wall Street Journal*, 2019.

[25] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, “Ssd: Single shot multibox detector,” in *European conference on computer vision*. Springer, 2016, pp. 21–37.

[26] R. M. May, “Simple mathematical models with very complicated dynamics,” *Nature*, vol. 261, no. 5560, pp. 459–467, 1976.

[27] E. M. Newton, L. Sweeney, and B. Malin, “Preserving privacy by de-identifying face images,” *IEEE transactions on Knowledge and Data Engineering*, vol. 17, no. 2, pp. 232–243, 2005.

[28] J. R. Padilla-López, A. A. Chaaraoui, and F. Flórez-Revuelta, “Visual privacy protection methods: A survey,” *Expert Systems with Applications*, vol. 42, no. 9, pp. 4177–4195, 2015.

[29] W. B. Pennebaker and J. L. Mitchell, *JPEG: Still image data compression standard*. Springer Science & Business Media, 1992.

[30] S. Phatak and S. S. Rao, “Logistic map: A possible random-number generator,” *Physical review E*, vol. 51, no. 4, p. 3670, 1995.

[31] L. Rakhmawati *et al.*, “Image privacy protection techniques: A survey,” in *TENCON 2018-2018 IEEE Region 10 Conference*. IEEE, 2018, pp. 0076–0080.

[32] S. Ribaric, A. Ariyaeeinia, and N. Pavesic, “De-identification for privacy protection in multimedia content: A survey,” *Signal Processing: Image Communication*, vol. 47, pp. 131–151, 2016.

[33] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, “Imagenet large scale visual recognition challenge,” *International journal of computer vision*, vol. 115, no. 3, pp. 211–252, 2015.

[34] M. Saini, P. K. Atrey, S. Mehrotra, and M. Kankanhalli, “W 3-privacy: understanding what, when, and where inference channels in multi-camera surveillance video,” *Multimedia Tools and Applications*, vol. 68, no. 1, pp. 135–158, 2014.

[35] A. Senior, S. Pankanti, A. Hampapur, L. Brown, Y.-L. Tian, A. Ekin, J. Connell, C. F. Shu, and M. Lu, “Enabling video privacy through computer vision,” *IEEE Security & Privacy*, vol. 3, no. 3, pp. 50–57, 2005.

[36] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.

[37] C. Streiffer, A. Srivastava, V. Orlikowski, Y. Velasco, V. Martin, N. Raval, A. Machanavajjhala, and L. P. Cox, “eprivateeye: To the edge and beyond!” in *Proceedings of the Second ACM/IEEE Symposium on Edge Computing*. ACM, 2017, p. 18.

[38] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, “Inception-v4, inception-resnet and the impact of residual connections on learning,” in *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[39] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[40] Z. Tang, Y. Yang, S. Xu, C. Yu, and X. Zhang, “Image encryption with double spiral scans and chaotic maps,” *Security and Communication Networks*, vol. 2019, 2019.

[41] B. Varghese and R. Buyya, “Next generation cloud computing: New trends and research directions,” *Future Generation Computer Systems*, vol. 79, pp. 849–861, 2018.

[42] J. Wang, B. Amos, A. Das, P. Pillai, N. Sadeh, and M. Satyanarayanan, “Enabling live video analytics with a scalable and privacy-aware framework,” *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 14, no. 3s, p. 64, 2018.

[43] E. W. Weisstein, “Feigenbaum constant,” *delta*, vol. 5, p. 6, 2003.

[44] T. Winkler and B. Rinner, “Security and privacy protection in visual sensor networks: A survey,” *ACM Computing Surveys (CSUR)*, vol. 47, no. 1, pp. 1–42, 2014.

[45] R. Xu, S. Y. Nikouei, Y. Chen, A. Polunchenko, S. Song, C. Deng, and T. R. Faughnan, “Real-time human objects tracking for smart surveillance at the edge,” in *2018 IEEE International Conference on Communications (ICC)*. IEEE, 2018, pp. 1–6.

[46] J. Yu, B. Zhang, Z. Kuang, D. Lin, and J. Fan, “iprivacy: image privacy protection by identifying sensitive objects via deep multi-task learning,” *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 5, pp. 1005–1016, 2017.

[47] L. Yuan and T. Ebrahimi, “Image privacy protection with secure jpeg transmorphing,” *IET Signal Processing*, vol. 11, no. 9, pp. 1031–1038, 2017.

[48] X. Zhang and X. Wang, “Chaos-based partial encryption of spiht coded color images,” *Signal Processing*, vol. 93, no. 9, pp. 2422–2431, 2013.

[49] Y. Zhou, L. Bao, and C. P. Chen, “A new 1d chaotic system for image encryption,” *Signal processing*, vol. 97, pp. 172–182, 2014.