

Novel Ransomware Hiding Model Using HEVC Steganography Approach

Iman Almomani^{1,2,*}, Aala AlKhayer¹ and Walid El-Shafai^{1,3}

¹Security Engineering Lab, Computer Science Department, Prince Sultan University, Riyadh, 11586, Saudi Arabia

²Computer Science Department, King Abdullah II School for Information Technology, The University of Jordan, 11942, Jordan

³Department of Electronics and Electrical Communications Engineering, Faculty of Electronic Engineering, Menoufia University, Menouf, 32952, Egypt

*Corresponding Author: Iman Almomani. Email: imomani@psu.edu.sa

Received: 14 March 2021; Accepted: 15 April 2021

Abstract: Ransomware is considered one of the most threatening cyberattacks. Existing solutions have focused mainly on discriminating ransomware by analyzing the apps themselves, but they have overlooked possible ways of hiding ransomware apps and making them difficult to be detected and then analyzed. Therefore, this paper proposes a novel ransomware hiding model by utilizing a block-based High-Efficiency Video Coding (HEVC) steganography approach. The main idea of the proposed steganography approach is the division of the secret ransomware data and cover HEVC frames into different blocks. After that, the Least Significant Bit (LSB) based Hamming Distance (HD) calculation is performed amongst the secret data's divided blocks and cover frames. Finally, the secret data bits are hidden into the marked bits of the cover HEVC frame-blocks based on the calculated HD value. The main advantage of the suggested steganography approach is the minor impact on the cover HEVC frames after embedding the ransomware while preserving the histogram attributes of the cover video frame with a high imperceptibility. This is due to the utilization of an adaptive steganography cost function during the embedding process. The proposed ransomware hiding approach was heavily examined using subjective and objective tests and applying different HEVC streams with diverse resolutions and different secret ransomware apps of various sizes. The obtained results prove the efficiency of the proposed steganography approach by achieving high capacity and successful embedding process while ensuring the hidden ransomware's undetectability within the video frames. For example, in terms of embedding quality, the proposed model achieved a high peak signal-to-noise ratio that reached 59.3 dB and a low mean-square-error of 0.07 for the examined HEVC streams. Also, out of 65 antivirus engines, no engine could detect the existence of the embedded ransomware app.

Keywords: Ransomware embedding; steganography; HEVC; LSB; hamming distance; applications; apk; stego; security; confidentiality



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1 Introduction

One of the main challenges facing the digital transformation of almost all our life aspects is cybersecurity attacks. Such attacks are launched in many different ways. A common source of attacks is malicious software (malware) that harms the users' devices and data. The cyberthreats of malware are characterized into several types such as Trojan, Spyware, and Adware [1]. Among the most threatening cyberattacks is Ransomware which is a form of malware that blocks access to the victim's device or data until a ransom is paid, consequently gaining an astonishing growth in causing monetary loss against individuals, businesses, and governments [2]. Generally, ransomware is categorized into two main types crypto-ransomware and locker-ransomware. The crypto-ransomware encrypts the user's sensitive information and requests payment to retrieve the decrypted data. On the other hand, the locker-ransomware blocks the interaction with the victim's device by displaying a lock screen window. Subsequently, the lock window is only removed after a ransom is paid [3]. However, the recent success in Ransomware results in the appearance of new families [4].

Many research solutions have been proposed to detect ransomware attacks [5–7]. These solutions have either utilized permissions [5] or API package calls [7] or both [8] to apply static or dynamic analysis for applications, whether benign or ransomware. Additionally, they have applied machine learning algorithms to build effective ransomware detection systems [6,9]. However, the current ransomware detection solutions have assumed that the application is visible to be analysed. They did not investigate the possibility of hiding this ransomware and making it difficult to apply static or dynamic analysis.

In the context of malware in general, there are some attempts by the developers to create well-established techniques to bypass the detection systems. One of the utilized techniques was malicious components or activity hiding using steganography, which concealed the presence and communication between the active malware application and the attacker [10]. The existing steganography techniques can be categorized according to how the hidden communication is implemented into three main groups [11]: (a) techniques that hide malware by mimicking benign software, (b) techniques that inject one or more component into the network traffic, and (c) techniques that hide the malware or part of its components in a digital media file. However, driven by the vigorous expansion of multimedia, video steganography is gaining momentum gradually. Furthermore, research interest increases in utilizing video streaming due to its low-quality loss and high embedding capacity. Specifically, the high-efficiency video coding (HEVC) standard [12,13] which provides a high bit-rate reduction.

The significant spread of ransomware and the possible advances of its anti-detection techniques creates an urgent need for further investigation in this field. Furthermore, applying steganography algorithms to embed ransomware applications increases ransomware risk with respect to individuals and businesses where anti-viruses software might fail in detecting the hidden ransomware. Accordingly, the major contributions of this paper can be summarized as follow:

- Deep investigation in the literature to check if there are any attempts to hide ransomware applications.
- Conduct a comparative analysis among existing techniques utilizing steganography in hiding malicious data.
- Propose an efficient, novel approach to hide complete ransomware using block-based HEVC steganography.
- Apply comprehensive subjective and objective tests to evaluate the proposed approach.

- Obtain high similarity between the original video and the corresponding video after embedding the ransomware; as part of the subjective tests' results.
- Achieve high performance in terms of 16 metrics used to assess the quality of the video after embedding the ransomware; as part of the objective tests' results.
- Bypass 65 well-known Antivirus engines by the embedded ransomware video; as part of the security tests.

The rest of the paper is organized as follows. Section 2 provides a comparison among previous suggested works on steganography. Section 3 presents the proposed HEVC steganography-based ransomware hiding approach. Section 4 presents the approach evaluation and results' discussions. Finally, Section 5 concludes the paper and suggests possible future work.

2 Literature Review

This section highlights different steganography techniques presented in the literature [14–28]. Tab. 1 presents a comparison among several proposed steganography schemes in terms of the proposed solution's main goal, the implemented steganography technique, the type of both cover media object and the secret message, and the used evaluation metrics.

Table 1: A comparison among several proposed steganography schemes

Authors	The aim	Steganography technique	Cover object	Secret message type	Evaluation metrics	Year
Liao et al. [18]	To hide data in multiple images to apply adaptive payload distribution	Image texture	Multiple images	Image	–	2020
Weng et al. [19]	To conceal a video inside another video by applying the deep neural network	Deep neural network	Video	Full-size video	PSNR, SSIM, VIF, RMSE, APD	2019
Sahu et al. [15]	To improve the efficiency of the steganography process by utilizing a dual-layer LSB hiding technique.	Dual-layer LSB	Image	Image	PSNR, SSIM, PDH	2020
Hindi et al. [16]	To enhance the security of the steganography process by utilizing two eight decimal digits keys	LSB	Image	Text	MSE, PSNR	2019
Liao et al. [17]	To adaptively partition the capacity of the secret data between the RGB channels of the cover image	Payload partition	RGB image	Image	–	2019
Bak et al. [25]	To implement a hidden communication system between the attacker and the active malicious software	StegBlocks	Network traffic	Text file	–	2018
Kazerooni et al. [26]	To propose a traffic masking technique to conceal the malware application identity	Generative adversarial network (GAN)	Network traffic	Malware app. traffic	RF XGBoost	2020

(Continued)

Table 1 (continued).

Authors	The aim	Steganography technique	Cover object	Secret message type	Evaluation metrics	Year
Kaushik et al. [27]	To analyze the performance of several malware detecting algorithms before and after implementing OSINTs steganography	–	Image	Malware app.	–	2020
Wang et al. [20]	To implement video steganography by utilizing the features of HEVC for cover selection	IPM	HEVC video	Text file	PSNR, SSIM, BDBR	2019
Lui et al. [21]	To combine three intra-frame prediction modes aiming to enhance the visual quality of the secret video	Three intra-frame prediction modes	HEVC video	Text file	PSNR, SSIM, BER	2020
Zhang et al. [22]	To increase the capacity of PU to enable hiding information in HEVC video without affecting its resolution	Modified exploiting modification direction unit	HEVC video	Text file	PSNR, BRI	2021
Galiano et al. [23]	To hide information in high-resolution HEVC videos without affecting the video quality	Luminance intra-blocks	HEVC video	Text file	PSNR, SSIM	2020

Sahu et al. [15] proposed a dual-layer steganography system by applying a reversible information hiding (RIH) technique utilizing the least significant bit (LSB) in the hiding process. In the first layer, each pixel of the secret image is hidden within two bits of the cover data by implementing the LSB matching algorithm resulting in a pair of intermediate pixels.

Subsequently, during the second layer of embedding, this pair was used to hide four bits of the secret information. According to the conducted evaluation experiments, applying reversible information hiding in dual-layer resulted in high efficiency in information hiding. Hindi et al. [16] also, used an image to hide a secret message. However, in their proposed work, they have utilized two keys of eight decimal digits in implementing the hiding/extraction process aiming to enhance the level of security.

Moreover, in [17], a three components (Red, Green, Blue) RGB channel-based secret data partition was proposed to adaptively allocate the capacity of the hidden message between the RGB channels to hide an image inside an image without affecting the performance of the hiding process. Besides using a single image as a cover object, Liao et al. [18] developed a multiple images steganographic system in which it utilizes the features of the image texture. To distribute the secret data in multiple images, they have implemented an adaptive payload distribution technique. Furthermore, two payload division methods were proposed; distortion distribution (ES-DD) and image texture complexity (ES-ITC). However, experiment results showed that the proposed scheme provides an enhanced security performance.

The advances in video coding applications have raised the interest in video steganography [19–23]. In [19], the authors concealed a video in another video by employing the inter-frame references of the cover video. While creating the steganographic video, a novel technique for modeling the temporal residual was implemented to fully benefit from the sparse characteristic of the differences between the inter frames. The authors in [20] implemented video steganography by utilizing the intra-prediction mode (IPM) feature of HEVC for cover selection. The stego video stream combined the prediction unit of HEVC and the coding unit to implement the cover selection process. Another application of HEVC steganography was proposed by [21] in which three intra-frame prediction modes were combined to enhance the visual quality of the carrier video. Zhang et al. [22] also used prediction units (PU) of HEVC in implementing video steganography. However, to overcome the capacity limitation of the PU, they have modified the exploiting modification direction; consequently, two prediction units were combined, thereby enlarging the PU capacity. An additional suggested solution that focuses on obtaining a high-resolution HEVC stego video streaming was proposed by [23]. In order to conceal information without affecting the video quality, they modified the bits of the luminance intra-blocks.

Steganography can be further used as a technique that hides malicious software to increase its undetectability level. Network steganography plays a significant role in malware information hiding in which one or more components of the malicious software are embedded in the network traffic [24]. In [25], a hidden communication system was implemented utilizing the StegBlocks technique to perform text communication between the attacker and the active malicious software. However, the attacker output text file is restricted to 23 kB. Another network steganography application was proposed by [26], which implemented a masking technique to conceal the malware application identity. In the proposed scheme, the tunnel generates fake traffic that simulates normal network traffic encapsulating the actual malicious traffic. In addition to network steganography, the digital media steganography technique was used to embed the malware software by altering the carrier media file structure. In [27], they developed a Malware utilizing Metasploit operating system, then they embedded the malware in an image. Subsequently, they performed detection analysis using Open Source Intelligence Tools (OSIT) such as VirusTotal. However, even though the result showed an enhancement in hiding the malware, yet, some virus scanners software detected the malware. Stergiopoulos et al. [28] used a different digital media carrier to embed malware. They injected malware apps via audio frequencies. However, the proposed system needs to meet certain conditions such as high speaker volume and low noise environment in order to extract the injected malware.

In the light of the above discussion, few works have been focused on utilizing and investigating steganography techniques to hide malware software. Furthermore, no research was found that has discussed the ability to embed ransomware software applications in digital media files. In this work, a novel system is proposed to utilize HEVC videos as a cover media to conceal the ransomware software applications with high efficiency.

3 Proposed HEVC Steganography-Based Ransomware Hiding Model

This section introduces and discusses the proposed block-based HEVC steganography approach for hiding ransomware applications. This ransomware hiding approach is built based on the image steganography algorithm presented in [29]. The proposed steganography-based hiding approach consists of two main processes; ransomware embedding process (REP) and ransomware extraction process (RExP) as shown in Fig. 1.

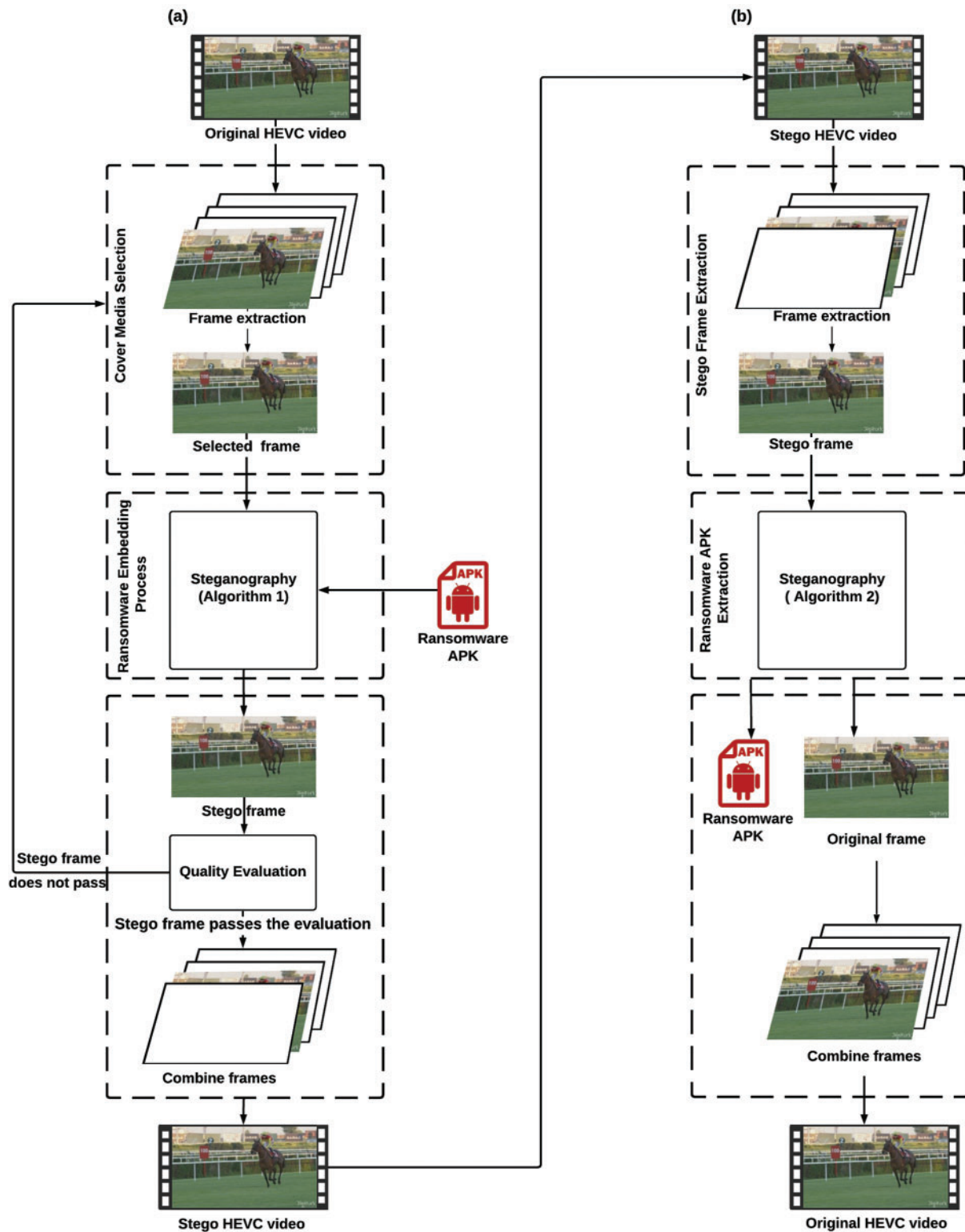


Figure 1: The proposed HEVC steganography-based ransomware hiding model (a) Ransomware Embedding Process (REP), (b) Ransomware Extraxting Process (RExP)

The REP starts by selecting the proper media cover (HECV video) and randomly extracting one of the video frames. The selection of cover video frames depends on the size of the ransomware sample and the capacity of the cover video. Therefore, prior to the embedding process, the proper video frame is selected based on its resolution in order to hide the ransomware sample without affecting the main features and quality of the cover frame. After that, the chosen frame is forwarded as an input to the embedding phase, as detailed in Algorithm (1).

Algorithm (1): Steps of the embedding phase

input: Plain HEVC frame.

–Divide both secret ransomware data and input cover HEVC frame into different blocks.

- Divide K -pixels cover video frame into different blocks of Q_j ($j = 1, 2, \dots, [K/(k+1)]$), where $k+1$ is the length of each block, $Q_{j1}, Q_{j2}, \dots, Q_{(j \ k+1)}$ refers to the pixel value within each block, $LSB_{j1}, LSB_{j2}, \dots, LSB_{jk}, LSB_{(j \ k+1)}$ refers to the LSB of each pixel value within each block, and $LSB_{(j \ k+1)}$ which is the marked bit of each block Q_j .
- Divide the secret ransomware data into blocks of S_j ($j = 1, 2, \dots, k$), where $S_{j1}, S_{j2}, \dots, S_{jk}$ refers to the binary bits of each block within the secret data.

for all divided blocks, **do**

–Initialize the LSB-based steganography cost function.

–Calculate the hamming distance (HD) using Eqs. (1) and (2) between the $LSB_{j1}, LSB_{j2}, \dots, LSB_{jk}, LSB_{(j \ k+1)}$ of Q_j , and their corresponding $S_{j1}, S_{j2}, \dots, S_{jk}$ bits of S_j . So, HD_j represents the number of total differences between the LSBs of Q_j and the related bits of S_j .

$$HD_j = k - \sum_{i=1}^k \Psi(LSB_{ji}, S_{ji}) \quad (1)$$

$$\Psi(LSB_{ji}, S_{ji}) = \begin{cases} 0 & \text{if } LSB_{ji} \neq S_{ji} \\ 1 & \text{if } LSB_{ji} = S_{ji} \end{cases} \quad (2)$$

if $HD_j \leq k/2$, **then**

–Set the marked bit $LSB_{(j \ k+1)}$ to 0.

–Determine the LSB-based steganography cost function using Eq. (3) to hide $(S_{j1}, S_{j2}, \dots, S_{jk})$ into $(Q_{j1}, Q_{j2}, \dots, Q_{jk})$. Therefore, the LSBs of Q_j will be $(S_{j1}, S_{j2}, \dots, S_{jk}, 0)$.

$$\Delta(LSB_{ji}, S_{ji}) = \begin{cases} 0 & \text{if } LSB_{ji} \neq S_{ji} \\ \pm 1 & \text{if } LSB_{ji} = S_{ji} \end{cases} \quad (3)$$

–Calculate the value of the stego pixel $P(Q_{ji}, S_{ji})$ using Eq. (4).

$$P(Q_{ji}, S_{ji}) = Q_{ji} + \Delta(LSB_{ji}, S_{ji}) \quad (4)$$

–Gather the stego blocks Q'_j that have pixels values included the secret ransomware data.

else if $HD_j \geq k/2$, **then**

–Set the marked bit $LSB_{(j \ k+1)}$ to 1.

–Determine the LSB-based steganography cost function using Eq. (3) to hide $(\bar{S}_{j1}, \bar{S}_{j2}, \dots, \bar{S}_{jk})$ into $(Q_{j1}, Q_{j2}, \dots, Q_{jk})$, where \bar{S}_{j1} is the inverse of S_{j1} . Therefore, the LSBs of Q_j will be $(\bar{S}_{j1}, \bar{S}_{j2}, \dots, \bar{S}_{jk}, 1)$.

–Calculate the value of the stego pixel $P(Q_{ji}, S_{ji})$ using Eq. (5).

(Continued)

$$P(Q_{ji}, S_{ji}) = Q_{ji} + \Delta(LSB_{ji}, \bar{S}_{ji}) \quad (5)$$

–Gather the stego blocks Q'_j that have pixels values included the secret ransomware data.

end if

end for

output: Stego HEVC frame.

The output of this algorithm is the stego frame, which is the frame that is injected with the ransomware application (apk). The stego frame's quality will be deeply assessed by examining 16 different metrics. If the stego frame passes the quality check, it will be combined with the rest of the frames to restore the complete video. But, this video is now infected with ransomware.

In contrast, the RExP starts by taking the stego HEVC video as an input to extract the stego frame and forward it to the ransomware extraction phase, as detailed in Algorithm (2). This algorithm's outputs are the ransomware application itself and the original frame. The original frame is then combined with the rest of the frames to restore the original clean HEVC video.

Algorithm (2): Steps of the extraction phase

input: Stego HEVC frame.

–Divide each input K -pixels stego HEVC frame into different blocks of Q'_j
 $(j = 1, 2, \dots, [K/(k+1)])$, where $k+1$ is the length of each block, $Q'_{j1}, Q'_{j2}, \dots, Q'_{j(k+1)}$ refers to the pixel value within each block.

–Get the binary bits of the $LSB_{j1}, LSB_{j2}, \dots, LSB_{jk}, LSB_{j(k+1)}$ of each pixel value within each block Q'_j .

–Check the value of $LSB_{j(k+1)}$ which is the marked bit of each block Q'_j .

for all collected marked bits, do

if $LSB_{j(k+1)} = 0$, **then**

–Gather the obtained bits of all LSB_j to get the secret ransomware data.

else if $LSB_{j(k+1)} \neq 0$, **then**

–Put the inverse value of the whole obtained bits of all LSB_{ji} to be $LSB_{ji} = \overline{LSB_{ji}}$, where $i = 1, 2, \dots, k$.

–Gather the resulted bits of all LSB_j to get the secret ransomware data.

end if

end for

output: Secret ransomware data.

The main idea of the utilized steganography approach was dividing the secret ransomware and the cover HEVC frame into different blocks. The cover HEVC frame used to embed the secret ransomware data is selected randomly. After that, the LSB-based hamming distance calculation is performed amongst the divided blocks of the secret data (ransomware) and the cover frames. Finally, the secret data bits are hidden into the marked bits of the cover HEVC frame-blocks based on the estimated hamming distance value. The major improvement of the introduced HEVC steganography approach was utilizing an adaptive steganography cost function that reduced the embedding influence of ransomware hiding within the stego HEVC frames by conserving the histogram features of the cover frames while introducing a desirable imperceptibility.

Furthermore, this approach accomplishes high capacity and superior hiding efficacy by ensuring the undetectability of the hidden ransomware data within the video frames. More quality evaluation metrics are examined to assess the performance of the REP and RExP processes. In case the assessment metrics of the stego frame did not achieve the desired and expected values, the REP process is repeated. This is to select a more suitable resolution of the cover video frame concerning the size of secret ransomware data to achieve higher perception quality and adequate capacity performance.

4 Model Evaluation and Result Discussions

This section presents the features of the ransomware samples and the standard cover HEVC streams used in this research experiments. Also, it lists the subjective and objective evaluation metrics that were applied to examine the performance of the proposed hiding approach, as shown in Fig. 2. The objective-based evaluation included 16 metrics to assess the quality of the resulted stego frame. Moreover, 65 Antivirus engines were used to scan the stego frame and the stego video to check if these engines can detect the ransomwares' existence. The subjective-based tests were also considered in this study by comparing (a) the original frame and the stego frame and (b) the original video and the stego video. Finally, the results of all metrics will be presented and analysed.

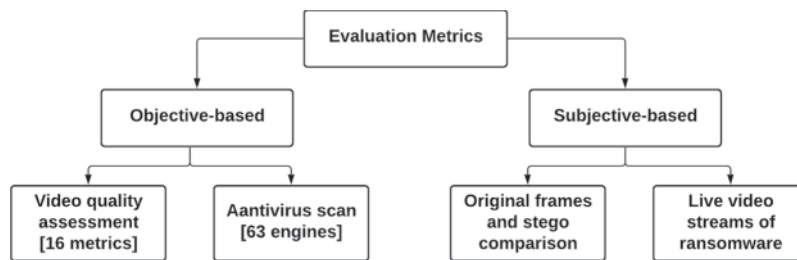


Figure 2: The applied evaluation metrics

4.1 Ransomware Samples and Standard HEVC Streams

To prove the efficiency of the proposed ransomware hiding approach, many experiments were conducted. In these experiments, we utilized different ransomware samples as secret messages and different HEVC streams¹ with various resolutions as cover media. The purpose was to check the capability of the proposed approach in hiding different ransomware sizes within different resolutions of cover video frames without affecting the main features and quality of the cover frames. Additionally, the proposed approach aims to achieve high secrecy of the ransomware by making it undetectable even by specialized antivirus engines. Tab. 2 presents the sizes of the tested ransomware samples, while Tab. 3 introduces the resolutions of the tested HEVC streams.

4.2 Quality Assessment

As part of the objective-based evaluation, 16 metrics are mathematically presented in this section. Throughout the following equations, $x(m, n)$ signifies the original cover video frame, $x'(m, n)$ represents the resulted stego video frame, and M and N are the numbers of the pixels in rows and columns, respectively.

¹ <http://trace.eas.asu.edu/yuv/>.

Table 2: Size of the tested ransomware samples

Test sample	Size
Ransomware1	60 KB
Ransomware2	171 KB
Ransomware3	390 KB
Ransomware4	734 KB
Ransomware5	1.036 MB

Table 3: Resolution of the tested video streams

HEVC stream	Resolution
Race	640 × 480
Johnny	1280 × 720
Jockey	1920 × 1088
PeopleOnStreet	2560 × 1600
Bospharous	3840 × 2160

• Mean Square Error (MSE)

MSE [30] is one of the quality assessment metrics that are used in image and video quality evaluation applications. It is used to estimate the error between the cover and stego video frames. A lower value of the MSE metric means that the video frame has a good quality, and there is a higher similarity between the cover and stego video frames. This metric is mathematically represented in Eq. (6):

$$MSE = \frac{1}{M \times N} \sum_{m=1}^M \sum_{n=1}^N (x(m, n) - x'(m, n))^2 \quad (6)$$

• Peak Signal to Noise Ratio (PSNR)

The PSNR metric [30] is a function of the MSE metric. So, it is preferable to get a large PSNR value to obtain a good quality for the resulted stego video frame. The PSNR is measured in decibels and it is represented in Eq. (7):

$$PSNR = 10 \log_{10} \left(\frac{255^2}{MSE} \right) \quad (7)$$

• Signal-to-Noise Ratio (SNR)

SNR [31] is described as the ratio between the two average powers of signal and noise. It is measured in decibels and it is presented in Eq. (8):

$$SNR = 10 \log_{10} \left(\frac{\sum_{m=1}^M \sum_{n=1}^N (x(m, n))^2}{\sum_{m=1}^M \sum_{n=1}^N (x(m, n) - x'(m, n))^2} \right) \quad (8)$$

It is superior to achieve a large SNR value to obtain a good quality for the resulted stego video frame.

• Weighted Signal-to-Noise Ratio (WSNR)

The WSNR metric is a weighted form of the SNR metric which is developed by Varkur and Mitsa [32] utilizing the sensitivity contrast function. It is the ratio between the average weighted powers of the signal and noise, respectively. It is measured in decibels and it is better to obtain a large WSNR value to attain a good quality for the resulted stego video frame.

• Noise Quality Measure (NQM)

NQM [33] is used to determine the distortion caused to a video frame due to both frequency shift and noise effect. Also, the NQM metric can be employed to estimate the effects of local luminance, contrast perception, texture masking, and contrast masking. Thus, it can be considered a weighted version of the SNR metric between the cover and stego video frames. Consequently, it is desirable to get a large value of NQM to obtain a good quality for the resulted stego video frame. The NQM is measured in decibels and it is represented in Eq. (9):

$$NQM = 10 \log_{10} \left(\frac{\sum_{m=1}^M \sum_{n=1}^N (x'(m,n))^2}{\sum_{m=1}^M \sum_{n=1}^N (x'(m,n) - x(m,n))^2} \right) \quad (9)$$

• Structural Content (SC)

The SC metric [34] is the ratio of the power of the original signal (cover video frame) to the power of the processed signal (stego video frame). So, it is preferable to obtain a small SC value to get good quality for the resulted stego video frame. It can be defined as in Eq. (10):

$$SC = \frac{\sum_{m=1}^M \sum_{n=1}^N x(m,n)^2}{\sum_{m=1}^M \sum_{n=1}^N x'(m,n)^2} \quad (10)$$

• Maximum Difference (MD)

The MD metric [35] determines the maximum amount of error in the processed signal compared to the original signal. It estimates the difference between the reference cover video frame and the processed stego video frame. Thus, it is preferable to obtain a small MD value to get a good quality of the resulted stego video frame. It is defined in Eq. (11):

$$MD = \max |x(m,n) - x'(m,n)| \quad (11)$$

• Normalized Absolute Error (NAE)

The NAE metric [30] is the ratio between the MD metric and the absolute value of the reference cover video frame. For achieving a good quality of the stego video frame, it is recommended to get a low value of the NAE metric. It is represented in Eq. (12):

$$NAE = \frac{\sum_{m=1}^M \sum_{n=1}^N |x(m,n) - x'(m,n)|}{\sum_{m=1}^M \sum_{n=1}^N |x(m,n)|} \quad (12)$$

• Laplacian Mean Square Error (LMSE)

LMSE evaluation metric [36] is based on estimating the measurement of video frame edges. It is better to achieve a small LMSE value to obtain a good quality for the resulted stego

video frame. The LMSE metric is mathematically represented in Eq. (13):

$$LMSE = \frac{\sum_{m=1}^M \sum_{n=1}^N [L(x(m, n)) - L(x'(m, n))]^2}{\sum_{m=1}^M \sum_{n=1}^N [L(x'(m, n))]^2} \quad (13)$$

where the Laplacian operator is symbolized by $L(x(m, n))$ for the signal $x(m, n)$, and it is given as:

$$L(x(m, n)) = x(m+1, n) + x(m-1, n) + x(m, n+1) + x(m, n-1) - 4x(m, n) \quad (14)$$

• Structural Similarity Index (SSIM)

The SSIM metric [37] is utilized to estimate the visual effect of the luminance shift, contrast changes, and structural alterations of a video frame. So, it is used to extract the structural information of the objects inside the input video frame. Thus, a degree of estimated structural similarity is a clear indication of the recognized video quality. The SSIM metric between the cover and stego frames of x and x' signals is described in Eq. (15):

$$SSIM(x, x') = [l(x, x')]^\alpha [c(x, x')]^\beta [s(x, x')]^\gamma \quad (15)$$

where $s(x, x')$, $c(x, x')$, and $l(x, x')$ refer to the structural, contrast, and luminance components of the video frame index, respectively. They are represented as:

$$s(x, x') = \frac{\sigma_{xx'} + c_3}{\sigma_x \sigma_{x'} + c_3} \quad (16)$$

$$c(x, x') = \frac{2\sigma_x \sigma_{x'} + c_2}{\sigma_x^2 + \sigma_{x'}^2 + c_2} \quad (17)$$

$$l(x, x') = \frac{2\mu_x \mu_{x'} + c_1}{\mu_x^2 + \mu_{x'}^2 + c_1} \quad (18)$$

where c_1 , c_2 , and c_3 are positive small constants, μ_x and $\mu_{x'}$ signify the means of the cover and stego video frames, respectively. σ_x and $\sigma_{x'}$ signify the standard deviations of the cover and stego video frames, respectively. $\sigma_{xx'}$ is the covariance between the cover and stego video frames. For an 8-bit grayscale video frame combined of $L = 2^8$ gray-levels, $c_1 = (k_1 L)^2$, $c_2 = (k_2 L)^2$, and $c_3 = 2$, where $k_2 = 0.03$ and $k_1 = 0.01$. It is noticed that in the case of $c_1 = c_2 = 0$, the SSIM metric is reduced to the Universal Quality Index (UQI) metric. The range of the SSIM metric is -1 to 1 . Therefore, obtaining a high value of SSIM indicates high similarity between the cover and stego video frames.

• Multi-Scale SSIM Index (MS-SSIM)

The MS-SSIM metric is considered as an improved version of the SSIM metric. It is devised to determine the visual quality of a video frame based on multiple scales [38]. So, it has different forms of scales. The lowest scale is utilized to measure the luminance component, whilst the structural and contrast components are determined based on the j scale, and it has also the highest scale represented as M . The range of the MS-SSIM metric is -1 to 1 . Therefore, obtaining a high value of MS-SSIM indicates high similarity between the cover and stego video frames.

• Feature Similarity Index (FSIM)

The FSIM metric [39] is utilized to extract the low-level features within a video frame such as gradient magnitude and phase congruency. The gradient magnitude composes the contrast

information, while the phase concurrency contains great information of the primary features. The range of the FSIM metric is -1 to 1 . So, achieving a high value of FSIM means a high similarity between the cover and stego video frames. The FSIM metric is described in Eq. (19):

$$FSIM = \frac{\sum_{x \in \Omega} PC_m(x) \cdot S_L(x)}{\sum_{x \in \Omega} PC_m(x)} \quad (19)$$

where the gradient magnitude information can be estimated using the Sobel operator $S_L(x)$, the spatial domain of video frames is provided by Ω , and the projected phase congruency information can be determined by $PC_m(x)$.

• Universal Quality Index (UQI)

UQI [40] is global instead of being local or specifically intended for the video frames being examined or on the particular observers. Therefore, in quality assessment evaluation for image and video applications, the UQI is recommended to be utilized for quality assessment where it composes the correlation, luminance, and contrast components as it is determined as in Eq. (20):

$$UQI = \text{Contrast component} \times \text{Luminance component} \times \text{Correlation component} \quad (20)$$

So, the UQI metric is defined in Eq. (21):

$$UQI = \frac{4\sigma_{xx'}\mu_x\mu_{x'}}{(\sigma_x^2 + \sigma_{x'}^2)(\mu_x^2 + \mu_{x'}^2)} \quad (21)$$

The range of the UQI metric is -1 to 1 , so, there is a higher similarity between the cover and stego video frames in the case of obtaining a higher UQI value.

• Normalized Cross Correlation (NK)

The NK metric [41] is used to compare the processed stego video frame and the reference cover video frame. It is expressed in Eq. (22):

$$NK = \frac{\sum_{m=1}^M \sum_{n=1}^N x(m,n) \cdot x'(m,n)}{\sum_{m=1}^M \sum_{n=1}^N x(m,n)^2} \quad (22)$$

For the success of the steganography process, it is preferable to get the highest value of 1 between the cover and stego frames to achieve higher performance efficiency.

• Average Difference (AD)

The AD metric [42] determines the average variation between the reference cover video frame and the processed stego video frame. So, it is desirable to get a smaller value of AD to obtain a good quality for the resulted stego video frame. It is calculated in Eq. (23):

$$AD = \frac{\sum_{m=1}^M \sum_{n=1}^N x(m,n) - x'(m,n)}{M \times N} \quad (23)$$

• Pixel-Based Visual Information Fidelity (VIFP)

The VIFP metric is an improved version of the Visual Information Fidelity (VIF) metric with a low computational cost. It is used to extract and compare the pixel-level information within the cover and stego video frames [43]. It is preferable to get the highest value of 1 between the cover and stego frames to accomplish the high performance of the employed steganography process.

• Entropy (E)

The entropy metric is utilized to estimate the amount of information in the cover and stego frames. It is preferable to get identical entropy values for the stego and cover frames. It is calculated in Eq. (24):

$$E = - \sum_{j=0}^{255} P(m_j) \times \log P(m_j) \quad (24)$$

where the j^{th} grey frame value is denoted by m_j and the probability of m_j in a video frame is given by $P(m_j)$.

4.3 Results Discussion
















This section presents and discusses the results of all examined evaluation metrics considered in this study.

4.3.1 Video Quality Assessment

To evaluate the employed HEVC steganography approach, we performed various experiments using the different HEVC streams and ransomware samples that were presented in Section 4.1.

Tab. 4 presents the subjective findings of the tested HEVC frames with distinct resolutions in case of hiding five ransomware samples with different sizes, while Tab. 5 introduces the histogram findings.

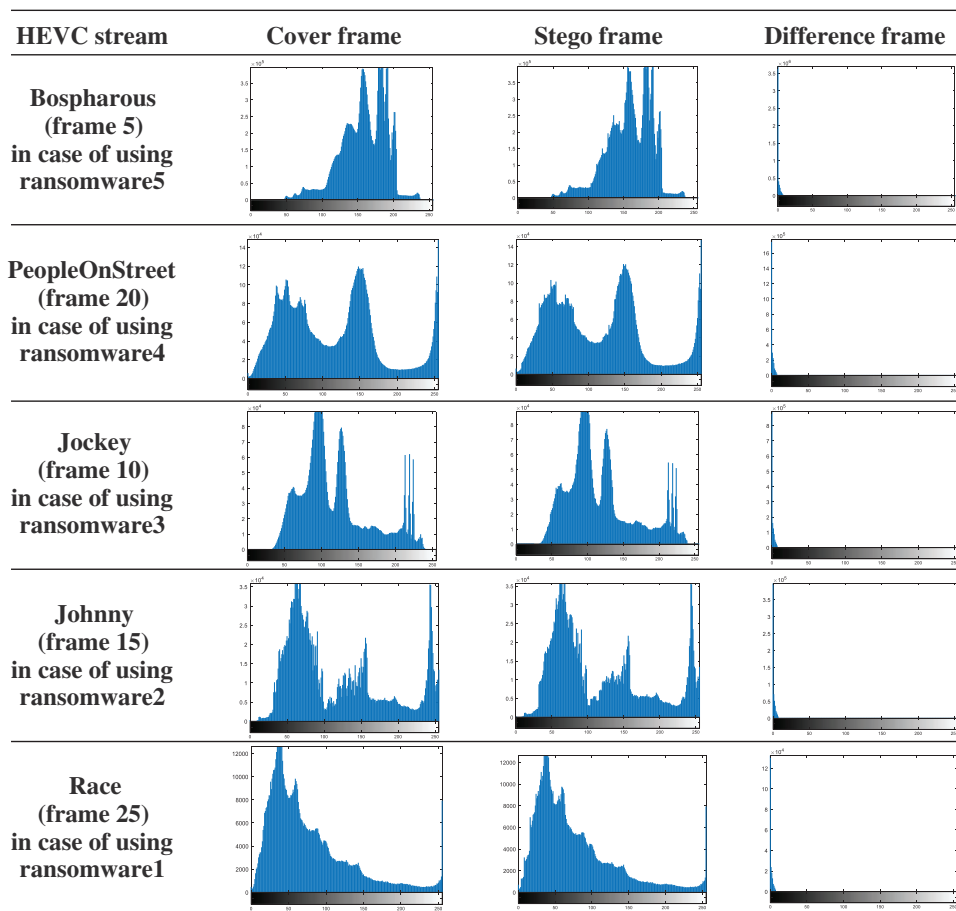
Table 4: Subjective outcomes of the tested HEVC frames in case of using different ransomware samples

HEVC stream	Cover frame	Stego frame	Difference frame
Bospharous (frame 5) in case of using ransomware5	 (Entropy=6.8609)	 (Entropy=6.8620)	 (Entropy=0.4172)
PeopleOnStreet (frame 20) in case of using ransomware4	 (Entropy=7.6189)	 (Entropy=7.6222)	 (Entropy=0.5657)
Jockey (frame 10) in case of using ransomware3	 (Entropy=7.2139)	 (Entropy=7.2290)	 (Entropy=0.6090)
Johnny (frame 15) in case of using ransomware2	 (Entropy=7.4922)	 (Entropy=7.4980)	 (Entropy=0.5893)
Race (frame 25) in case of using ransomware1	 (Entropy=7.3763)	 (Entropy=7.3783)	 (Entropy=0.6009)

It is observed from the introduced results in [Tab. 4](#) that the suggested steganography approach achieves high imperceptibility results, where the stego frames are visually similar to the cover frames with a minor difference in their entropy values. This can also be observed by the obtained difference frames between the cover and stego frames, where their entropies (the amount of information) have very low values near to zero. This is clearly shown by the completely black pixels in the resulted difference frames.

Furthermore, the acquired histogram results in [Tab. 5](#) further prove the imperceptibility efficacy amongst cover and stego frames by achieving approximately the same pixel intensity distributions with similar histograms. Moreover, it is also demonstrated that there is no pixel distribution of the obtained histograms of the difference frames except a low distribution around the zero-pixel value.

Table 5: Histogram outcomes of the tested HEVC frames in case of using different ransomware samples



[Tab. 6](#) provides the objective quality assessment results of the tested video streams after embedding the ransomware samples. The table shows the results of the 16 different evaluation metrics that assess the quality of the stego frames. The targeted optimal values to be achieved by each of these metrics are also listed in [Tab. 6](#). Therefore, the obtained results greatly declare

that the employed steganography approach achieves significant performance. This is revealed by attaining low values of MSE, SC, MD, LMSE, NAE, and AD metrics and accomplishing high values of PSNR, SSIM, UQI, FSIM, NQM, NK, SNR, VIFP, WSNR, and MS-SSIM in all tested video streams.

Table 6: Objective quality assessment results of tested video streams

Metric	Optimal Value	Bosphorus	PeopleOnStreet	Jockey	Johnny	Race
MSE	0	0.0763	0.1416	0.1082	0.4671	0.3195
PSNR (dB)	>20 dB	59.3076	56.6211	57.7873	51.4364	53.0854
SNR (dB)	>20 dB	36.6950	35.0550	37.6017	30.0256	31.9640
WSNR (dB)	>20 dB	59.9652	56.8041	61.6324	49.5267	51.4879
NQM (dB)	>20 dB	31.6079	44.4609	38.2935	38.1307	39.3277
SC	1	1.0002	1	0.9997	1	0.9999
MD	≤ 5	4	4	4	5	6
NAE	0	3.7599×10^{-04}	9.5144×10^{-04}	7.4547×10^{-04}	0.0021	0.0022
LMSE	0	0.0019	3.4359×10^{-04}	0.0025	0.0016	0.0036
SSIM	1	0.9996	0.9993	0.9994	0.9967	0.9973
MSSIM	1	0.9999	0.9999	0.9999	0.9978	0.9969
FSIM	1	0.9998	0.9998	0.9996	0.9979	0.9989
UQI	1	1	1	1	1	0.9979
NK	1	0.9996	1	1	1	1
AD	0	0.0177	0.0068	-0.0166	0.0036	-0.0061
VIFP	1	0.9993	0.9992	0.9994	0.9949	0.9982

4.3.2 Antivirus Scan

The antivirus scanning was performed using the VirusTotal² platform as part of the security test. VirusTotal conducts malware detection scanning utilizing over 65 antivirus scanning vendors such as Kaspersky, McAfee, Avast, Symantec, and many others.

In this experiment, the scanning has been implemented in three different stages. Initially, the original ransomware was scanned before hiding it inside the cover video frame. Following that, both the video frame with the embedded ransomware file (stego frame) and the combined video (stego video) were scanned to investigate the effectiveness of the applied steganography algorithm.

The results of VirusTotal scanning are demonstrated in Fig. 3. As it can be seen, a total of 35 out of 65 engines detected the ransomware file before applying the steganography algorithm (Fig. 3a). However, the ransomware was not detected by any engine after embedding it within the video frame (Fig. 3b). Furthermore, the antivirus scan of the combined HEVC stream, where the ransomware is concealed inside the video frame, shows that none of the VirusTotal engines was able to detect it (Fig. 3c).

² <https://www.virustotal.com/>.



Figure 3: The results of virustotal scanning before and after embedding the ransomware APK file (a) The scanning results of the original ransomware APK file (b) The scanning results of the video frame (png) after embedding the ransomware file (c) The scanning results of the whole video (avi) after embedding the ransomware file

In addition to the antivirus scan, we managed to upload and stream the stego videos through the research lab YouTube channel³, which is another proof of bypassing the existing security checks. This stresses the high efficiency of the proposed hiding approach.

5 Conclusion and Future Works

This paper has proposed an efficient, novel ransomware hiding approach using HEVC steganography. This work highlighted the shortcomings of the existing ransomware detection systems as they did not investigate the possibility of hiding the ransomware itself and finding ways to detect it, extract it, and then analyze it. Therefore, this work has utilized steganography and, in specific video steganography to hide ransomware with high efficiency in terms of (a) preserving the quality of the video and its characteristics after embedding the ransomware (b) protecting the privacy of the ransomware itself by making it difficult to be detected even by well-known antivirus engines. The proposed hiding approach was heavily examined using different subjective

³ <https://sel.psu.edu.sa/>.

and objective metrics and embedding different ransomware samples into video covers with various resolutions. The results revealed that the proposed approach succeeded in hiding ransomware and bypassing all quality and security tests. As future work, different steganography approaches can be experienced to hide new ransomware families or different malware apps in general. Also, an encryption stage can be added to encrypt the ransomware samples before embedding them within the cover video frames. Furthermore, different formats of multimedia files (e.g., image and audio) may be utilized as cover media. Moreover, advanced artificial intelligence tools and well-trained deep learning models can be utilized for testing the possibility of detecting the hidden ransomware apps within video frames.

Acknowledgement: The authors would like to acknowledge the support of Prince Sultan University for paying the Article Processing Charges (APC) of this publication.

Funding Statement: The authors received no specific funding for this study.

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

References

- [1] F. Alswaina and K. Elleithy, "Android malware family classification and analysis: Current status and future directions," *Electronics*, vol. 91, no. 6, pp. 1–20, 2020.
- [2] J. Silva, L. López, A. Valdivieso and M. Hernández, "A survey on situational awareness of ransomware attacks-detection and prevention parameters," *Remote Sensing*, vol. 11, no. 10, pp. 1–20, 2019.
- [3] D. Su, J. Liu, X. Wang and W. Wang, "Detecting android locker-ransomware on chinese social networks," *IEEE Access*, vol. 7, pp. 20381–20393, 2018.
- [4] S. Sharma, R. Kumar and C. Krishna, "RansomAnalysis: The evolution and investigation of android ransomware," in *Proc. of Int. Conf. on IoT Inclusive Life*, Singapore, NITTTR Chandigarh, India: Springer, pp. 33–41, 2020.
- [5] S. Alsoghyer and I. Almomani, "On the effectiveness of application permissions for android ransomware detection," in *Proc. 6th IEEE Conf. on Data Science and Machine Learning Applications*, Riyadh, Saudi Arabia, pp. 94–99, 2020.
- [6] S. Imtiaz, S. Rehman, A. Javed, Z. Jalil, X. Liu *et al.*, "DeepAMD: Detection and identification of android malware using high-efficient deep artificial neural network," *Future Generation Computer Systems*, vol. 115, pp. 844–856, 2020.
- [7] S. Alsoghyer and I. Almomani, "Ransomware detection system for android applications," *Electronics*, vol. 8, no. 8, pp. 1–36, 2019.
- [8] H. Faris, M. Habib, I. Almomani, M. Eshtay and I. Aljarah, "Optimizing extreme learning machines using chains of salps for efficient android ransomware detection," *Applied Sciences*, vol. 10, no. 11, pp. 1–25, 2020.
- [9] R. Qaddoura, I. Aljarah, H. Faris and I. Almomani, "A classification approach based on evolutionary clustering and its application for ransomware detection," in *Evolutionary Data Clustering: Algorithms and Applications*. Singapore: Algorithms for Intelligent Systems, Springer, 2021.
- [10] K. Cabaj, L. Caviglione, W. Mazurczyk, S. Wendzel, A. Woodward *et al.*, "The new threats of information hiding: The road ahead," *IT professional*, vol. 20, no. 3, pp. 31–39, 2018.
- [11] W. Mazurczyk and I. Caviglione, "Information hiding as a challenge for malware detection," *IEEE Security & Privacy*, vol. 13, no. 2, pp. 89–93, 2015.
- [12] G. Sullivan, J. Ohm, W. Han and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, 2012.

- [13] O. Faragallah, W. El-Shafai, A. Sallam, I. Elashry, E. EL-Rabaie *et al.*, “Cybersecurity framework of hybrid watermarking and selective encryption for secure HEVC communication,” *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–25, 2021.
- [14] N. Soliman, M. Khalil, A. Algarni, S. Ismail, R. Marzouk *et al.*, “Efficient HEVC steganography approach based on audio compression and encryption in QFFT domain for secure multimedia communication,” *Multimedia Tools and Applications*, vol. 80, no. 3, pp. 4789–4823, 2021.
- [15] A. Sahu and G. Swain, “Reversible image steganography using dual-layer LSB matching,” *Sensing and Imaging*, vol. 21, no. 1, pp. 1–21, 2020.
- [16] A. Hindi, M. Dwairi and Z. AlQadi, “A novel technique for data steganography,” *Engineering Technology & Applied Science Research*, vol. 9, no. 6, pp. 4942–4945, 2019.
- [17] X. Liao, Y. Yu, B. Li, Z. Li and Z. Qin, “A new payload partition strategy in color image steganography,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 3, pp. 685–696, 2019.
- [18] X. Liao, J. Yin, M. Chen and Z. Qin, “Adaptive payload distribution in multiple images steganography based on image texture features,” *IEEE Transactions on Dependable and Secure Computing*, pp. 1–14, 2020.
- [19] X. Weng, Y. Li, L. Chi and Y. Mu, “High-capacity convolutional video steganography with temporal residual modeling,” in *Proc. of IEEE Int. Conf. on Multimedia Retrieval*, Ottawa, Canada, pp. 87–95, 2019.
- [20] J. Wang, X. Jia, X. Kang and Y. Shi, “A cover selection HEVC video steganography based on intra prediction mode,” *IEEE Access*, vol. 7, pp. 119393–119402, 2019.
- [21] S. Liu and D. Xu, “A robust steganography method for HEVC based on secret sharing,” *Cognitive Systems Research*, vol. 59, pp. 207–220, 2020.
- [22] Z. Zhang, Z. Li, J. Liu, H. Yan and L. Yu, “Steganography algorithm based on modified EMD-coded PU partition modes for HEVC videos,” *EURASIP Journal on Image and Video Processing*, vol. 2021, no. 1, pp. 1–20, 2021.
- [23] D. Galiano, A. Del-Barrio, G. Botella and D. Cuesta, “Efficient embedding and retrieval of information for high-resolution videos coded with HEVC,” *Computers & Electrical Engineering*, vol. 81, no. 106, pp. 1–15, 2020.
- [24] V. Radunović and M. Veinović, “Malware command and control over social media: Towards the serverless infrastructure,” *Serbian Journal of Electrical Engineering*, vol. 17, no. 3, pp. 357–375, 2020.
- [25] P. Bąk, J. Bieniasz, M. Krzemiński and K. Szczypiorski, “Application of perfectly undetectable network steganography method for malware hidden communication,” in *Proc. of 4th IEEE Int. Conf. on Frontiers of Signal Processing*, Poitiers, France, pp. 34–38, 2018.
- [26] S. Fathi-Kazerooni and R. Rojas-Cessa, “GAN tunnel: Network traffic steganography by using GANs to counter internet traffic classifiers,” *IEEE Access*, vol. 8, pp. 125345–125359, 2020.
- [27] M. Kaushik, M. Malik and B. Narwal, “Developing malware and analyzing it afore & after steganography with OSINTs,” in *Proc. of IEEE Int. Conf. for Innovation in Technology*, Bangluru, India, pp. 1–4, 2020.
- [28] G. Stergiopoulos, D. Gritzalis, E. Vasilellis and A. Anagnostopoulou, “Dropping malware through sound injection: A comparative analysis on android operating systems,” *Computers & Security*, vol. 105, pp. 1–22, 2021.
- [29] J. Cheng, Z. Chen and R. Yang, “An efficient histogram-preserving steganography based on block,” *EURASIP Journal on Image and Video Processing*, vol. 2018, no. 1, pp. 1–13, 2018.
- [30] W. El-Shafai, S. El-Rabaie, M. El-Halawany and F. Abd El-Samie, “Enhancement of wireless 3D video communication using color-plus-depth error restoration algorithms and bayesian kalman filtering,” *Wireless Personal Communications*, vol. 97, no. 1, pp. 245–268, 2017.
- [31] W. El-Shafai, S. El-Rabaie, M. El-Halawany and F. Abd El-Samie, “Recursive bayesian filtering-based error concealment scheme for 3D video communication over severely lossy wireless channels,” *Circuits Systems, and Signal Processing*, vol. 37, no. 11, pp. 4810–4841, 2018.

- [32] T. Mitsa and K. Varkur, "Evaluation of contrast sensitivity functions for the formulation of quality measures incorporated in halftoning algorithms," in *Proc. of IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Minneapolis, MN, USA, pp. 301–304, 1993.
- [33] N. Damera-Venkata, T. Kite, W. Geisler, B. Evans and A. Bovik, "Image quality assessment based on a degradation model," *IEEE Transactions on Image Processing*, vol. 9, no. 4, pp. 636–650, 2000.
- [34] A. Mason, J. Rioux, S. Clarke, A. Costa, M. Schmidt *et al.*, "Comparison of objective image quality metrics to expert radiologists' scoring of diagnostic quality of MR images," *IEEE Transactions on Medical Imaging*, vol. 39, no. 4, pp. 1064–1072, 2020.
- [35] A. Lahoulou, A. Bouridane, E. Viennet and M. Haddadi, "Full-reference image quality metrics performance evaluation over image quality databases," *Arabian Journal for Science and Engineering*, vol. 38, no. 9, pp. 2327–2356, 2013.
- [36] W. El-Shafai, I. Almomani and A. Alkhayer, "Optical bit-plane-based 3D-JST cryptography algorithm with cascaded 2D-FRFT encryption for efficient and secure HEVC communication," *IEEE Access*, vol. 9, pp. 35004–35026, 2021.
- [37] W. El-Shafai, S. El-Rabaie, M. El-Halawany and F. Abd El-Samie, "Security of 3D-HEVC transmission based on fusion and watermarking techniques," *Multimedia Tools and Applications*, vol. 78, no. 19, pp. 27211–27244, 2019.
- [38] Z. Wang, E. Simoncelli and A. Bovik, "Multiscale structural similarity for image quality assessment," in *Proc. IEEE Thrity-Seventh Asilomar Conf. on Signals, Systems & Computers*, Pacific Grove, CA, USA, pp. 1398–1402, 2003.
- [39] L. Zhang, L. Zhang, X. Mou and D. Zhang, "FSIM: A feature similarity index for image quality assessment," *IEEE Transactions on Image Processing*, vol. 20, no. 8, pp. 2378–2386, 2011.
- [40] Z. Wang and A. Bovik, "A universal image quality index," *IEEE Signal Processing Letters*, vol. 9, no. 3, pp. 81–84, 2002.
- [41] W. El-Shafai, S. El-Rabaie, M. El-Halawany and F. Abd El-Samie, "Proposed adaptive joint error-resilience concealment algorithms for efficient colour-plus-depth 3D video transmission," *IET Image Processing*, vol. 12, no. 6, pp. 967–984, 2018.
- [42] C. Gokilavani, N. Rajeswaran, V. Karthick, R. Kumar and N. Thangadurai, "Comparative results performance analysis of various filters used to remove noises in retinal images," in *Proc. IEEE Online Int. Conf. on Green Engineering and Technologies*, Coimbatore, India, pp. 1–5, 2015.
- [43] H. Sheikh and A. Bovik, "Image information and visual quality," *IEEE Transactions on Image Processing*, vol. 15, no. 2, pp. 430–444, 2006.