IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 8, 2020, accepted October 9, 2020, date of publication October 12, 2020, date of current version October 26, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3030608

SLEPX: An Efficient Lightweight Cipher for Visual Protection of Scalable HEVC Extension

RIZWAN ALI SHAH[1], MAMOONA N. ASGHAR[1,2], SAIMA ABDULLAH[1], NADIA KANWAL[2,3], (Senior Member, IEEE), AND MARTIN FLEURY[4], (Member, IEEE)

[1]Department of Computer Science and IT, The Islamia University of Bahawalpur, Punjab 63100, Pakistan
[2]Software Research Institute, Athlone Institute of Technology, Athlone, N37 HD68 Ireland
[3]Lahore College for Women University, Lahore 54000, Pakistan
[4]School of EAST, University of Suffolk, Ipswich IP4 1QJ, U.K.

Corresponding author: Rizwan Ali Shah (riz.shah1985@gmail.com)

ABSTRACT This paper proposes a lightweight cipher scheme aimed at the scalable extension of the High Efficiency Video Coding (HEVC) codec, referred to as the Scalable HEVC (SHVC) standard. This stream cipher, **S**ymmetric Cipher for **L**ightweight **E**ncryption based on **P**ermutation and E**X**lusive OR (SLEPX), applies Selective Encryption (SE) over suitable coding syntax elements in the SHVC layers. This is achieved minimal computational complexity and delay. The algorithm also conserves most SHVC functionalities, i.e. preservation of bit-length, decoder format-compliance, and error resilience. For comparative analysis, results were taken and compared with other state-of-art ciphers i.e. Exclusive-OR (XOR) and the Advanced Encryption Standard (AES). The performance of SLEPX is also compared with existing video SE solutions to confirm the efficiency of the adopted scheme. The experimental results demonstrate that SLEPX is as secure as AES in terms of visual protection, while computationally efficient comparable with a basic XOR cipher. Visual quality assessment, security analysis and extensive cryptanalysis (based on numerical values of selected binstrings) also showed the effectiveness of SLEPX's visual protection scheme for SHVC compared to previously-employed cryptographic techniques.

INDEX TERMS Selective encryption, HEVC, SHVC, CABAC, CDN as a Service, middlebox, quality metrics.

I. INTRODUCTION

In terms of bandwidth utilization, the main content that is currently transferred over the Internet is visual multimedia [1], i.e. video and still images, with applications such as Video-on-Demand (VoD), video conferencing, video phone, and telemedicine. Multimedia content requires a strong compression method, one that sufficiently reduces the bitrate over the transmission medium because otherwise, even in compressed form, its bitrate can be significant. Video content is stored and forwarded in numerous formats, differing in terms of frame rate, video quality, spatial resolution, bit depth and codec when fulfilling the wide spectrum of user requirements, including mobile users. However, the process of encoding, adaptation and delivering video consumes substantial storage resources as well as bandwidth. One way to address this issue is to use a state-of-the-art, standardized video codec [2], as video codecs have tended to almost double their compression ratio every decade [3]. Standardized codecs, such as those of the Motion Picture Experts Group (MPEG) series [4], guarantee the exchange of compressed video content. However, commercial operators should ensure that licensing issues are resolved [5]. Further, a scalable standardized codec is a flexible way to code in advance for the variety of expected video configurations requested by client devices, allow a set of video configurations to be embedded in a single compressed bitstream, which are subsequently extracted at a server or at an intermediate point in the network path.

The Scalable High Efficiency Video Coding extension (SHVC) [6], has been designed as Annex-H of the High Efficiency Video Coding (HEVC) codec standard [7], to encode the same video as a single bitstream, in a number of different configurations, such as a varying number of quality layers, resolutions, and frame rates. Compared to the prior Scalable Video Coding (SVC) standard [8] based on the

The associate editor coordinating the review of this manuscript and approving it for publication was Nilanjan Dey.

H.264/Advanced Video Coding (AVC) codec standard [9], SHVC provides two key advantages. Firstly, SHVC has a coding architecture simply built upon the HEVC standard and inter-layer prediction, consequently needing to make only relatively minor, high-level alterations. Secondly, the official release date of SHVC, being only one and half years later than HEVC's official release, in 2013, made the scalable extension relatively easier to deploy, without the need to account for many subsequent upgrades to HEVC. As SHVC is based on HEVC so it provides greater compression efficiency compared to the previous SVC standard but further computational complexity [10] has to be coped with in terms of additional encoding latency, as indeed happens with HEVC compared to H.264/AVC. The principal cause of that computational complexity are the number of coding choices made during temporal or inter prediction, which takes up to 70% of encoding time. Consequently, efforts started shortly after the introduction of HEVC [11] to streamline the prediction mode process, with the potential to reduce encoding complexity and, hence, encoding latency by between 30 to 50%. An entirely different method of encoding, other than by existing hybrid codecs, is through fractal coding [12]. Unfortunately, research interest has generally been limited due to the patents that exist for fractal coding, which has led to its rejection for commercial purposes.

Confidential, scalable bit-stream distribution is presently a desirable feature for multimedia applications over digital devices, including for real-time applications such as cloud-based video conferencing. Whatever the application, protection of the content is essential to make marketing feasible for a video streaming service, as otherwise there would be no financial encouragement to create videos and share them. Likewise users, especially businesses, will be concerned if there were to be a threat to the confidentiality of their VoD streams or indeed confidential, real-time streams such as those arising from telemedicine or video surveillance. New applications of video streaming continue to appear. For example, substantial streams of video are being newly accessed by users in countries such as Namibia [13] in south-western Africa, where wide-area wireless LANs are preferred in the absence of fixed networks. However, even if the configuration of the wireless packet sizes can be intelligently managed [13] such schemes lack protection of content, for which the proposal herein could be a solution. Added to that distributed management of such streaming and its protection might become advantageous, allowing lightweight encryption protection of the streamed video, with the degree of protection according to local device capability arranged intelligently [14]. Thus, encryption stands in the way of unlawful or illicit access to a video server's content and also shields the video stream during transmission over the Internet. For instance, the HTTP Live Streaming (HLS) specification [15] suggests deploying full encryption upon video segments using the Advanced Encryption Standard (AES) [16] with a 128-bit key, as well as Encrypted Media Extensions (EME)-based specification for key management of HTML5 video [17]. The Real-Time Messaging Protocol (RTMP) also provides full encryption based on the symmetric Rivest Cipher 4 (RC4) stream cipher (now considered cryptographically insecure [18]) or better, enclosed inside a Transport Layer Security (TLS) session, which uses asymmetric encryption for initial key exchange. However, full encryption, with HLS or RTMP (for example) tackles video data as being essentially textural and does not exploit the basic features of encoded video streams. Subsequently, the practice of full encryption increases the performance overhead [19] in terms of both bitrate overhead and possibly computational overhead. It also creates additional security weaknesses and processing latency if intermediate video adaptation should prove necessary, as discussed hereafter.

As a result of the short-comings of full encryption, to encrypt a video stream, Selective Encryption (SE) is preferable, as the amount of data encrypted with SE is reduced [20], which proportionally affects the processing latency at the codec and subsequently, if adaptation is involved. In addition, SE of video streams can be made decoder format compatible. In other words, because the format of an encrypted (and compressed) bitstream matches the specification of a standardized codec such as HEVC and its extensions, it can be decoded at intermediate devices in a network or while at rest in a cloud, without the need for the decryption that occurs with full encryption. Typical video adaptation includes splicing (or multiplexing) of other video clip bitstreams within the main bitstream, as would be needed for targeted advertising, no-reference video quality monitoring [21], and watermarking, as might be needed for the insertion of logos or to allow covert tracing of a video's source or destination [22]. Watermarking can serve as a form of retrospective content protection, usually working in conjunction with encryption. Some watermarking schemes can also be applied in the compressed domain, reducing the latency of the adaptation. A video splice should anyway be separately decodable but a problem arises if watermarks are inserted after the initial compression because the lack of synchronization between encoder and decoder will result in error drift over time at the decoder. In fact, watermarks should ideally be inserted in the main encoding loop because inserting before compression means that encoding constitutes an attack on the watermark. Insertion in the main encoding loop should ideally be after quantization because again quantization could otherwise harm the watermark. Watermarks, which are mostly applied covertly, have the advantage that removing a watermark usually involves severe distortion of watermarked video frames. However, there exist disadvantages as well. In a real-time environment the cost of decoding with a watermark in place increases, which may lead to a need for a simple watermark to be designed, which in turn opens that watermark to tampering. For the same reason, if there is a need within an application to remove one watermark and insert another, this could also lead to a simple watermark that could be tampered with. Even if a watermark needs to be designed to resist the many types of attack that have been devised against

them [23], there is no accepted standard for their use and legal doubts exist as to their enforceability. Returning to SE, if the scheme used to perform SE is not format compatible, such as in [24], applied to encrypt intra-coded I-frames, then the problem of full-encryption remains, even though the latency of decryption is reduced by virtue of the reduced amount of data encrypted within the bitstream. Instead, the current paper proposes a new lightweight encryption scheme, which is decoder format compatible and is applicable as SE for SHVC encoders. Notice that in standardized codecs, standardization is enforced by specification of the compressed bitstream sent to the decoder. Thus, a bitstream should be format compatible to be decodable.

Creating an SHVC stream is intended to avoid the need for transcoding, as the relevant segment from the composite bitstream, see Section II, can be extracted according to the configuration of the client device. For example, if the target device, acting as a client to a video server were to signal that in some way it has a screen with a limited spatial resolution, or a decoding rate that restricts the display frame rate, or a reduced bandwidth that does not allow high quality video to be sent to it, for example over a wireless network, then the matching segments can be extracted. In fact, SHVC, unlike SVC, also allows [25] selection of the color gamut, bit depth and even the base-layer codec. However, some devices, especially mobile ones may not have a hardware codec chipset that matches that of SHVC or even a software SHVC. In which case, codec format transcoding, e.g. [26], could become necessary at the MANE. For some video adaptation applications, such as watermark insertion or codec format conversion, it may be necessary to ensure that the bits affected by the application are not part of the SE scheme, though this is beyond the scope of the current paper.

It should be mentioned that commercial solutions for the lucrative infotainment sector of the market have tended to select an alternative method of multi-format streaming, typically based on an advanced form of HTTP Adaptive Streaming (HAS) [27], for example HLS [15], in which a device (client side) can dynamically opt for the most suitable format of video from diverse representations of the same video according to the best available device configurations. There are trade-offs in terms of storage (higher for HAS-type solutions) compared to bandwidth utilization (higher in the case of scalable solutions). There are also comparisons to be made when a real-time streaming service is the target, which is now topical in the current circumstances, for video conferencing using a cloud-based multipoint server. As an example, [28] contains a proposal for a scalable, video-conferencing scheme for real-time applications such as telehealth, even before the current marketplace for video conferencing took off.

A common misconception is that only selectively encrypting the Base-Layer (BL) output by a scalable codec is sufficient for content privacy. This notion is based upon the utilization of the base reference layer within the Enhancement Layers (ELs). Unfortunately, enhancement layers also contain intra-coded data parts, so that video data remains visible [29]. Likewise, encryption applied only on enhancement layer does not yield acceptable outcomes. Therefore, the proposed cipher, called **S**ymmetric Cipher for **L**ightweight **E**ncryption based on **P**ermutation and E**X**clusive OR SLEPX, acts upon both the base and enhancement layers' data and can do this by means of diverse selections for encryption.

The SE based ciphering scheme proposed in this research paper has the following contributions compared to the traditional ones:

- SLEPX is based on a SE method that deals with uniformly-distributed syntax elements in the encoded video stream. Due to this approach, the encrypted video stream preserves the statistical properties of the bitstream and its length.
- The decryption scheme is always accomplished at the target device end, without involving any intermediate device (such as within a Content Delivery Network (CDN) — see Section II.C), where otherwise the decryption keys and/or content may be visible. This stipulation implies that video is not decrypted prior to video adaptation.
- SE is performed at the last stage of SHVC encoder pipeline (the entropy-coding stage) as a joint cryptographic-compression scheme. As a result, there is limited computational overhead and, importantly, the encoding statistics established at earlier stages (such as changes to the correlation during frequency transforms) remain intact.
- The choice of which syntax elements to encrypt is determined both by the need to preserve the statistical properties of the bitstream and allow seamless intermediate video adaptation but also by the need to application across all the layers of the scalable video, as developed in Section IV-D.
- Choosing SE in place of full encryption opens up a threat that the content of a video may not be appropriately distorted and, as a result, the video content may be revealed without using the legitimate decryption process. Thus, additional video quality metrics are applied in this paper's work to check that the distortion of the video content is substantial so that only when absolute secrecy is needed, such as for military, legal, or some medical applications, is full encryption preferable. In fact, in an implementation, no-reference video quality metrics could be applied at a Media Aware Network Element (MANE) (see Section II.C) without the need for SLEPX decryption.
- Therefore, a vital contribution of this paper is the employment of the proposed SE algorithm with both HEVC [30] and SHVC [2] encoding. In fact, to find the effectiveness of the scheme, different experiments are reported that estimate: file sizes, the computation costs involved, and the effect, as part of the SE procedure, of choosing diverse syntax elements of those codecs'

Context Adaptive Binary Arithmetic Coding (CABAC) entropy coder [31].

- Frequently, the standardized AES cipher [16] is utilized for full encryption and also for the selected data encrypted as part of SE, prior to placing them back within their original positions within the compressed data. However, it is worth noting that in terms of images, AES may have poor resistance to cropping and noise attacks. Regardless of its security strength [32], AES involves multiple encryption rounds, which consume considerable computation. SLEPX is a cryptographically-scalable cipher as it can choose nominated syntax elements during entropy coding and it can reduce the computational load when subsequently encrypting those syntax elements. Thus, it remains in comparison to AES, a lightweight cipher, capable of superseding other ciphers for streamed video, especially when real-time communication is involved.

The remainder of this paper is arranged as follows. Section II introduces the background concepts of the SHVC encoder and the SE-based schemes utilized in this paper, as well as CDN-based applications. Section III follows with a review of related and recent research in the field. Section IV formulates the design of the proposed SLEPX cipher for the SHVC encoder. To determine the effectiveness of SLEPX, a variety of experiments have been conducted, which, in Section V, answer questions about the pictorial distortions achieved and computational overheads. In that Section also, the advantages of SLEPX over conventional ciphers are analyzed, finishing by discussing the effectiveness of a SLEPX-based security scheme for SHVC video bitstreams. Finally, Section VI makes some concluding remarks concerning this research, highlighting its significance.

II. BACKGROUND CONCEPTS

This Section provides overviews of the major modules of the scheme rather than full explanations of these modules. Above all, this Section outlines the SHVC and SE methods applied in this paper, as well as describing how SHVC would operate in a CDN.

A. SHVC

The ITU-T Video Coding Experts Group (VCEG) and ISO/IEC Moving Picture Experts Group (MPEG) requested proposals for the new scalable extension of the HEVC standard in July 2012 [2]. As a result, in October 2012, 20 different responses were received from research institutes, companies and academic institutes. As with HEVC, the scalable extension of HEVC was also developed under the supervision of the JCT-VC committee. Earlier video coding standards which were upgraded with scalable versions include H.262/MPEG-2 [33], H.263 standard [34], MPEG-4 Visual standard [35], and H.264/AVC [36]. The H.264/AVC extension to deal with scalable video is called as Scalable Video Coding (SVC), which is the most recent scalable video coding standard preceding SHVC.

Scalable encoding delivers multiple layers of the same coded video, each and every layer having distinct quality attributes for the same video scene. One of the layers is named as the BL having the lowest quality by some criterion. Though video quality (strictly Signal-to-Noise (SNR) ratio — Peak SNR (PSNR) for video) is the main scaling criterion (and there are other video quality metrics, including those based on the Human Visual System (HVS)), scaling can be by temporal (i.e. frame-rate) and/or spatial resolution (pixels per video frame). Other layers are referred to as ELs, providing enhanced video quality (and/or temporal and /or spatial resolution) when combined with the BL and are coded by referencing lower layers in the coding hierarchy stemming from the BL. Scalable video codecs only decode the subdivision (fragment) of a layer belonging to a scalable video bitstream that provides the lowest still-acceptable video quality compared to the fully decoded video bitstream. Consequently, they provide graceful degradation of the video compared with conventional non-scalable video bitstreams, when severely degraded video may result if there is an attempt to reduce the bitrate, once the video has already been encoded.

Alternatives to scalability are either transcoding or simulcast. In transcoding, the whole bitstream is often decoded and re-encoded with different encoding parameters, though it is possible to transcode through a partial decode by operating in the compression domain [26]. Video transcoding represents additional delay in the network path, (see Section II.C), which is why alternatives to full decoding are sometimes sought, especially if the cost of a hardware bank of transcoders is undesirable. Simulcast is based on distinct encodings of the same video sequence, clip, or film, at dissimilar bit rates owing to different video quality (usually achieved through different Quality Parameters (QPs) [4]) and possibly at diverse frame rates or spatial resolutions. At the receiving side, decoders may opt for specific simulcast bitstreams to be transmitted, as a response to network congestion causing display latency and or device capability. Though there has been progress in getting simulcast codecs to encode at real-time, there remains the issue of storage of multiple layers, which can be a problem for a server on a mobile device. A scalable video scheme provides benefits in multicast protocols, i.e. comprising streaming, broadcast, and video conferencing where multiple receivers request diverse versions of the same video. Scalable coding at real-time rates is common, as it has already been used for n-to-n video conferencing [28], where n is the number of conference participants. However, it can also be utilized for 1 to n video conferencing, in which n streams are collated at a multipoint server, possibly sited in a cloud data center, before being transmitted as a composite stream, with different versions transmitted, depending on device capabilities at the receiver ends. Capabilities include spatial resolution or supported frame rate of the display, computation rate and available energy for decoding.

As previously mentioned, there are traditionally three major categories of video scalability – Temporal, Spatial,

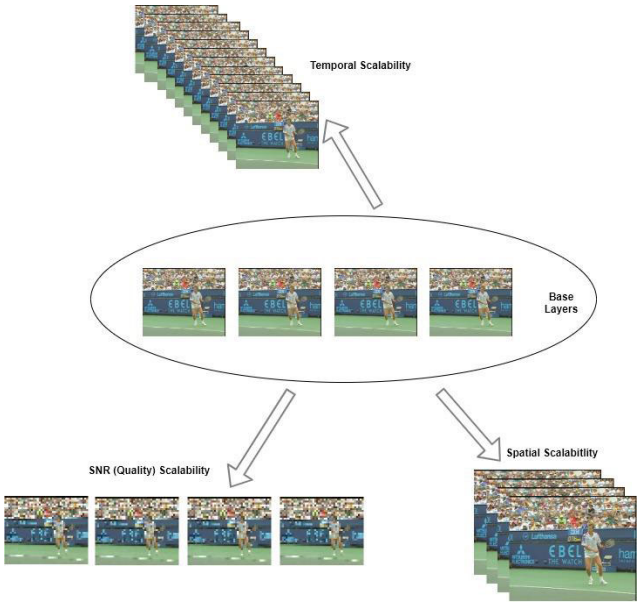


FIGURE 1. Different versions of scalability i.e. temporal, spatial and SNR scalability.

and Quality, see Fig. 1. To support the Ultra High Definition (UHD) video format in SHVC, three new kinds of scalability were additionally introduced, i.e. color gamut, bit depth, and hybrid codec scalability — for example combining a non-HEVC BL, e.g. H.264/AVC BL, with HEVC ELs [25] (hybrid scalability does not extend beyond the BL in SHVC). However, for simplicity the following discussion does not extend beyond the traditional forms of scalability.

SHVC works only with top-level syntax elements, generally at the Video Parameter Set (VPS) header level [7]. Syntax elements deliver statistics about the video layers such as the total number of layers, and for individual layers: their resolution, bit depth and the inter-layer dependencies. The SHVC architecture contains a number of HEVC encoders and is considered as a single encoder which encodes individual layers by utilizing N number of layers: in which there is one BL and $N-1$ ELs. In the context of the spatial scalability of SHVC, the HEVC encoder of the BL encodes an original video with an inferior resolution and forwards it to the first EL codec, which decodes the frame, including its Motion Vector (MV) information. The EL encoders (numbered L = 2, . . ., N-1) encode videos of higher resolution by means of the previous encoded frames from a lower layer, which supply a reference frame. The inter-layer reference frame is up-sampled, as are its MVs. Fig. 2 depicts the spatial scalability architecture when an SHVC encoder encodes two layers. In the case of quality scalability, each layer has the same resolution, but inter-layer processing between the reference frame and the EL encoding takes place rather than up-sampling. Subsequently the QP of frames are successively reduced in the ELs. Similar arrangements take place for temporal scalability, except that the reference frames passed to an EL are chosen according to the desired (lower) frame rate of that EL. Clearly in SHVC, it is possible to combine the three traditional forms of scalability within the same composite bitstream.

From the encryption point of view, it is important to notice that in an SHVC encoder, in each layer the final stage of the hybrid encoder is an entropy coder [31], which remains untouched, being the same as that of HEVC that is the CABAC coder mentioned in Section I. Thus, each layer in the SHVC encoder contains an autonomous CABAC engine acting as and entropy coder (see Fig. 2). Each CABAC engine performs three major functions: binarization, context modeling and arithmetic coding [4]. Initially, syntax elements of the video bitstream, if not in suitable binary form already, are converted into binary symbols (bins) in the binarization step to form binstrings. Notice that there are multiple alternative ways to perform binarization according to the type of symbol. There are additional binary code-trees available in

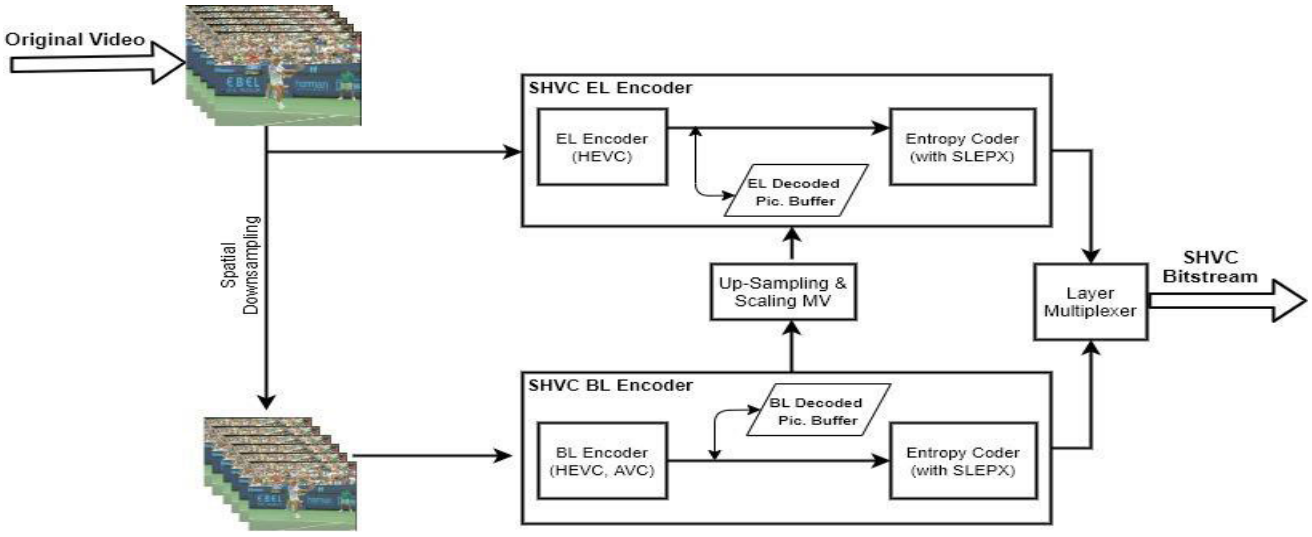


FIGURE 2. Block diagram of the SHVC encoder encoding two spatially scalable layers.

HEVC compared to H.264/AVC, basic examples being unary code, truncated unary code, truncated Rice code, kth order Exp-Golomb codes, and fixed-length codes. Notice that some basic codes are combined in HEVC, as is the case for the REG0, which combines kth order Exp-Golomb codes with fixed-length codes. Secondly, the probabilities of some of the binstrings are updated in the context modeling step, unless fixed or predetermined contexts are used. The binstrings that do not have their probabilities updated are important for format compliant SE and as such are further considered in Section IV-A. Finally, whether an adaptive or fixed context has been used binary arithmetic coding [4] is employed to compress the binstrings into bits. As presented in Fig. 2, multiplexing is performed on the outputs of the two encoders (one for each layer) to make a single composite bitstream that conforms to the SHVC standard. As is usual, it is the bitstream that is standardized rather than the encoder and every compliant SHVC decoder must be able to decode such a bitstream.

B. SELECTIVE ENCRYPTION

Selective Encryption (SE) is an efficient way [26] to render video contents confidential by making them sufficiently distorted. To achieve that, in the SLEPX scalable encryption scheme, according to a range of criteria, the most effective syntax elements are selected from the encoded video frames and encryption is applied only to those elements. Fig. 3 is an illustrative example of that distortion, which in this case renders the sports broadcast unwatchable. Due to the reduction in the amount of encrypted material, SE decreases computational costs compared to conventional full encryption [24]. In fact, full encryption of a standardized bitstream also removes its format compliance, making it no longer necessarily decodable by a decoder, which implies that a decoder could crash if decryption did not take place before decoding. SE is often utilized in real-time applications of video streaming such as video conferencing, video phone, and telemedicine. However, not all SE schemes offer sufficient data protection. Some SE algorithms have flaws in terms of: lack of decoder format compliance, additional bit-rate overhead, and unsatisfactory confidentiality. However, generally, the said issues can be overcome by deploying SE at the last stage of an encoder, i.e. the entropy-coding phase, by choice of suitable syntax elements to encrypt, as well as determining that the statistical distributions of the selected syntax elements will not be changed [37]. Only then does SE become beneficial, as there is no extra bit-rate overhead.

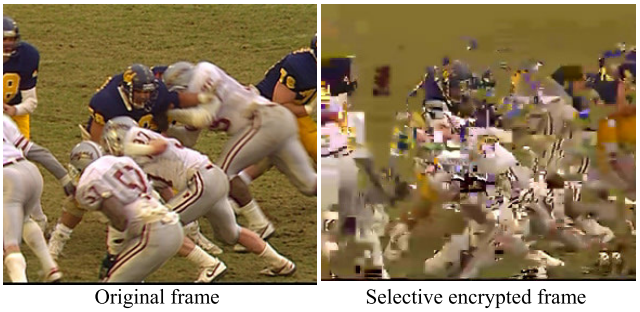


FIGURE 3. Illustrative example video frame with SE to make the frame unwatchable.

The SE algorithm employed in this paper operates with the CABAC entropy coder, which is the only entropy coder specified for HEVC [7]. It has an extended range of options compared to the H.264/AVC CABAC coder. The H.264/AVC family of codecs also had the option of Context-Adaptive Variable-Length (CAVLC) coding [38], which traded faster operation for increased bitrate (around 15% according to [39]). Either coder is inherently lossless. CABAC contains a number of parameters which could be employed during the encryption operation, i.e. Motion Vector Differences (MVDs); Coded Block Flag; the Transform Coefficients (TCs); the Macroblock (MB) types; the numerical signs of the MVDs and TCs; and the delta QPs (dQPs). However, not all the aforementioned syntax elements deliver decoder compliance and are also appropriate for scalable encoding due to this we only choose: the MVD signs, together with the Non-Zero-TC (NZ_TC) signs and their absolute value suffixes. The purpose of that selection is to ensure format compliance and constant bit-rate encryption. The latter is achieved because it is likely that the population of each of the signs will assume a Uniform distribution, before and after encryption, for sufficiently long video stream. Notice that encryption of MVD suffixes is omitted because, though their encryption would have a significant impact upon distortion, it would also break format compliance [28]. Finally, notice that the selection of syntax elements in this paper is dependent on the need to work across the three traditional scalabilities (see Section II-A) as embodied in SHVC. The issue of why the aforementioned syntax elements are suitable for that purpose is returned to in Section IV-D.

C. SHVC OVER A CDN

One possibility is to distribute an SHVC stream over a CDN using RTMP. The CDN might be based on a cloud and provide the CDN as a Service (CDNaaS) [40]. As remote devices are involved, so further issues arise if the video stream is not confidential and secured in general, for example if key management is not robust [41], though the latter is not the main subject of the current paper. Unfortunately, a video content shield through encryption may result in further latency, especially if the aforementioned intermediate decryption (as seen in Fig. 4), is needed to allow middlebox processing [21] before the bitstream is re-encrypted. In Fig. 4's middlebox, specifically a MANE, part of a CDNaaS, the traditional arrangement still requires time consuming decryption prior to adaptation of the decoded video bitstream, especially if full encryption has been applied and even if compressed domain processing is possible. Notice, however, that compressed domain processing requires a limited amount of decoding, for example, entropy decoding and re-coding. If it is known that video splicing is intended at the MANE, it is important to avoid encryption of the Network Adaptation Layer (NAL)

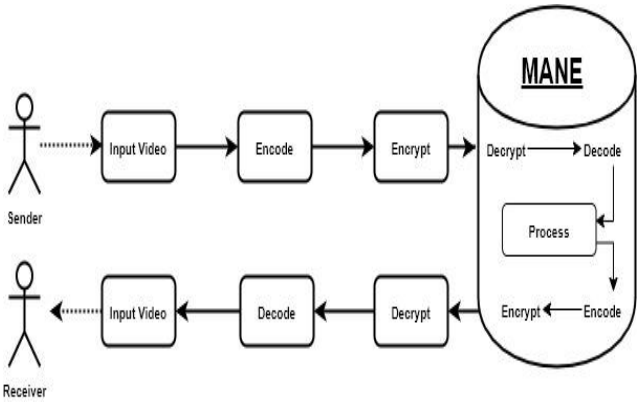


FIGURE 4. Traditional set-up, requiring decryption with key distribution, in a MANE as part of a CDNaaS.

that indicate the insertion point, which is usually possible with SE but not full encryption, though some SE schemes only operate on NALs [42].

Whatever the form of encryption, if intermediate decryption is required then the decryption key must be sent to each and every MANE, where it is exposed to attack. The same applies to content stored at rest in the cloud, where the key might be exposed to third-party companies, other than the cloud data center owners. Thus, full encryption presents a major potential security weakness due to the increased need for key distribution, given that the types of future processing of encrypted video may not be knowable in advance, i.e. whether video adaptation may eventually prove to be necessary.

III. RELATED WORK

There are various SE methods offered in the prior research into HEVC (sometimes known as H.265). The application of SE to H.264/AVC has already been examined in [24], [37]. This Section also presents emerging lightweight encryption procedures, which have also found a niche in the Internet-of-Things (IoT) and wireless sensor networks in general because of their need for reduced consumption of resources In fact, SE can be applied at different stages of a hybrid encoder, including within the header fields [42] but also during the transform stage, during quantization, within intra-frames only or within inter frames, and during entropy coding, which is favored in this paper for the reasons outlined in Section II.

Considering pioneering SE techniques for HEVC CABAC components, in [43] a novel choice of coding elements was made, i.e. mvd_sign_flag, coeff_sign_flag, the suffix of abs_mvd_minus2, and cu_qp_delta_abs. Once the elements had been extracted from the coded stream they were encrypted using the AES cipher before replacing them in their original positions in the bitstream. Cryptanalysis demonstrated robustness to replacement attacks, amongst other forms of attack. Computational overhead was also very low as compare to other suggested schemes. Thus, the scheme was beneficial for real-time application and also for delivery to systems having limited resources.

A real-time encryption scheme for H.264 and H.265 (HEVC) codecs appeared in [44]. The authors of [44] inserted a symmetric encryption/decryption transcoder at the output of a codec's encoder and before the input to the codec's decoder. The advantage of this architecture is that the ciphering transcoder can work independently of the codec, that it is not integrated into the encoder and decoder software for joint encryption/compression. It works either with CAVLC or CABAC entropy coding. A more significant change is probably that it does not only use bypass syntax elements in the entropy coder but also uses regular elements. The former elements are thought to have a Uniform distribution and, hence, do not alter in the long term the probability of a symbol occurring. Consequently, bypass mode symbols do not update the probability tables maintained by an encoder. On the other hand, regular mode elements do update the internal probability tables. The use of the encryption/decryption transcoder allows regular mode elements to be utilized in encryption because the tables have already been updated before encryption takes place. Equally, as decryption takes place before these elements reenter the decoder, they do not affect the decoder's tables, which, hence, replicate the probability table's values at the encoder. The authors of [44] do not specify which syntax elements they choose, which means that their implementation is proprietary and their method appears to rely on obfuscation, which is often an incentive to those attempting to break the cipher, for example through a replacement attack. The other main shortcoming is that if non-bypass mode elements are encrypted then the bitrate of the compressed bitstream could increase after encryption. This is because the non-Uniform distributed elements may have a lower bitrate than those of a Uniform distributed element. On the other hand, encryption usually imposes a Uniform distribution on those encrypted elements.

However, the authors of [44] then returned to these issues in a journal version of [44], namely in [45]. Firstly, non-bypass elements are specified for use in the scheme, namely the intra-coding luma prediction modes are selected, which improves upon the treatment of Intra-coding in other SE schemes. Secondly, it is found that the average change to the bitrate by ciphering these regular mode elements is an increase of only 0.5%. However, in [45] for the HEVC-CABAC, encryption of the intra-coding luma prediction mode has an impact on decoder compatibility, which risks decoder failure. In [45], the details of how this risk can be overcome, with limited additional complexity in the symmetric transcoder through the use of two sets of probability tables, according to the size of the Intra coding unit. Only the last coefficient is coded to make sure that the same number of coefficients is coded in the original and encrypted versions. Both [44] and [45] retain the use of AES encryption of the selected bits, while, if encryption of the structural elements in [44] and [45] (the regular mode elements, specifically

the intra-coding mode) was retained, lighter-weight SLEPX could be substituted.

If it is thought preferable to limit encryption to by-pass elements of the CABAC entropy coder then [46] is particularly useful, as it reports results on the impacts of encrypting those elements, either individually or in combination with other elements. The paper confirms that encryption of syntax elements take from the CABAC by-pass engine has 0% statistical impact on the Bjøntegaard delta (BD) rate of the output bitstream. (The BD rate is a measure of the average rate-distortion increase in bits/s over a specified quality range, measured in PSNR, as a result of an encoding change.) It also ranks a set of elements in terms of the change in PSNR when using them for SE, with a lower PSNR indicating greater distortion in terms of that quality metric. The ranking is also reported for Structural Similarity (SSIM) index change ranking [47], with SSIM being more closely correlated to the HVS response than PSNR for the same tests across many video sequences, with a range of QPs. The many other observations of [46] cannot be reported in this short overview. For example, Sample Adaptive Offset (SAO) signaling proceeds through the bypass filter but has the least distortion impact according to the paper's rankings and, moreover, is frequently turned off anyway. The authors of [46] conclude that for a given multimedia production chain, their work can be used to manage which elements are not involved in any middlebox adaptation, possibly allowing compression domain changes at the middlebox without the need to decode. That is a very different approach to SE compared to [45], which abstracts away the SE process, even separating it from the codec itself.

Region-of-Interest (ROI) encryption is of particular interest to video surveillance applications, when the privacy of those surveilled is required. In [48], either the ROIs can be tracked, for example with face recognition with video frames, or HEVC's rectangular tiles can form ROIs. Two SE methods are considered, the one format compliant and the other not so. However, whichever method is used if Motion Vectors (MVs) are encrypted prior to subsequent compression processing of the whole frame then there could be leakage of the MV data to the rest of the video frame. In [48] this is avoided by restricting the reference extent of MVs from within a ROI. It is found that this restriction results in no more than 1 to 2.5% on the bitrate, depending on the video content and the ROI size.

In [49] an interesting encryption scheme is outlined, in which encryption is applied prior to compression. Normally, encryption would not be applied prior to compression because it removes any exploitable correlation between pixels. Nevertheless, for a resource-constrained, possibly mobile device it may be better to perform encryption first and then on another device with greater resources perform compression prior to transmission. In [49] this is achieved for a set of spatially encoded images, which could be a sequence of video frames that are losslessly compressed. The method also requires the existence of a secret channel in which the first image in the sequence is transmitted. Other images are transmitted through the normal untrusted channel. Before, the first image is compressed a pixel sorting order is derived from that image. The sorting order is applied to the next image in the sequence, after which it is compressed, and so on for subsequent images. The authors of [49] also provide a variant of the scheme for 'lossy' compression. The sorting order or permutation is said in some cases to improve upon the existing compression performance, without encryption, for 'lossy' compression. However, the test sequences are by today's standards small and confined to somewhat static grayscale sequences. The main weakness of the method is that knowledge of the first image in the sequence, which is not encrypted but passed over a secret channel, compromises the confidentiality of all subsequent images.

The authors of [50] also offered an encryption-then-compression solution targeted at still images. The aim was to improve upon the compression performance of prior such encryption methods because according to [50], the compression ratio of prior solutions was generally weak. In [50] and some other proposed solutions of this type, side information is also generated which is then encrypted separately. In [50], the side-information is the prediction error that a given predictor would generate for each pixel's value given the surrounding pixels, which must in some way have caused the current pixel. The prediction errors are then clustered, with the number of clusters being a trade-off between improvement in compression and improvement in security. The clusters of prediction errors are then permuted, with the permutation information encrypted. Following that an arithmetic coder is employed to compress the prediction error clusters. Given a start pixel, the predictions errors, once recovered, allow the image to be recovered with the aid of the side-information. However, though the compression performance rivals that of state-of-the-art, JPEG2000 still-image compression of unencrypted images, it does rely on significant pre-processing and the encrypted communication of side information. Thus, though the encryption-before-compression approach offers greater flexibility compared to those SE approaches, the majority, that involve joint encryption and compression, this appears to be traded-off against performance in terms of the amount of compression needed and the current restriction to intra-coded images. The authors of [51] proposed a real-time protection scheme for HEVC. In this scheme, Truncated Rice (TR) codes were used for the binarization of Quantized Transform Coefficients (QTCs), RC codes being one of the enhanced choices offered by the HEVC version of CABAC. The solution works by first converting the TR codes' non-dyadic Encryption Space (ES) into a dyadic ES. After that AES encryption was applied to the selected syntax elements. The solution did not affect the ability of HEVC to operate in parallel because SE is applied independently to each and every coding slice. The encrypted bitstream appeared to maintain the same bitrate as the matching unencrypted bitstream. In addition, experiments were reported that showed a very limited computational cost due to encryption.

TABLE 1. Summary of recently proposed, SE-based encryption schemes, compared with SLEPX.

Proposed Scheme	Application	Algorithm used	Comparisons
Yang *et al.* [43]	HEVC-based video transmission system	AES algorithm used to encrypt a few of the encoded parameters	**Pros:** Secure, low computational cost, robust scheme **Cons:** Scheme not suitable either for low-power devices or real-time applications
H. Hofbauer *et al.* [52]	HEVC-based video transmission system	Transparent encryption works on AC signs	**Pros:** Fast encryption, format compliant, low computational time and bit-stream size preservation **Cons:** Weak encrypt-able parameters, AC signs demand increasing the number of I frames
Al-Salami *et al.* [53]	Smart Home	Private Key Generator, Diffie-Hellman (DH) stateful encryption	**Pros:** Offers transmission effectiveness as well as low computational cost **Cons:** Overhead of dealing with private key generator
Yao *et al.* [54]	IoT	Elliptic Curve Cryptography, DH, attribute based encryption	**Pros:** Suggests little communication cost, as well as growth in execution efficiency **Cons:** Poor scalability and weak elasticity in revoking attributes
Dufaux and Ebrahimi [55]	Video surveillance system	Pseudo-random sign inversion based encryption	**Pros:** Offers compression independent and format compliance scheme **Cons:** Scheme is not robust for transcoding, increases the bitrate and does not provide binarization for context modeling
Proposed cipher-SLEPX	SHVC bit-streams	SLEPX	**Pros:** Provides a sufficient level of visual protection, format compliancy, compression friendliness, efficiency for applications working on resource-constrained devices, and is transcodeable **Cons:** Security can be further enhanced by incorporating key-management schemes

In [21], practical scenarios in which encryption, watermarking, transcoding and/or compression are required for the same video crossing a network. Assuming, as the authors of [21] do that partial encryption or SE is the acceptable form of encryption in such circumstances, the authors analyze the impact of video quality transcoding (through alteration of the QP) upon watermarks. Notice that error drift through transcoding has long been the source of analysis in the literature [4] and consequently, in [21] quality transcoding is restricted to inter-code blocks to reduce such drift. In the case of watermarking, the authors of [21] select a compression domain watermarking method. They determined that HEVC encoding means that the re-quantization attack, in effect, of quality transcoding has much less impact on the watermark than similar transcoding of H.264/AVC streams.

In addition to the above analysis, Table 1 compares some recent SE-based schemes with the proposed cipher. Table 1 indicates some targeted applications, features of the algorithms employed and notes on advantages and disadvantages of these more recent contributions. These are compared with SLEPX. Further comparisons with SLEPX, concentrating on SE features, can be found in Table 9.

IV. SLEPX

The following section contains a detailed description of the SLEPX cipher.

A. SLEPX'S CABAC CONTEXT

SLEPX works in conjunction with CABAC, which is the only entropy coder supported by HEVC and, hence by SHVC. Bypass Binary Arithmetic Coding (BAC) elements (rather than elements passing through the Regular BAC module) do not affect context modelling [51] and consequently there is no context mismatch between the entropy coder and decoder if the fixed context of the Bypass BAC module is utilized.

Among all six basic HEVC binarization procedures used in the CABAC coder, only two codes, that of kth order Exponential (Exp)-Golomb coding (EG_k) and Truncated Rice Coding with context p (TRp) do not result, in general, in an encrypted bitstream being different in length from the input non-encrypted bitstream. Not all Truncated Rice Codings avoid updates of context models, as, for example, the context level p must be greater than zero [51]. In fact, the proposed cipher (SLEPX) encrypts MVD sign bits (one bit each) along with the NZ_TC sign bits and the absolute value of the suffixes (division remainders during coding), which are binarized by EG0 (NZ_TC sign bits and suffixes) and EG1 (MVD sign bits) respectively, i.e. with $k = 0$ and 1. SLEPX exhibits low-computational complexity and, as a result, it has a reduced impact on processing latency compared to other ciphers such as AES (see also Section I). Fig. 5 shows the insertion of SE within CABAC processing. Not shown in Fig. 5 is reference to context modelling, which takes place prior to the regular BAC, with retrospective update of the context models immediately after the output from regular BAC and prior to multiplexing of the output from the various forms of coding. Encryption by SLEPX takes place by assembling blocks of bits, selected according to the SE procedure. These encrypted bits are taken from their blocks and reinserted at the same place within the input entering the Bypass BAC module. Given that the key-length for SLEPX is a power of two, it is important to assemble blocks of SE elements that are also dyadic, i.e. a power of two. Specifically, for SLEPX the current key length is 128-bits and thus the assembled blocks of bits should be 128 bits in length. This procedure was first described in [51], where it was noted that HEVC

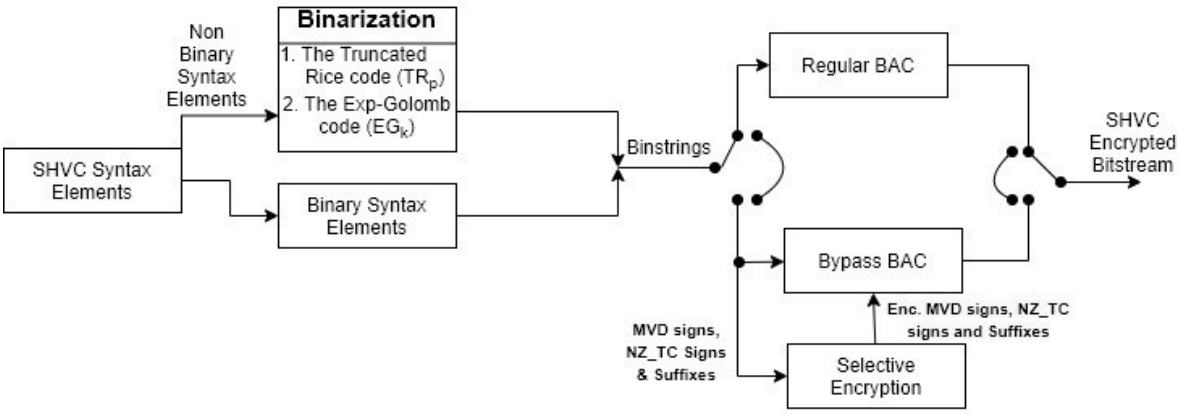


FIGURE 5. The insertion of SE within the CABAC coder.

(and hence SHVC) (in contrast to H.264/AVC) requires conversion from any non-dyadic lengths to dyadic lengths. After dyadic conversion and encryption, the SE bits are reinserted into the bitstream in such a way that the bitstream remains format compliant. Given the precautions mentioned in terms of elements selected and chosen binarization method, the resulting encrypted SHVC bitstream after BAC maintains bitstream length and, naturally, decoder compatibility

HEVC introduced the idea of entropy slices, which allows each slice to be separately coded and decoded. If more than one slice is defined for a frame, the advantage would be that entropy coding could proceed in parallel, as all context is independent for each slice. On the other hand, the data in each slice that is used to form the context is reduced, implying that adaptation of the context could be less sensitive for shorter slices.

B. SUMMARY OF SLEPX FEATURES

SLEPX has the following features:

- The cipher is a symmetric keystream-based scheme;
- For encryption and decryption purposes, the algorithm normally uses a 128-bit main key;
- For each encryption session, the cipher generates a different Initialization Vector (IV) by means of a Pseudo_Random_Number_Generator (PRNG) function, synchronized between encryption and decryption modules. Thus, synchronization of the PRNG between the encryption and decryption modules is also part of symmetric key exchange management, which synchronization, however, is outside the scope of this paper the main key and the IV are initially XORed together. In this way, the XOR function generates a new 128-bit key (namely the session key), which is split into two sub-keys (see Section IV-C). This procedure ensures that every time an encryption session takes place, the process always generates random results, even when the data are the same or the main key is the same or both. Exchange of the main key and either the session key or the two sub-keys through key management [41] is outside the scope of the current paper.
- The cipher was developed to work with the SHVC standard but, in fact, it could easily be applied to HEVC as well; and
- If the cipher is employed with HEVC then it can behave like a crypto-transcoder, i.e. a selectively encrypted bitstream can further be transcoded directly without showing the original video contents to 3rd parties, unidentifiable or not, as might happen within a CDNaaS [56].

C. ENCRYPTION ALGORITHM

Confusion and diffusion are the fundamental requirements of any encryption algorithm [57] and SLEPX fulfills both requirement in the sense that it uses both permutation and XOR functions. Both functions deal with a selected bitstream (comprised of binstrings) arranged as 128-bit acts blocks and transformed into a 128-bit block encrypted bitstream. To fulfill the confusion property of the cipher, an Initiation Vector (IV) is used to alter the output for every encryption session. The SLEPX IV is generated by a PRNG function by utilizing the session key for each session as a seed, thus generating a different IV every session, behaving like a One-Time Pad. Subsequently SLEPX will deliver a different output for the same session key for each encryption session. The main reason to use this one-time pad mechanism is to thwart chosen plaintext and known-plaintext attacks. As previously mentioned, and as shown in the encryption box of Fig. 6, the 128-bit key will be Exclusive-ORed (XORed) with the IV. Subsequently, the resulting modified 128-bit key is divided into two equal sub-keys (K_1 and K_2 in Fig. 6) (each 64-bits in length).

In many block ciphers, such as AES [16], a Substitution box (S-Box) introduces a non-linear element to the cipher, which, unfortunately increases the computational cost of those ciphers. To make the cipher more suitable for real-time processing an S-Box function is not utilized in SLEPX. However, XORing, which is also a non-linear process, is employed. In fact, the encryption algorithm converts the selected binstrings into blocks of 128-bits each, through the dyadic conversion process mentioned in Section IV-A.

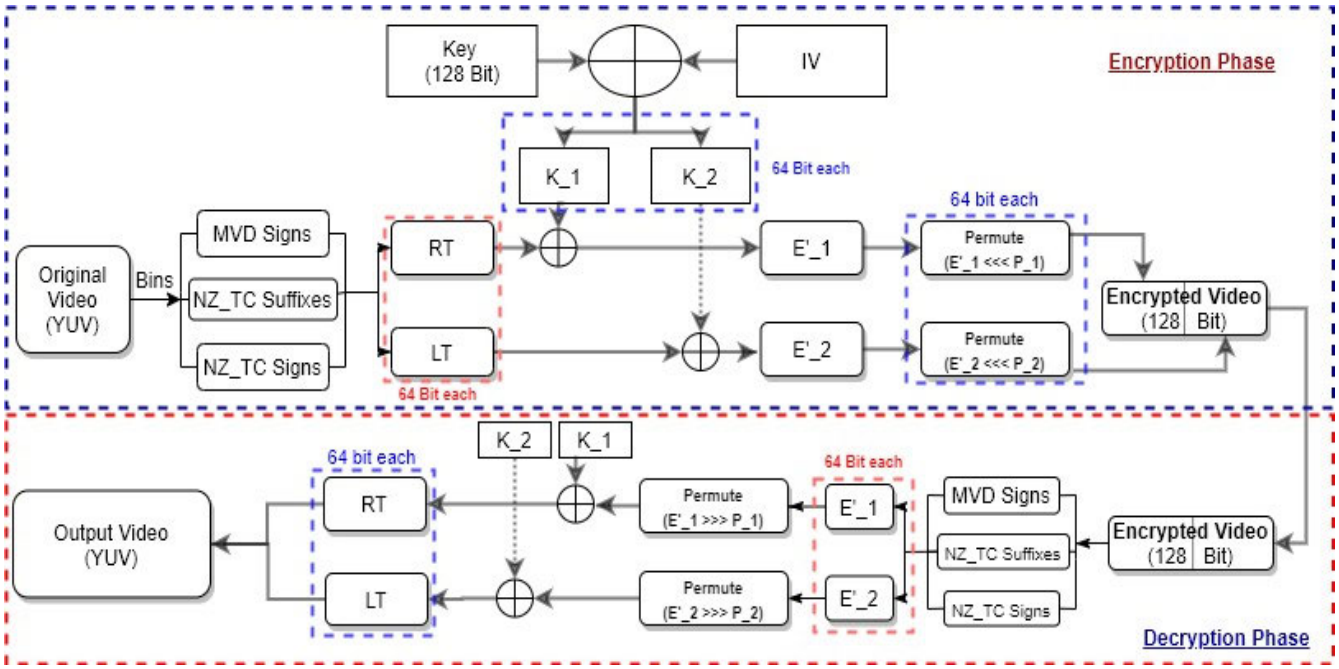


FIGURE 6. Flow diagram of SLEPX. Notice that this diagram only shows the Bypass BAC segment of the video, which is multiplexed with the output of the Regular BAC etc. It also does not show the application of arithmetic coding to both the Regular and Bypass entropy coded elements (see Fig. 5), Notice also that for reasons of space the input binstrings have been abbreviated to bins.

The algorithm then splits each block into two halves (referred to as the RT and LT). The RT and LT sub-blocks are then separately XORed with sub-keys K_1 and K_2 respectively. The XORed sub-blocks are subsequently permute each bitwise by a randomly selected offset (namely P_1 and P_2) ranging from 1 to 9. As indicated in Fig. 6, the type of permutation is a left-wise circular shift, which is easily implemented in hardware, if need be. Apply the randomized circular shifts to the outputs of the earlier XORs makes the final output statistically independent from the results of the former step, while these permutations also meet the need to include a confusion step, as mentioned at the start of this Section. After the permutation step both halves are concatenated to restore a 128-bit stream of blocks (now encrypted).

Subsequently, the encrypted blocks, after decomposing back into their binstring parts are combined with the un-encrypted binstrings to form a bitstream which undergoes BAC prior to multiplexing with the regular BAC etc. to form the final output compressed bitstream, in this manner, a receiver decoder gets a compressed video bitstream which fulfills the format compatibility requirements, as well as maintaining the statistical properties of the bitstream, i.e. it is uniformly distributed, and its length. Moreover, the SLEPX cipher works effectively in a distributed environment in that it encrypts sub-parts of the original bitstream independently of each other. The pseudo-code of Fig. 7 is a confirmation of the operation of Fig. 6, as it is important to clarify the operation of a cipher.

The encryption space for the suggested SHVC SE cipher works on the MVD sign bits, and the NZ_TC sign and suffixes bits, (the former being discussed in [58]). Arithmetic coding is very sensitive to errors so that a single alteration in a bit might cause the entire bitstream to become non-format compliant [58]. As a result, in this research the authors encrypt the binstring after the binarization process. The impact of the SHVC based SE scheme is to encrypt the video content having different layers (for scalable video) by guaranteeing the same bit rate, format compliance features and real-time requirements.

D. SHVC OPERATION OF SLEPX

Videos encoded by an SHVC encoder contains base layer and number of enhancement layers, normally based on one or more of the three scalabilities of spatial, temporal and quality at several different resolutions. In SHVC, spatial layers involve the adjustment of coefficients only. Moreover, every temporal layer requires the conversion of transform coefficients, delta QPs (dQPs), and MVDs. Signal-to-Noise Ratio (SNR) layers deal with the conversion of transform coefficients and dQPs. Such SHVC layer behaviors indicate that the signs of coefficients (the signs of the NZ_TCs) and their suffixes are the most appropriate parameters for SHVC layer encryption, because every kind of scalability deals with these coefficients. MVD sign encryption is more appropriate for temporal scalability, but it is significant for all three scalabilities, as SNR and spatial scalability is commonly combined with temporal scalability.

Fig. 8 shows SLEPX operating across multiple layers of an SHVC encoder. A SLEPX encryption module is inserted at each layer of SHVC, within the BL and within one or more ELs.

Pseudocode of SLEPX Cipher

1. Input Original YUV Video (referred as OV)
2. Binstrings: MVD signs, and NZ_TC signs and suffixes
3. IV = Pseudo_Random_Number_Generator()
4. [Generate session key for encryption function]
   Key = Key ⊕ IV
   [128-bit Key XORed by an IV]
5. *K [] = splitter (Key)
   [128-bit Key split into two sub_keys, 64-bits each and stored into *K_1 and *K_2 respectively]
6. Form a 128-bit data Block by the dyadic conversion process. (Blocks are based on MVD signs, and NZ-TC signs and suffixes data)
   *Blocks [] = splitter (Block)
   [128-bit data Blocks split into two sub-blocks of 64-bits each, and each block stored separately into *RT and *LT respectively]
7. XOR function applied individually to each block of data
   {
8. *E'_1 = (RT ⊕ *K_1) <<< P_1 [e.g. P_1 = 2]
9. *E'_2 = (LT ⊕ *K_2) <<< P_2 [e.g. P_1 = 4]
   }
10. *EV = merger (*E'_1, *E'_2)
    [merger() concatenates two 64 bit blocks of data into a single 128-bit block]
11. Output encrypted video stream (referred as *EV) after including the selectively encrypted elements (MVD signs, and NZ_TC signs and suffixes) into the Bypass BAC bitstream

FIGURE 7. Pseudocode of SLEPX.

V. RESULTS AND ANALYSIS

This Section describes the experiments performed upon the SLEPX cipher upon different video sequences, along with an analysis of SLEPX's impact uon those sequences.

A. EXPERIMENTAL SETUP

The proposed encryption algorithm was implemented within the Scalable Reference Model (SHM) encoder version 7.0 [59]. The decryption procedures, the counterpart of SLEPX encryption, are based upon the improved SHVC decoder, referred to as OpenHEVC [60]. The results help to assess the computational cost of the decryption method by supposing the SHVC decoder to have real-time performance features. For evaluation purposes, different experiments conducted within common SHVC test environments, as given in [61]. The detail of the videos acting as test sequences are mentioned in Table 2. The original raw video sequences for the experiments were sourced from [62] and [63].

Video frames were encoded into four layers, i.e. the BL (Layer 0) and three ELs (layers 1–3). The chroma sampling utilized was 4:2:0 [4] and the encoding method utilized was Variable Bit Rate (VBR) [4]. We consider four Quantization Parameter (QP) configurations (QP = 12, 24, 36 and 48).

TABLE 2. Test video sequences utilized.

Class	Video Seq.	Resolution (pixels/frame)	Frame Rate (Hz)
A	Jockey	4096 × 2160	50
B	Beauty	1920 × 1080	50
	Kimono		
C	Four People	1280 × 720	40
	Mobcal		
D	Paris	352 × 288	30
	Foreman		

Objective video distortion in the video samples is described in decibels (dB) for Peak Signal to Noise Ratio (PSNR) [64] for the YUV video signals. Structural distortion in the video sequences was measured by using the SSIM index [47], which targets the human perceptual response with a valued score ranging from 0 to 1.

Figs. 9 (a), (b), (c) and (d) represent the sample encoded frames, QP = 12, as mentioned in Table 2, along with their PSNR and SSIM. Then, Figs. 9 (a1), (b1), (c1) and (d1) present the visual effect arising from SE upon the same video frames. Thus, Fig. 9 presents the impact of the encryption technique, which for these frames makes them unwatchable, though the level of distortion across some frames is variable, as a result of the content characteristics and the syntax elements selected for encryption. For example, the face in the original encoded Beauty video frame is highly distorted in Fig. 9 (a1), although there is little else of interest in the frame shown.

B. VIDEO QUALITY METRICS

For accurate assessment, the PSNR and SSIM of the mentioned video sequences (in Table 2) were assessed by luminance (Y) as well as chrominance (two color components U and V). Luminance is considered to be the most significant element in pictorial recognition. However, chrominance values of distortion are also given because it is possible that the chrominance components could be decoded separately, as a way of establishing the content. Recall from Figs. 4 and 8 that in non-hybrid SHVC, BL and ELs are encoded independently of each other except that upscaling for possible spatial resolution layering and MV scaling in the case of possible temporal layering, speed up the process of spatial and/or temporal layering for ELs. If those forms of scalability are not employed but SNR scalability is applied then the QP is applied independently of other layers by application of different values of QP at each layer, with ascending quality with higher ELs in the hierarchy. Thus, setting a QP value determines the video quality at whatever layer, making subsequent comparisons across QP settings possible.

For calculation of the PSNR and the SSIM index video-quality metrics eqns. (1) and (2) respectively were applied:

$$PSNR = 10log_{10}\frac{(2^x - 1)^2}{MSE} \quad (1)$$

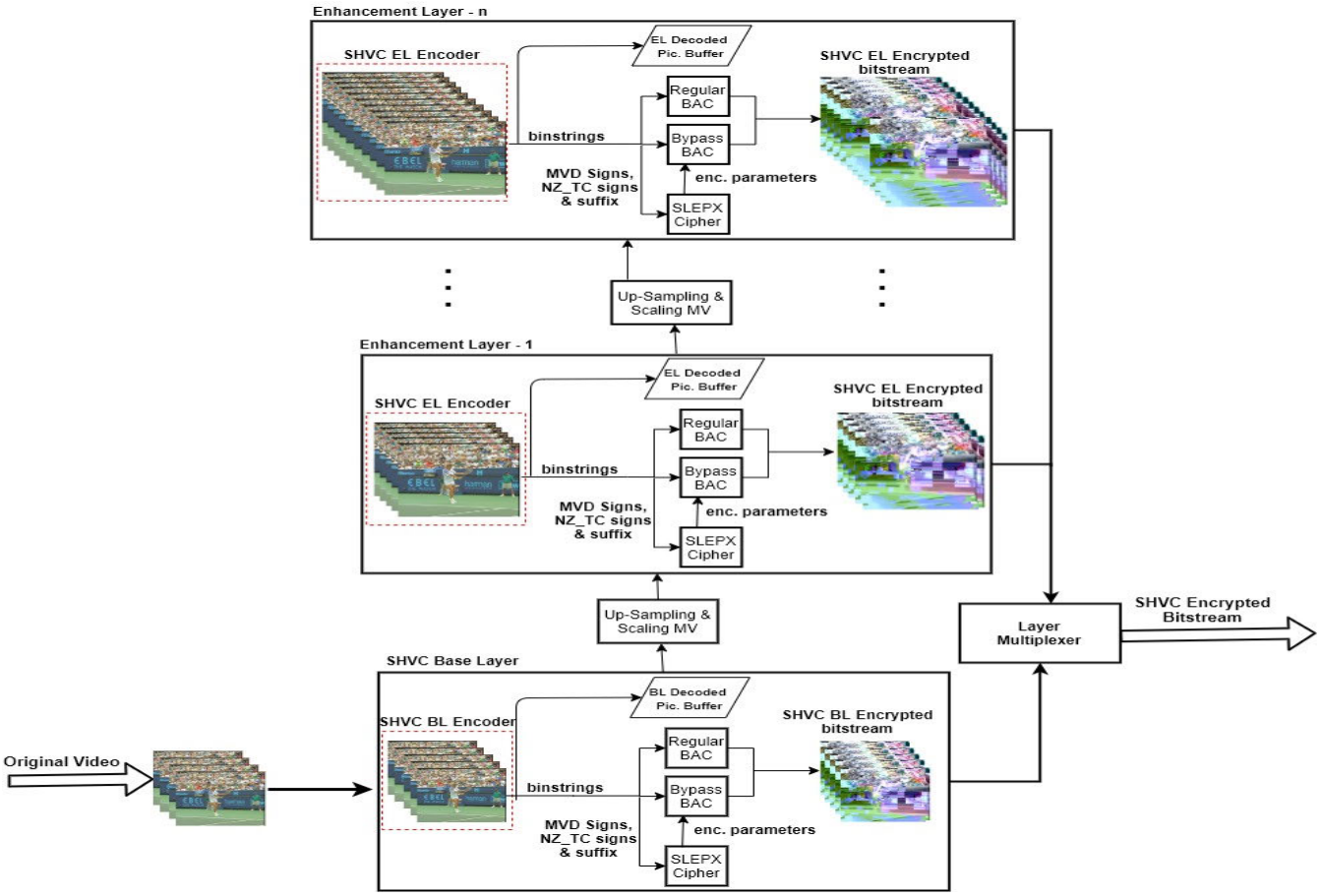


FIGURE 8. SLEPX operating across multiple layers of SHVC.

where MSE refer to the Mean Square Error between the original plain video sequence and the resulting processed video obtained after applying the SE scheme, and x refers to the bits per pixel. The PSNR can also be found for individual video frames, rather than averaging the PSNR across a sequence of frames.

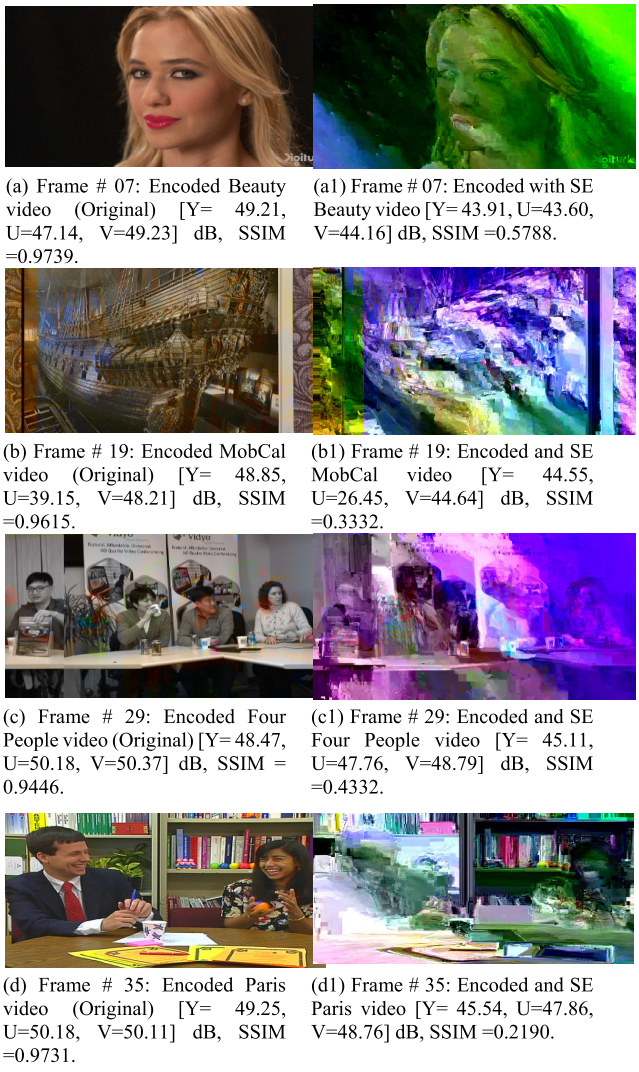
$$SSIM\ (a, b) = \frac{(2\mu_a\mu_b + c1)(2\sigma_{ab} + c2)}{(\mu_a^2\mu_b^2 + c1)(\sigma_a^2 + \sigma_b^2 + c2)} \qquad (2)$$

In equation (2) the parameters a and b refer to the pair of video frames being compared, with μ_a and μ_b being the average (arithmetic mean) value of pixel (intensity) within two video frames labelled as a and b, with σ_a and σ_b being the variance of the intensity values within video frames a and b video respectively, while σ_{ab} refers to the covariance. Variables $c1$ and $c2$ are utilized to facilitate calculations when the denominator terms in (2) are particularly small [47]. It should be mentioned that PSNR remains a widely employed metric when making comparisons between different research papers on the same video sequence. It is also employed within codecs for estimating the rate-distortion of coding. SSIM is said to be helpful in reflecting the relative human response if subjective testing is not available for whatever reason. Currently it is less widely deployed within codecs.

TABLE 3. Comparison between the PSNR of 'Plain' and SE videos, with QP = 12.

Videos	Plain PSNR (dB)			SE PSNR (dB)		
	Y	U	V	Y	U	V
Jockey	39.825	49.298	46.185	21.676	18.778	19.211
Beauty	41.759	47.127	49.199	22.210	24.846	26.200
Kimono	47.301	41.008	43.490	19.390	22.178	21.160
Four People	48.206	43.175	43.886	11.190	15.499	14.392
Mobcal	48.851	45.153	47.214	16.814	22.051	21.127
Paris	49.254	50.181	50.114	19.498	11.393	10.047
Foreman	36.099	41.901	42.971	19.878	21.249	20.419

In Tables 3 and 4, with QP = 12, the term 'plain' denotes the unencrypted form of a video sequence, whereas 'SE' denotes the encrypted version of the same plain videos. As is clear from Table 3, the average (arithmetic mean) PSNR of both the luminance and chrominance components are much degraded after applying SE over the entire video sequences. The results for SSIM in Table 4 tell a similar story to that of Table 4.



(a) Frame # 07: Encoded Beauty video (Original) [Y= 49.21, U=47.14, V=49.23] dB, SSIM =0.9739.

(a1) Frame # 07: Encoded with SE Beauty video [Y= 43.91, U=43.60, V=44.16] dB, SSIM =0.5788.

(b) Frame # 19: Encoded MobCal video (Original) [Y= 48.85, U=39.15, V=48.21] dB, SSIM =0.9615.

(b1) Frame # 19: Encoded and SE MobCal video [Y= 44.55, U=26.45, V=44.64] dB, SSIM =0.3332.

(c) Frame # 29: Encoded Four People video (Original) [Y= 48.47, U=50.18, V=50.37] dB, SSIM = 0.9446.

(c1) Frame # 29: Encoded and SE Four People video [Y= 45.11, U=47.76, V=48.79] dB, SSIM =0.4332.

(d) Frame # 35: Encoded Paris video (Original) [Y= 49.25, U=50.18, V=50.11] dB, SSIM =0.9731.

(d1) Frame # 35: Encoded and SE Paris video [Y= 45.54, U=47.86, V=48.76] dB, SSIM =0.2190.

FIGURE 9. Effect of structural distortion in videos with SLEPX at QP 12.

TABLE 4. Comparison between the SSIM values of 'plain' and SE videos, with QP = 12.

Video Sequences	Plain SSIM	SE SSIM
Jockey	0.9442	0.4163
Beauty	0.9739	0.4688
Kimono	0.9641	0.3814
Four People	0.9446	0.4232
Mobcal	0.9615	0.3932
Paris	0.9731	0.3390
Foreman	0.9519	0.3848

The distortion was also evaluated at different QP values. Recall that the QP controls the granularity of quantization [4], having values 0–51 within HEVC [7], with lower values of QP resulting in better video quality. For visual analysis, Fig. 10 illustrates the video quality of an EL (using SNR scalability) of video sequences selected from Table 2, namely Beauty, Four People, MobCal, and Paris at different QPs, before and after applying SE. SE significantly distorts the luminance and chrominance components, as is also revealed by the SSIM scores. As the video quality is better in frames having QP = 12, then, after encryption, the higher encrypted ELs introduce more distortion than lower ELs video frame (QP = 24, 36 and 48). Those scenarios (with higher QPs or equivalently lower video quality in the plain versions) can also be seen in Fig. 10. Thus, an encrypted layer having QP = 48 displays less distortion compared to an encrypted EL having QP = 12. This behavior is related to the relative number of syntax elements selected for SE at the different QPs. Thus, at lower QPs (higher video quality) there are relatively more of such syntax elements than at higher QPs (lower video qualities). On the other hand, a video with (say) QP = 48 is already distorted relative to one with QP = 12. Therefore, it seems that there is a tradeoff, trading the amount of distortion from compression against the amount of distortion from SE.

C. COMPUTATIONAL COST

The time needed to perform the encoding (with and without SE) using the SHVC standard for reference video sequences was evaluated, with the results shown in Fig. 11. The additional time needed to apply the SE procedure on each video sequence is referred to as encoding delay, found to be on average 69.19 ms over the set of video sequences. (The delay was calculated by subtracting the encoding times with SE from those times without SE.). To the human observer, this average time may well be imperceptible. However, for much longer sequences and when processing batches of videos then, as with all encryption, there is a latency impact. However, Section V-E.2 will compare the computational overheads between SLEPX, simple XOR and industry standard AES, when it will be shown that SLEPX is a good compromise in terms of confidentiality versus computational overhead.

D. SECURITY ANALYSIS

This Section considers different experiments aimed at validating security aspects of the SLEPX cipher in respect to standard objectives and measures of security robustness. The investigations include a histogram evaluation of encrypted relative to original video frames, edge differential ratio analysis, correlation coefficient analysis, distribution of selected encryption parameters, and the linear regression model of encrypted syntax elements respectively.

1) HISTOGRAM ANALYSIS

A histogram enquiry is a visual illustration of the image pixel distribution which shows the intensity level of each color pixel [65]. For an encryption system, comparisons of histograms of encrypted and original video frames shows the differences between plain and SE forms. The histograms of the original video frames taken from the video sequences under test, i.e. Beauty, Four People, Mobcal and Paris, are shown in Fig. 12 (a) (c) (e) and (g), whereas the histograms of the SLEPX versions of the chosen video frames are shown

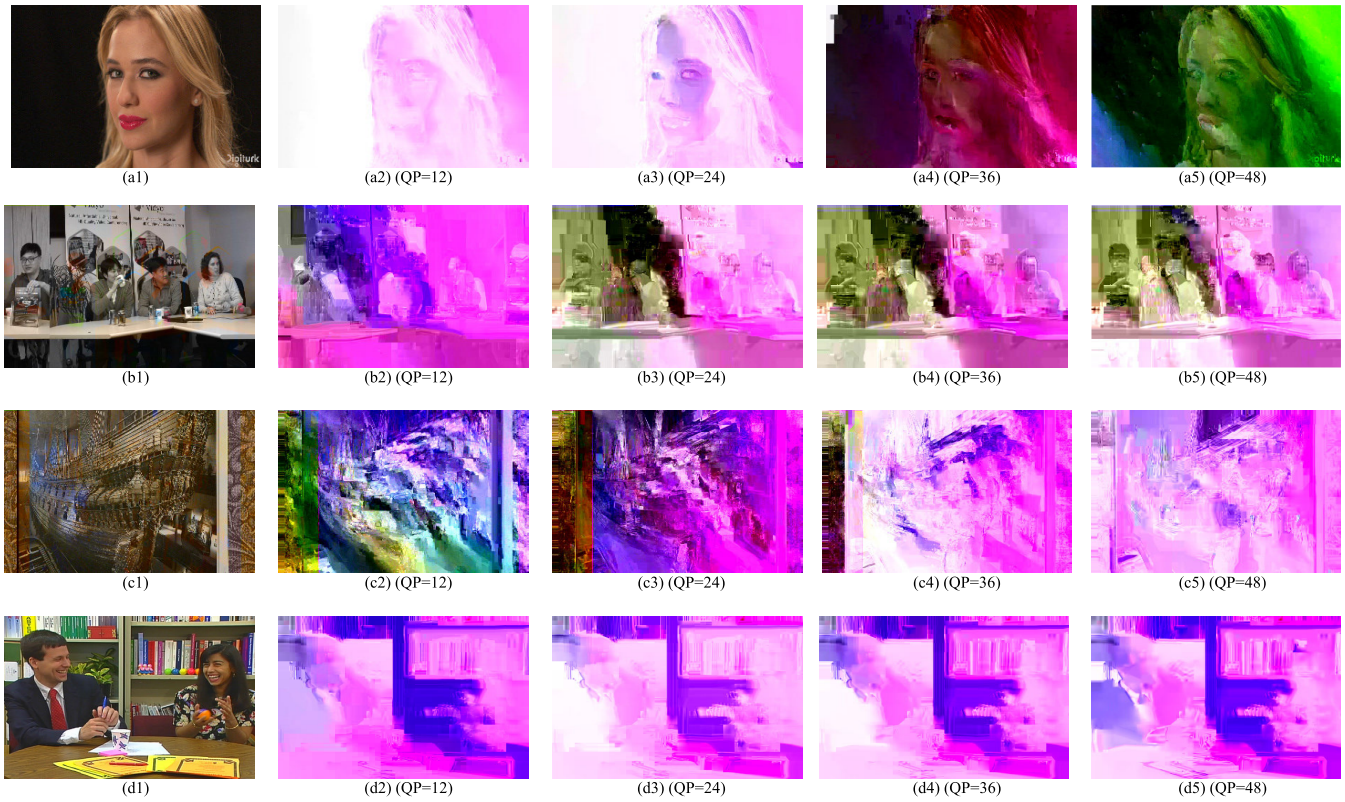


FIGURE 10. Impact of visual distortion of SLEPX at different QP levels with respect to various videos. Parts (a1) – (d1) show frames #07, 29, 19 and 35 of the plain encoded versions of *Beauty, Four People, MobCal* and *Paris* video sequences respectively. Parts (a2)-(d2), (a3)-(d3), (a4)-(d4) and, (a5)-(d5) show the encrypted versions of *Beauty, Four people, MobCal* and *Paris* video sequences for the indicated QP.

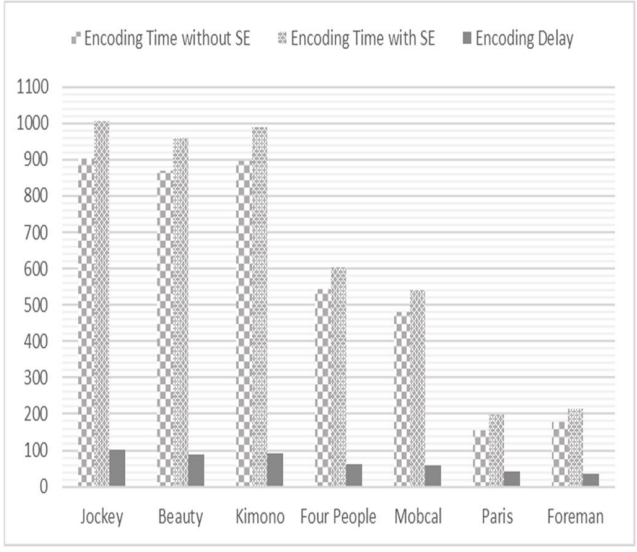


FIGURE 11. Effect of encoding delay (in ms) of SLEPX at QP = 12.

in Fig. 12 (b) (d) (f) and (h). It is apparent that the histograms (original and encrypted video frames) are certainly dissimilar.

2) EDGE DIFFERENTIAL RATIO

The Edge Differential Ratio (EDR) estimates the edge dissimilarities between the encrypted and original video sequences [66], [67]. The effectiveness of any cipher can be judged by the impact of encryption upon the edges, which should not be preserved in the encrypted form of any video frame.

The mathematical formula used to calculate EDR between original and encrypted video frame, given as (3) is as follows:

$$EDR = \frac{\Sigma_{i,j=1}^{N} \left| S\left(i,j\right) - \bar{S}(i,j) \right|}{\Sigma_{i,j=1}^{N} \left| S\left(i,j\right) + \bar{S}(i,j) \right|} \tag{3}$$

where S(i, j) and $\bar{S}(i, j)$ are the pixel values of the detected edges in the original and encrypted video frames. The maximum EDR value which indicates the highest confidentiality according to objective statistics for a given video frame must be equal to or close to '1' and *vice versa* the minimum value is '0. Fig. 13 shows the resulting images after using Sobel edge detection [68] to find the edges for the original images and after SLEPX was used. The QP was 18, i.e. a high-quality, before SE. From visual inspection, once SE has been applied there remains little to be gained applying Sobel edge detection to the resulting video frames in terms of identifying imagery within the distorted video frames. In fact, the EDR values for the selected frames from MobCal, Four People and Paris were 0.94, 0.91 and 0.90 respectively, which confirms the effectiveness of SLEPX combined with SE by showing that the encrypted and original video frames have EDRs close to one. Sobel edge detection is a staple method of image

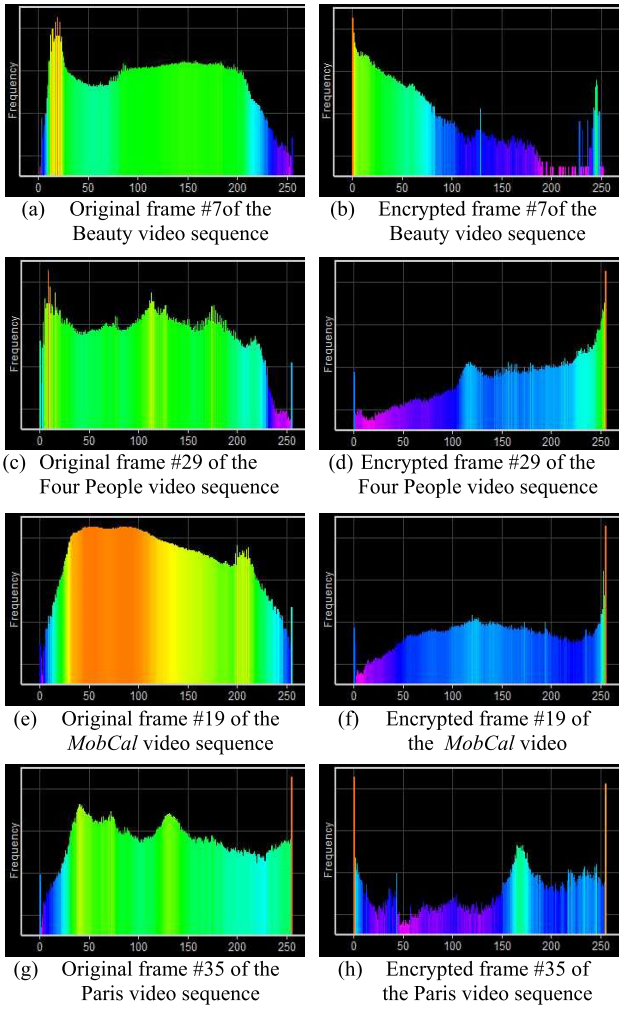


FIGURE 12. Histograms of original and encrypted video frames selected from the *Beauty*, *Four People*, *Mobcal* and *Paris* video seq. with QP = 18.

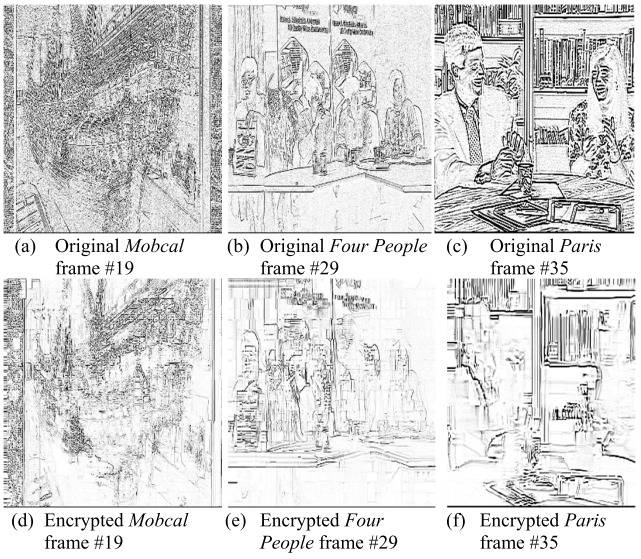


FIGURE 13. Visual results of EDR of different frames belonging to Mobcal, Four People and Paris video sequence with SNR scalability and $QP_{EL} = 18$.

processing, though there have been many alternative edge detection methods, a very recent method being the active contour method described in [69]. Therefore, Fig. 13 is an estimate of how edge detection might aid deciphering an encrypted video frame, i.e. have little effect. However, given that in [69], the active contour method is only tested on three images, much needs to be done to confirm the robustness of this very new edge detection method, though comparative results in [69] indicate that it is promising. Future work will involve checking whether other edge detection methods informed by artificial intelligence can improve much in deciphering already distorted video frames. There is also the issue of real-time performance, which was not tested in [69], while Sobel edge detection is strong in that respect.

3) CORRELATION COEFFICIENT ANALYSIS

In the original plain video frames, the correlation between spatially adjacent pixel intensity values is high, as the video sequences are natural images, which usually have statistically formed from a Markov field of order one [70]. Accordingly, if the encrypted video frames still having a high degree of correlation between frame pixel values then it is possible that the contents of the original video frame will be discernible. Hence, suitable encryption methods always reduce the correlation among neighboring pixel values after encryption. Basically, correlation coefficient analysis quantifies the linear dependency of two neighboring pixels in the same video frame or matches pixel values in altered video frames at similar locations [71], [72]. SE combined with SLEPX results in low correlations between neighboring pixels, as can be confirmed statistically. The r correlation coefficient can be calculated by means of eqn. (4):

$$r(x, y) = \frac{1}{n-1} \sum_{0}^{n} \left(\frac{x_i - \bar{x}_i}{\delta_x} \right) \left(\frac{y_i - \bar{y}_i}{\delta_y} \right) \tag{4}$$

In (4), n refers to the number of pixels in a video frame, x_i and y_i are the intensity values of two pixels at position i in two different video frames, $\bar{x}_i$ and $\bar{y}_i$ are the horizontal and vertical local means and δ_x and δ_y are denote the standard deviation of pixels x and y correspondingly.

To examine the analysis of correlation coefficient of the SLEPX, the following steps occurred:

- Selection of 1000 pixels randomly from the video sequences under test (Jockey, Kimono, Four People and Paris).
- Selection of the equivalent 1000 pixels after SLEPX and SE at identical locations.
- Computing the correlation coefficient for those 1000 pixels utilizing equation (4).

Table 5 shows the resulting correlation coefficients analysis for the 1000 pixels. If the degree of correlation (based on pixel values) between the original and encrypted frames is high (near to 1) then that shows a minimum level of encryption; hence, the encrypted data can be compromised straightforwardly. The correlation coefficients findings for

TABLE 5. Correlation coefficient pixel values of the Jockey, Kimono, Four people and Paris between plain and encrypted video using SLEPX with SE.

Video Sequences	Resolution (pixels/frame)	Correlation Coefficient
Jockey	3840 × 2160	-0.2146
Kimono	1920 × 1080	0.1591
Four People	1280 × 720	0.0926
Paris	352 × 288	-0.1915

the SLEPX cipher upon different video frames present particularly low values, as confirmed in Table 5, thus revealing the effectiveness of SLEPX cipher in terms of offering protection through distortion of the video.

4) SCATTERING OF CHOSEN ENCRYPTION PARAMETERS

The distribution of selected encryption parameters within a video provides better video confidentiality resilience. If syntax elements are chosen that are abundant within a video sequence and not bunched within a few video frames, leaving other frames unaffected then the distortion is maximized and confidentiality is improved. The same consideration is also significant in respect to resisting cryptanalysis of a given SE technique, as if the distribution of selected syntax elements is properly managed and, in addition, the amount of selected elements is large then it could well be difficult to predict them all properly, reducing the ability to reconstruct the contents through (say) a substitution attack.

The extracted statistics shown in Table 6 are from seven dissimilar video sequences with different file sizes. These provided the basis for a distribution analysis. The most likely distribution matching the probability distribution of chosen syntax elements is a Poisson Distribution because the distribution is discrete, the probability of occurrence of a single event (a given number of syntax elements for any one chosen element) does not affect other events, and the count size range is, in theory, is unbounded.

TABLE 6. Counts of occurrences of selected syntax elements in test videos.

Video Seq.	File Size (in MB)	No. of encoded Frames	Signs of MVD	NZ_TC Suffixes	Signs of NZ- TC
Jockey	3353	50	964208	73029	6895435
Beauty	2078	75	765394	49220	4383004
Kimono	2226	75	798340	43981	3197156
Four People	831	100	698230	55912	2838680
Mobcal	666	100	713941	49302	3239530
Paris	155	150	457391	7391	2108654

5) LINEAR REGRESSION MODEL OF ENCRYPTED SYNTAX ELEMENTS

Regression analysis is a statistical approach that measures the association between two or more variables of concern. Given a good fit with a linear model of regression then a proposed cipher will have an avalanche effect upon confidentiality.

That is confidentiality may well improve linearly across a video sequence as more and more video frames are added. To find the linear regression of each of the encrypted syntax elements compared to the independent variable 'file size'. Equation (5) is normally used in linear regression analysis, in this instance to establish the extent of linearity of encrypted parameters with respect to the file size [73]:

$$Y = \alpha + \beta X \tag{5}$$

where Y is used as a dependent variable (encrypted syntax element), X denotes the independent variable (file size) and α and β are the values of the coefficients computed by utilizing the available data. Analysis of the linear regression is conducted by computing 95% confidence intervals for both higher and lower series. The confidence interval for 95% is the interval lying between the upper and lower values, within which there is a 95% probability of finding the population value, as opposed to the calculated sample value. That is, based upon the sample values, here for α or β, the confidence values say that there is a 95% probability that if all possible file sizes were available (the population) then the values of α or β would be found within that confidence interval. The test sequences mentioned in Table 2 with dissimilar file sizes were used in this experiment. The dependent variable Y takes its values from the results the number of MVD signs, NZ-TC signs, and NZ-TC suffixes. Table 7 tabulates the values of the two coefficients from eqn. (5) along with the confidence intervals. Using those values allows construction of eqns. (6)–(8) for MVD signs, NZ_TC suffixes and NZ-TC signs respectively.

$$\mathrm{Y} = 538853.18 + (125.082)\ \mathrm{X} \tag{6}$$

$$\mathrm{Y} = 26544 + (12.844)\ \mathrm{X} \tag{7}$$

$$\mathrm{Y} = 2\mathrm{E} + 06 + (1248.5)\ \mathrm{X} \tag{8}$$

TABLE 7. Linear regression analysis results for MVD signs, and NZ-TC suffixes and signs.

	Coefficients	Values	Lower 95%	Upper 95%
MVD Signs	α	538853.18	389479.011	688227.346
	β	125.082	46.489	203.674
NZ-TC Suffixes	α	26544	-6736.78	59825.44
	β	12.844	-4.666	30.35521
NZ-TC Signs	α	2E+06	109712.5	3570452
	β	1248.5	338.037	2158.894

The effects of the linear regression investigation for all the nominated syntax elements are revealed in Fig. 14. The two graphical plots depict the relationship between the number of MVD signs and NZ-TC signs in relation to the file sizes, showing that there is a connection between video file size and the number of a particular encrypted element, though the connection is firmer for NZ-TC signs than it is for MVD signs. Moreover, the analysis also concluded that the numbers of signs of NZ-TC and MVD elements will be very large,

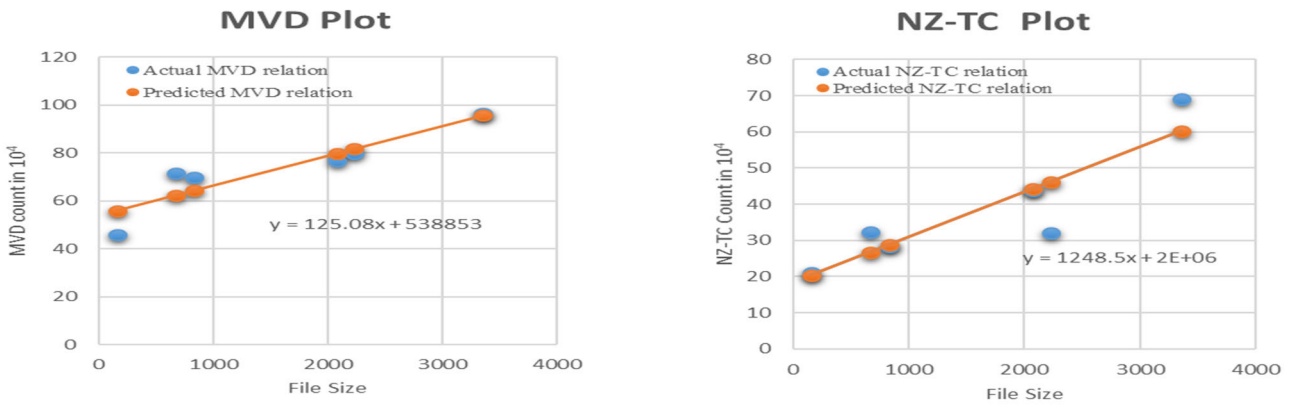


FIGURE 14. Graphical illustration of linear regression analysis for encrypted MVD signs and NZ-TC signs.

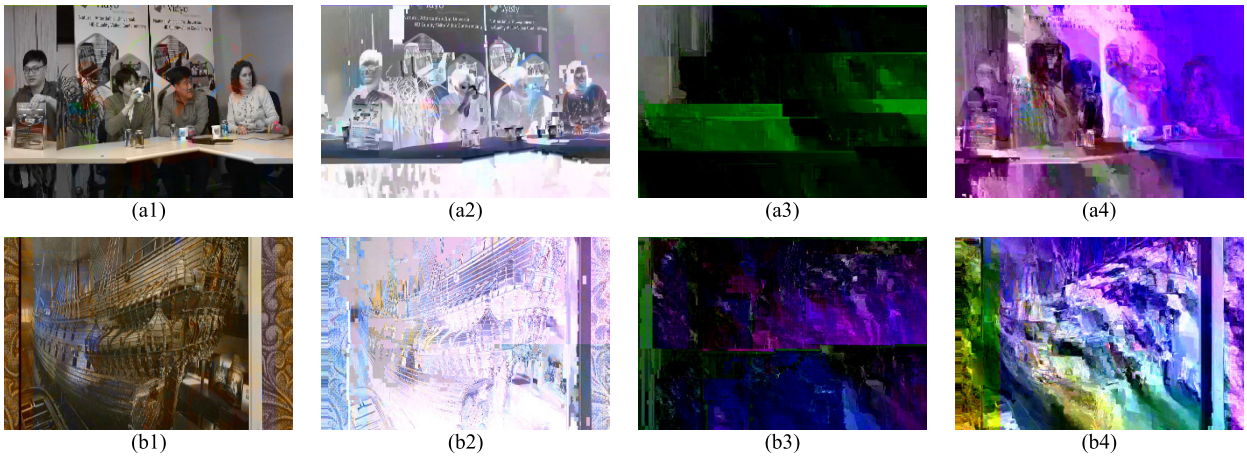


FIGURE 15. Visual Effects comparison of SE (on I, P and B video frames) on both the texture and motion syntax elements with XOR, AES-CFB and SLEPX on two frames from the Four People and MobCal test videos at QP = 24 encoded with CABAC-based SE. (a1-b1) Original Four People and Mobcal videos frames; (a2-b2) encrypted with XOR; (a3-b3) encrypted with AES-CFB; (a4-b4) encrypted with SLEPX.

making it difficult to predict these or conduct a substitution attack. The number of NZ-TC suffixes is not so great, making this syntax element more vulnerable to such an attack.

E. COMPARATIVE EVALUATION AGAINST OTHER CIPHERS

To find the effectiveness of the SLEPX on the one hand it was compared with base-level XOR encryption and, at the other end of the cipher sophistication, industry-standard AES. Simple XOR can be implemented in hardware for very rapid block-encryption and introduces a non-linear element into the processing. Both AES and XORing as a component of SLEPX were already described in Section IV-C. AES was used in Cipher Feedback Mode (CFB) mode [74], which is one of a number of well-established and common modes of operation for block ciphers. Similarly to SLEPX, CFB is initiated with an IV (refer back to Section IV-C). CFB allows AES to operate as a stream cipher, once the IV has initially been applied. Thereafter only the key is applied to each block of plaintext. CFB has the advantage of self-synchronization, which means that if a block of data is lost, after several blocks of further input decryption can continue. It has been used before for CABAC-based SE [51] [37].

Fig. 15 allows visual comparisons of the effect of encryption by XOR, AES-CFB and SLEPX. In Fig. 15 (a2)–(a4) and (b2)–(b4) portray the SE of the Four People and MobCal videos after encryption with the three ciphers. It is apparent that for XOR encryption, Fig. 15 (a2) and (b2) parts of the original scenes are still discernible, less so than for SLEPX in parts (a4) and (b4).

1) COMPARATIVE VISUAL QUALITY ANALYSIS

PSNR and SSIM (refer to Section V-B) were calculated to make a quantitative comparison between the impact of SE with the XOR, AES-CFB and SLEPX ciphers. PSNR values are tabulated in Table 8, results based on the encryption performed by utilizing both the MVD and NZ_TC based syntax elements at different QP values. The luminance (Y) parameter of PSNR of *Four People* video sequence was 21.28 dB, 10.87 dB and 11.19 dB after SE performed by XOR, AES-CFB and SLEPX at QP = 12 respectively.

TABLE 8. Comparison of avg. PSNR (in dB) of encoded plain video and SE with XOR, AES-CFB and SLEPX of diff. videos at QP levels (12, 24, 36 and 48).

Videos	QP	Plain PSNR			SE with XOR			SE with AES-CFB			SE with SLEPX		
		Y	U	V	Y	U	V	Y	U	V	Y	U	V
Four People	12	48.47	41.18	42.37	21.28	24.50	19.39	10.87	13.57	11.57	11.19	15.50	14.39
	24	40.15	35.42	39.81	23.49	26.84	21.60	11.25	14.01	13.82	12.15	16.12	15.12
	36	31.59	31.39	35.71	26.82	26.79	23.51	13.86	15.54	15.49	14.89	16.83	16.05
	48	25.51	29.18	31.27	27.48	27.53	23.78	14.59	16.82	16.55	16.11	17.76	18.79
MobCal	12	48.85	45.15	47.21	24.81	27.05	27.74	14.74	17.75	15.37	16.81	22.05	21.13
	24	45.02	41.83	44.16	26.47	27.97	28.71	15.49	17.58	17.93	17.24	22.95	21.57
	36	41.93	37.77	40.21	26.81	28.13	29.26	16.87	18.11	18.42	19.59	23.15	23.31
	48	35.39	31.19	38.33	28.81	29.08	29.85	17.25	19.45	20.66	20.55	23.85	24.04
Paris	12	49.25	50.18	50.11	24.11	18.93	19.48	17.47	10.31	9.93	19.50	11.39	10.05
	24	41.62	41.61	44.72	25.73	20.56	21.45	18.04	11.54	10.34	19.59	12.34	11.78
	36	37.92	36.42	41.48	26.55	21.63	22.82	19.37	13.45	12.05	21.75	14.26	13.54
	48	35.65	35.45	39.04	27.23	23.39	24.08	21.56	15.82	14.04	22.54	17.86	16.76

This indicates the distortion introduced relative to the YUV version of the video sequences under test. The value for simple XOR encryption is relatively high, confirming the visual impression of Fig. 15. Comparison of all SE using AES (with PSNR at 10.87 dB) and SLEPX (delivering 11.19 dB), shows that SLEPX comes close to effect of the encryption standard AES-CFB. The reader can judge from Table 8, that the effect is similar across the QPs for all three video sequences under test, across not just the Y component but for the two chrominance components.

SSIM index values of different video sequences encrypted by XOR, AES_CFB and SLEPX are shown in Fig. 16. The Figure shows that the video sequences CABAC-based encrypted by AES-CFB and SLEPX have lower SSIM index values than the XOR cipher. (Lower SSIM index values indicate better content protection.) Moreover, it is also again evident that SLEPX delivers similar confidentiality compared to AES-CFB. Hence, both objective (PSNR) and HVS-based (SSIM) results indicate that SLEPX renders almost as much protection as encryption of the SE syntax elements as when that is performed by AES-CFB.

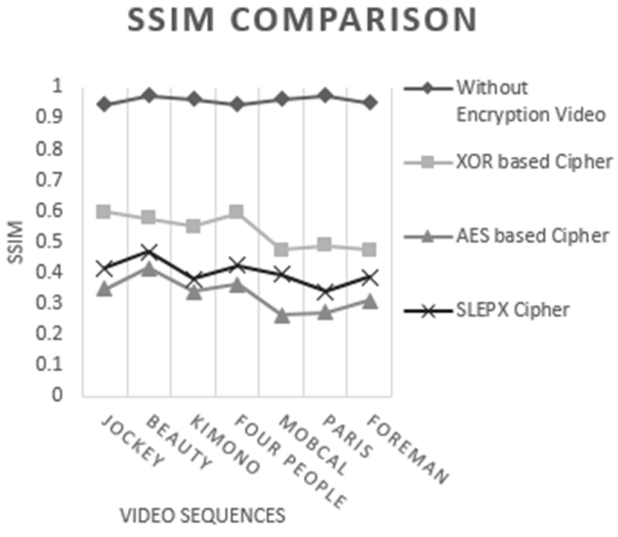


FIGURE 16. Comparison of SSIM values for different video sequences after SE making use of XOR, AES-CFB and SLEPX ciphers at QP = 18.

2) COMPUTATIONAL COST ANALYSIS

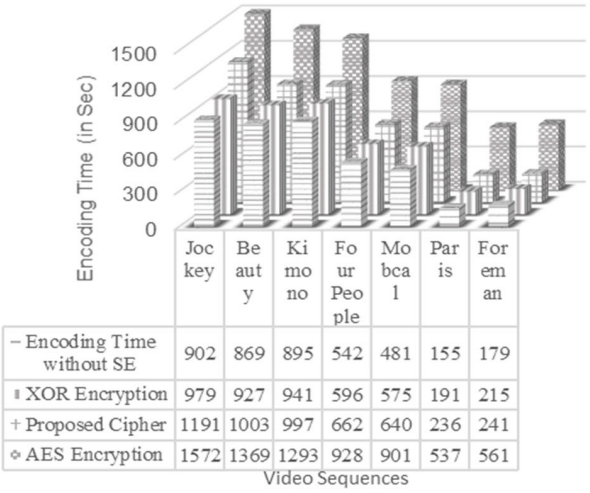
In a real-time processing environment, the duration in time of encryption also plays a vital role in assessing the efficiency of the ciphers. To assess that efficiency, comparisons were made of the encoding times of XOR and AES-CFB ciphers with those of SLEPX. Fig. 17 showing the results of absolute encoding times for the usual MV-based and texture-based syntax elements after encryption of SLEPX with respect to the XOR and AES-CFB ciphers. The Figure establishes that the encryption of the Kimono video sequence with SLEPX consumes 997 s, which is little bit more than with simple XOR (941 s) but significantly lower than when using AES-CFB (1293 s). In fact, overall timing results demonstrate that SLEPX is approximately equivalent to XOR-based encryption (with respect to computational time, though having superior content protection). SLEPX is clearly faster than AES, while the results from the previous Section demonstrate a similar level of content protection.

F. CRYPTANALYSIS

This Section considers the possibilities for several common attacks against SLEPX.

1) DIFFERENTIAL ATTACK

A differential attack is particularly critical for any encryption algorithm. On the other hand, the resilience to key-based



	Jockey	Beauty	Kimono	Four People	Mobcal	Paris	Foreman
Encoding Time without SE	902	869	895	542	481	155	179
XOR Encryption	979	927	941	596	575	191	215
Proposed Cipher	1191	1003	997	662	640	236	241
AES Encryption	1572	1369	1293	928	901	537	561

FIGURE 17. Comparison of encoding time of SLEPX with respect to XOR and AES-CFB encryption when performing SE.

sensitivity attacks is demonstrated if a minor modification to the encryption key yields entirely dissimilar encrypted video frames [75]. The testing scenario against this attack assumes that there is an original video frame called *P* and it is encrypted using a key called *K1*. If an attacker successfully decrypts *P* by utilizing another key, say *K2* then this shows that the key has no worth in the cipher. Assume that the keys (*K1* and *K2*) are slightly different in the sense that only one bit (the most significant bit) is dissimilar in *K1* and *K2*. Fig. 18 shows the result of encrypted a test video frame by utilizing *K1* and decrypting it not by K1 but by K2 from Fig. 18, it seems likely that a distorted frame will be the result of decrypting with a different key, however similar that key is to the correct key.

2) BRUTE FORCE ATTACK

A Brute-Force attack or exhaustive search attack tries out every possible key until a plausible plaintext is found. The key space size of a key is 2^n, where n is the number of bits in the key, which herein is 128 bits. Assuming that Moore's law of an effective doubling of computing speed every two years continues to hold and quantum computing does not replace conventional computing for this type of problem then it will be approximately 100 years before this size of key can be broken within one second. Therefore, encryption with SLEPX and other 128-bit keys are effectively immune to this type of attack [76], which constitutes approximately 5% of all successful attacks.

3) KNOWN AND CHOSEN PLAINTEXT ATTACK

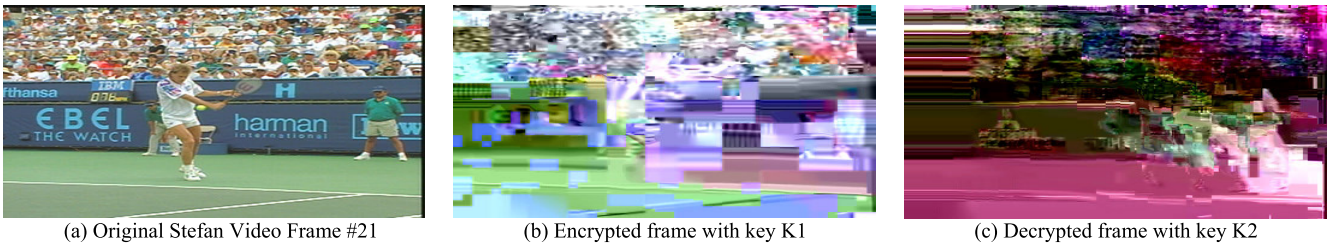
A Known Plaintext Attack (KPA) can take place if an attacker has in their possession both the encrypted data and the original video frames used to generate the encrypted data. However, even if an attacker is able to identify the video frames encrypted from their distorted versions, which seems unlikely, see Fig. 15 a4-b4, they would have the problem of identifying from the ciphertext, which of the elements had actually been encrypted. That is because (see Table 6 and its commentary) the encrypted elements, as a result of the compression process, are randomly scattered within the ciphertext in a close approximation to a Poisson distribution. In addition, applications of SLEPX are mainly real-time ones in which the video data could well lose its practical significance before another key, through effective key management, is applied to a fresh streamed video. Thus, unlike situations in which a KPA has been successful, such as attacks against the pkzip method of 'encryption' in which unencrypted stored files could be processed using the same encryption key, that situation is unlikely in video streaming scenarios. Besides, unlike a simple cipher such as pure XORing in which the key is simply XORed with the plaintext to produce the ciphertext, there is no simple relationship between the key and the ciphertext. Instead the plaintext is first compressed and in practice the key for each SHVC layer (see Fig. 8) will be different.

A Chosen Plaintext Attack (CPA) is a subset of the Known Plaintext Attack. However, a Chosen Plaintext Attack additionally requires access to a video server or other video generating device. It is possible then that an artificial video could be constructed with (say) all blank video frames. However, even if no tampering checks were in place, SLEPX involves the same encryption processes as AES (see Section IV.C) in the sense that confusion and diffusion take place. Like AES it also has a non-linear step, in SLEPX's case through XORing and, importantly, an IV is used for every stream. In fact, it is the last two features that effectively thwart any PTA or CPA by making the relationship between (compressed) plaintext and ciphertext non-linear and by restricting the amount of plaintext available to an attacker before the stream configuration changes again.

G. COMPARATIVE ANALYSIS WITH PRIOR RESEARCH

To evaluate SLEPX relative to previous research, this Section contains a comparison with other CABAC-based SE schemes. Table 9 summarizes the comparisons, with the basis of the comparisons given in the note beneath the Table. In [46], not all syntax elements work with SE but the balance between the growth in confidentiality relative to the growth in bitrate for similar video quality is analyzed. The HEVC-based encryption scheme discussed in [46] does not follow the constant bit-rate property but is considered robust to transcoding.

The solution discussed in [51], offers an HEVC-based solution that uses the AES cipher. However, the scheme has not been applied to the layered architecture of SHVC and the use of the AES cipher increases the computational cost considerably. Hence, though CBR encoding, especially two-pass encoding, can be slower than VBR compression, it is preferable in situations in which bandwidth is constrained, such as in wireless sensor networks or during TV channel multiplexing. In [58], the authors applied the RC6 cipher for encryption over HEVC binstrings in comparison with multiple modes of AES. RC6 can be used as an alternative of AES, but still



(a) Original Stefan Video Frame #21 (b) Encrypted frame with key K1 (c) Decrypted frame with key K2

FIGURE 18. Test based on key sensitivity analysis for Stefan frame #21. (a) Showing the original encoded Stefan frame. (b) Encrypted frame after utilizing the key K1; (c) Decryption performed by utilizing dissimilar key namely K2.

TABLE 9. Evaluation of SLEPX compared to recent SE-based practices for video codecs.

SE Schemes	Preserves video format	Cipher	Domain	Codec	Supports Scalability	Computational Cost
[46] presented in 2013	Yes	AES as stream cipher	Transform	HEVC	No	High
[51] presented in 2014	Yes	AES (Block Cipher)	Binstrings	HEVC	No	High
[58] presented in 2018	Yes	RC6	Binstrings	HEVC	No	Moderate
[77] presented in 2017	Yes	XOR with addition and Chaos	Binstrings	HEVC & SHVC	Yes	Moderate
Proposed SLEPX	Yes	Rounds of Permutation and XOR (Stream Cipher)	Binstrings	HEVC & SHVC	Yes	Low

RC6 is computationally expensive (with suggested at least 12 rounds of operation of RC5) than the proposed SLEPX. The authors in [77] works with basic encryption (XOR with addition) and also applied chaos for output. Chaos is computationally expensive when applied in conjunction with XOR encryption, consequently much degrades visual quality.

Generally, the SLEPX structure integrates a multiple rounds of permutations and XOR cipher for use with SE applied to CABAC binstrings within SHVC layered encoding. Alternatively, recent techniques of encryption having some weaknesses in terms of efficiency, bitrate overhead and computational complexity and may be fulfilled by electing weak encryption parameters.

H. DISCUSSION

The SLEPX cipher is a robust approach that works to secure the confidentiality of the scalable version of HEVC. From the results, SLEPX has shown itself to be a suitable alternative form of encryption, compared to time-consuming full encryption of either the complete stream or, especially herein, of selected components of the compressed visual data, whether that video stream is transmitted or held in data storage. The effectiveness of the cipher was judged in this Section by its response to different requirements. The visual quality of SLEPX encrypted video frames has been examined in terms of objective and perceptual video-quality metrics, demonstrating that ciphered frames have significantly distorted appearances. Experiments also demonstrated that SLEPX has a competitive encryption rate, while offering a moderate to high level of confidentiality. Thus, the cipher compared well with state-of-the-art AES, in that SLEPX approaches the security of AES but consumes less computational resources, compared to a rudimentary (and insecure) XOR cipher. In addition, the SE algorithm has the potential to be tunable by trading off some parameters so as to extend to a greater range of applications. For example, for devices with very limited capability then some of the CABAC syntax element types to be encrypted might be neglected. This would reduce the confidentiality of the SE but would also reduce the SLEPX encryption task, thus reducing the computational burden on a low capability device (refer back to Table 6). Equally, for those applications for which security guarantees are important it may be possible to augment the types of CABAC syntax elements encrypted. As the trade-offs in that respect are likely to be complex, this aspect of the research is reserved for future work.

The need for video encryption continues to grow, especially for real-time applications such as video surveillance, as was indicated in Section I's initial discussion of video encryption applications. Such surveillance is close to becoming a necessity [78], not just within obvious venues such as airports and railways but within many private houses and business premises, especially in countries where political unrest exists. Developments such as classification of objects within video frames and the intelligent tracking of those objects thereafter has increased the desirability of such surveillance. Additionally, cheap, high resolution cameras with small form factors also add to the deployment of video surveillance. On the other hand, without encryption, alteration of video frames has become relatively easy because

of the widespread dissemination of image-processing software. In addition, legal constraints of privacy protection for streamed video are being put in place across the world, such as the European Union's General Data Protection Regulation (GDPR) [79], which mandates the employment of encryption, rather than other measures, in all except the simplest of video streaming contexts.

VI. CONCLUSION

This paper proposed an innovative cipher, SLEPX, for lightweight selective encryption over the scalable extension of HEVC, namely SHVC. Based on the results of Section V, SLEPX offers a robust, efficient, and format-compliant video content protection for compressed SHVC bit-streams. To avoid any impact on encoder/decoder synchronization SE is applied at the final entropy coding stage of SHVC with careful choice of encrypted syntax elements. Thus, the proposed privacy protection solution works by selecting significant syntax elements i.e. MVD signs, as well as the NZ-TC signs and absolute value of the NZ-TC suffixes from the CABAC coder at each SHVC layer. Comparison of SLEPX with other ciphers, i.e. industry-standard AES at one end of the block-encryption spectrum and simple XOR at the other end, was made on different grounds (objective video quality, subjective video quality and computation time, which impacts upon overall communication and storage latency). That investigation revealed that the SLEPX cipher offers a sufficient (moderate to high) level of protection, which is approximately as secure as AES, while the computational cost is almost as minimal as results from XORing. Linear regression testing was applied on the counts of the selected syntax elements from CABAC, the SHVC entropy coder, showing that the selected syntax elements are sparsely distributed within a video frame and among all frames. In fact, because of this scattering, video distortion is widespread across video sequences. SLEPX works with SE both for single-layer HEVC and the multi-layer version of HEVC (SHVC) with all three scalabilities (Fig. 1 & Section IV-D). The cipher is based on the concept of a joint crypto-transcoder, which means that encrypted bitstreams may be transrated at a later time without revealing and compromising the original contents of single-layer encoders.

REFERENCES

[1] T. Barnett, S. Jain, U. Andra, and T. Khurana, "Cisco visual networking index (VNI): Complete forecast update, 2017–2022," APJC Cisco Knowl. Netw. Presentation, Cisco Syst., San Jose, CA, USA, Dec. 2018.

[2] G. J. Sullivan, J. M. Boyce, Y. Chen, J.-R. Ohm, C. A. Segall, and A. Vetro, "Standardized extensions of high efficiency video coding (HEVC)," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 6, pp. 1001–1016, Dec. 2013, doi: 10.1109/JSTSP.2013.2283657.

[3] M. Ghanbari, M. Fleury, and E. Khan, "Future performance of video codecs," Office Commun., London, U.K., Res. Rep., Nov., 2006.

[4] M. Ghanbari, *Standard Codecs: Image Compression to Advanced Video Coding*. London, U.K.: The Institute of Engineering and Technology, 2003.

[5] J. Bankoski, M. Frost, and A. Grange, "The Internet needs a competitive, royalty-free video codec," *APSIPA Trans. Signal Inf. Process.*, vol. 6, pp. 1–7, Aug. 2017, doi: 10.1017/ATSIP.2017.14.

[6] J. M. Boyce, Y. Ye, J. Chen, and A. K. Ramasubramonian, "Overview of SHVC: Scalable extensions of the high efficiency video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 1, pp. 20–34, Jan. 2016, doi: 10.1109/TCSVT.2015.2461951.

[7] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012, doi: 10.1109/TCSVT.2012.2221191.

[8] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1103–1120, Sep. 2007, doi: 10.1109/TCSVT.2007.905532.

[9] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, Jul. 2003, doi: 10.1109/TCSVT.2003.815165.

[10] G. Correa, P. A. Assuncao, L. V. Agostini, and L. A. da Silva Cruz, "Fast HEVC encoding decisions using data mining," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 4, pp. 660–673, Apr. 2015, doi: 10.1109/TCSVT.2014.2363753.

[11] J. Vanne, M. Viitanen, and T. D. Hämäläinen, "Efficient mode decision schemes for HEVC inter prediction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 9, pp. 1579–1593, Sep. 2014.

[12] S. D. Kamble, N. V. Thakur, and P. R. Bajaj, "Fractal coding based video compression using weighted finite automata," *Int. J. Ambient Comput. Intell.*, vol. 9, no. 1, pp. 115–133, Jan. 2018, doi: 10.4018/IJACI.2018010107.

[13] D. S. Jat, L. C. Bishnoi, and S. Nambahu, "An intelligent wireless QoS technology for big data video delivery in WLAN," *Int. J. Ambient Comput. Intell.*, vol. 9, no. 4, pp. 1–14, Oct. 2018, doi: 10.4018/IJACI.2018100101.

[14] N. Bouchemal, R. Maamri, and M. Chihoub, "Securing ambient agents groups by using verification, judgment and surveillance," *Int. J. Ambient Comput. Intell.*, vol. 5, no. 3, pp. 44–60, Jul. 2013, doi: 10.4018/ijaci.2013070104.

[15] R. Pantos and W. May. (2011). *HTTP Live Streaming. Internet Draft*. [Online]. Available: https://tools.ietf.org/html/draft-pantos-http-live-streaming-23

[16] *Announcing the Advanced Encryption Standard (AES)*, FIPs Standard 197, Federal Information Processing Standards Publication, 2001, p. 3, vol. 197, nos. 1–51, doi: 10.6028/NIST.FIPS.197.

[17] D. Dorwin, J. Smith, and M. Watson, "Encrypted media extensions. World wide Web consortium, recommendation," REC-encrypted-media-20170918, Sep. 2017.

[18] A. Popov, *Prohibiting RC4 Cipher Suites*, document RFC 7465, Internet Engineering Task Force, 2015, doi: 10.17487/RFC7465.

[19] L. S. Choon, A. Samsudin, and R. Budiarto, "Lightweight and cost-effective MPEG video encryption," in *Proc. Int. Conf. Inf. Commun. Technol.*, Damascus, Syria, 2004, pp. 525–526, doi: 10.1109/ICTTA.2004.1307863.

[20] A. Massoudi, F. Lefebvre, C. De Vleeschouwer, B. Macq, and J.-J. Quisquater, "Overview on selective encryption of image and video: Challenges and perspectives," *EURASIP J. Inf. Secur.*, vol. 2008, Dec. 2008, Art. no. 179290, doi: 10.1155/2008/179290.

[21] A. Boho, G. Van Wallendael, A. Dooms, J. De Cock, G. Braeckman, P. Schelkens, B. Preneel, and R. Van de Walle, "End-to-end security for video distribution: The combination of encryption, watermarking, and video adaptation," *IEEE Signal Process. Mag.*, vol. 30, no. 2, pp. 97–107, Mar. 2013, doi: 10.1109/MSP.2012.2230220.

[22] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker, *Digital Watermarking and Steganography*. Amsterdam, The Netherlands: Morgen Kaufmann, 2008, doi: 10.1016/B978-012372585-1.50007-3.

[23] S. M. Oliveira, M. A. Nascimento, and O. R. Zaïane, "Digital watermarking: Status, limitations and prospects," Dept. Comput. Sci., Univ. Ontario, Edmonton, AB, Canada, Tech. Rep. TR. 02-01, 2002.

[24] N. M. Thomas, D. Lefol, D. R. Bull, and D. Redmill, "A novel secure H.264 transcoder using selective encryption," in *Proc. IEEE Int. Conf. Image Process.*, San Antonio, TX, USA, Sep./Oct. 2007, pp. IV-85–IV-88, doi: 10.1109/ICIP.2007.4379960.

[25] J. Chen, J. Boyce, and Y. Ye, *High Efficiency Video Coding (HEVC) Scalable Extension Draft 7*, document JCTVC-R1008_v7, Sapporo, Japan, Jun./Jul. 2014, doi: 10.1109/TCSVT.2015.2461951.

[26] M. Kim, H. Lee, and S. Sull, "Efficient transform domain transcoding: Intra frame of H.264/AVC to JPEG," *IEEE Trans. Consum. Electron.*, vol. 57, no. 3, pp. 1362–1369, Aug. 2011, doi: 10.1109/TCE.2011.6018895.

[27] M. Seufert, S. Egger, M. Slanina, T. Zinner, T. Hobfeld, and P. Tran-Gia, "A survey on quality of experience of HTTP adaptive streaming," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 1, pp. 469–492, 1st Quart., 2015, doi: 10.1109/COMST.2014.2360940.
[28] A. Eleftheriadis, M. R. Civanlar, and O. Shapiro, "Multipoint videoconferencing with scalable video coding," *J. Zhejiang Univ.-Sci. A*, vol. 7, no. 5, pp. 696–705, May 2006, doi: 10.1631/jzus.2006.A0696.
[29] C. Yuan, B. B. Zhu, Y. Wang, S. Li, and Y. Zhong, "Efficient and fully scalable encryption for MPEG-4 FGS," in *Proc. Int. Symp. Circuits Syst. (ISCAS)*, Bangkok, Thailand, vol. 2, 2003, pp. 620–623, doi: 10.1109/ISCAS.2003.1206050.
[30] J.-R. Ohm, G. J. Sullivan, H. Schwarz, T. K. Tan, and T. Wiegand, "Comparison of the coding efficiency of video coding standards—Including high efficiency video coding (HEVC)," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1669–1684, Dec. 2012, doi: 10.1109/TCSVT.2012.2221192.
[31] V. Sze and M. Budagavi, "High throughput CABAC entropy coding in HEVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1778–1791, Dec. 2012, doi: 10.1109/TCSVT.2012.2221526.
[32] A. Bogdanov and T. Isobe, "How secure is AES under leakage," in *Advances in Cryptology*, vol. 9453, T. Iwata and J. Cheon, Eds. Berlin, Germany: Springer, 2015, pp. 361–385, doi: 10.1007/978-3-662-48800-3_15.
[33] A. S. Butter, J. M. Kaczmarczyk, and A. Y. Ngai, "Scalable MPEG2 compliant video encoder," U.S. Patent 5 768 537, Aug. 16, 1998. [Online]. Available: https://patents.google.com/patent/US5768537A/en
[34] W. K.-H. Ho, W.-K. Cheuk, and D. P.-K. Lun, "Content-based scalable H.263 video coding for road traffic monitoring," *IEEE Trans. Multimedia*, vol. 7, no. 4, pp. 615–623, Aug. 2005, doi: 10.1109/TMM.2005.850959.
[35] W. Li, "Overview of fine granularity scalability in MPEG-4 video standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 3, pp. 301–317, Mar. 2001, doi: 10.1109/76.911157.
[36] H. K. Arachchi, X. Perramon, S. Dogan, and A. M. Kondoz, "Adaptation–aware encryption of scalable H.264/AVC video for content security," *Signal Process., Image Commun.*, vol. 24, no. 6, pp. 468–483, Jul. 2009, doi: 10.1016/j.image.2009.02.004.
[37] M. Asghar, M. Ghanbari, M. Fleury, and M. Reed, "Efficient selective encryption with H.264/SVC CABAC bin-strings," in *Proc. 19th IEEE Int. Conf. Image Process.*, Orlando, FL, USA, Sep. 2012, pp. 2645–2648, doi: 10.1109/ICIP.2012.6467442.
[38] M.-C. Tsai and T.-S. Chang, "High performance context adaptive variable length coding encoder for MPEG-4 AVC/H.264 video coding," in *Proc. IEEE Asia Pacific Conf. Circuits Syst. (APCCAS)*, Singapore, Dec. 2006, pp. 586–589, doi: 10.1109/APCCAS.2006.342056.
[39] J. Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockhammer, and T. Wedi, "Video coding with H.264/AVC: Tools, performance, and complexity," *IEEE Circuits Syst. Mag.*, vol. 4, no. 1, pp. 7–28, 2004, doi: 10.1109/MCAS.2004.1286980.
[40] T. Taleb, A. Ksentini, and R. Jantti, "Anything as a service' for 5G mobile systems," *IEEE Netw.*, vol. 30, no. 6, pp. 84–91, Nov. 2016, doi: 10.1109/MNET.2016.1500244RP.
[41] S. Bellovin and R. Housley, "Guidelines for cryptographic key management," in *Proc. Symp. Res. Sec. Privacy*, 2005, pp. 1–7.
[42] T. Stütz and A. Uhl, "Format-compliant encryption of H.264/AVC and SVC," in *Proc. 10th IEEE Int. Symp. Multimedia*, Berkeley, CA, USA, Dec. 2008, pp. 446–451, doi: 10.1109/ISM.2008.52.
[43] M. Yang, L. Zhuo, J. Zhang, and X. Li, "An efficient format compliant video encryption scheme for HEVC bitstream," in *Proc. IEEE Int. Conf. Prog. Informat. Comput. (PIC)*, Nanjing, China, Dec. 2015, pp. 374–378, doi: 10.1109/PIC.2015.7489872.
[44] B. Boyadjis, M.-E. Perrin, C. Bergeron, and S. Lecomte, "A real-time ciphering transcoder for H.264 and HEVC streams," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Paris, France, Oct. 2014, pp. 3432–3434, doi: 10.1109/ICIP.2014.7025697.
[45] B. Boyadjis, C. Bergeron, B. Pesquet-Popescu, and F. Dufaux, "Extended selective encryption of H.264/AVC (CABAC)- and HEVC-encoded video streams," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 4, pp. 892–906, Apr. 2017, doi: 10.1109/TCSVT.2015.2511879.
[46] G. Van Wallendael, A. Boho, J. De Cock, A. Munteanu, and R. Van de Walle, "Encryption for high efficiency video coding with video adaptation capabilities," *IEEE Trans. Consum. Electron.*, vol. 59, no. 3, pp. 634–642, Aug. 2013, doi: 10.1109/TCE.2013.6626250.
[47] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004, doi: 10.1109/TIP.2003.819861.
[48] M. Farajallah, W. Hamidouche, O. Deforges, and S. E. Assad, "ROI encryption for the HEVC coded video contents," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Quebec City, QC, Canada, Sep. 2015, pp. 3096–3100, doi: 10.1109/ICIP.2015.7351373.
[49] D. Socek, H. Kalva, S. S. Magliveras, O. Marques, D. Culibrk, and B. Furht, "A permutation-based correlation-preserving encryption method for digital videos," in *Proc. Int. Conf. Image Anal. Recognit. (ICIAR)*, 2006, pp. 547–558, doi: 10.1007/11867586_51.
[50] J. Zhou, X. Liu, O. C. Au, and Y. Y. Tang, "Designing an efficient image encryption-then-compression system via prediction error clustering and random permutation," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 1, pp. 39–50, Jan. 2014, doi: 10.1109/TIFS.2013.2291625.
[51] Z. Shahid and W. Puech, "Visual protection of HEVC video by selective encryption of CABAC binstrings," *IEEE Trans. Multimedia*, vol. 16, no. 1, pp. 24–36, Jan. 2014, doi: 10.1109/TMM.2013.2281029.
[52] H. Hofbauer, A. Uhl, and A. Unterweger, "Transparent encryption for HEVC using bit-stream-based selective coefficient sign encryption," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Florence, Italy, May 2014, pp. 1986–1990, doi: 10.1109/ICASSP.2014.6853946.
[53] S. Al Salami, J. Baek, K. Salah, and E. Damiani, "Lightweight encryption for smart home," in *Proc. 11th Int. Conf. Availability, Rel. Secur. (ARES)*, Salzburg, Austria, Aug./Sep. 2016, pp. 382–388.
[54] X. Yao, Z. Chen, and Y. Tian, "A lightweight attribute-based encryption scheme for the Internet of Things," *Future Gener. Comput. Syst.*, vol. 49, pp. 104–112, Aug. 2015.
[55] F. Dufaux and T. Ebrahimi, "Scrambling for privacy protection in video surveillance systems," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 8, pp. 1168–1174, Aug. 2008.
[56] R. A. Shah, M. N. Asghar, S. Abdullah, M. Fleury, and N. Gohar, "Effectiveness of crypto-transcoding for H.264/AVC and HEVC video bit-streams," *Multimedia Tools Appl.*, vol. 78, no. 15, pp. 21455–21484, Aug. 2019, doi: 10.1007/s11042-019-7451-5.
[57] C. E. Shannon, "Communication theory of secrecy systems," *Bell Syst. Tech. J.*, vol. 28, no. 4, pp. 656–715, Oct. 1949, doi: 10.1002/j.1538-7305.1949.tb00928.x.
[58] A. I. Sallam, O. S. Faragallah, and E.-S.-M. El-Rabaie, "HEVC selective encryption using RC6 block cipher technique," *IEEE Trans. Multimedia*, vol. 20, no. 7, pp. 1636–1644, Jul. 2018, doi: 10.1109/TMM.2017.2777470.
[59] *SHM-7.0*. Accessed: Oct. 30, 2015. [Online]. Available: https://hevc.hhi.fraunhofer.de/svn/svn_SHVCSoftware/tags/SHM-7.0
[60] W. Hamidouche, M. Raulet, and O. Déforges, "4K real-time and parallel software video decoder for multilayer HEVC extensions," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 1, pp. 169–180, Jan. 2016, doi: 10.1109/TCSVT.2015.2478705.
[61] V. Seregin and Y. He, *Common SHM Test Conditions and Software Reference Configurations*, document JCTVC-O1009, Geneva, Switzerland, 2013.
[62] *Video Sequences*. Accessed: Aug. 8, 2020. [Online]. Available: https://media.xiph.org/video/derf/
[63] *UHD Video Sequences*. Accessed: Aug. 8, 2020. [Online]. Available: http://ultravideo.cs.tut.fi/#testsequences
[64] Q. Huynh-Thu and M. Ghanbari, "The accuracy of PSNR in predicting video quality for different video scenes and frame rates," *Telecommun. Syst.*, vol. 49, no. 1, pp. 35–48, Jan. 2012, doi: 10.1007/s11235-010-9351-x.
[65] O. S. Faragallah and A. Afifi, "Optical color image cryptosystem using chaotic baker mapping based-double random phase encoding," *Opt. Quantum Electron.*, vol. 49, no. 3, pp. 1–28, Mar. 2017, doi: 10.1007/s11082-017-0909-7.
[66] N. Taneja, B. Raman, and I. Gupta, "Selective image encryption in fractional wavelet domain," *AEU-Int. J. Electron. Commun.*, vol. 65, no. 4, pp. 338–344, Apr. 2011, doi: 10.1016/j.aeue.2010.04.011.
[67] N. Taneja, B. Raman, and I. Gupta, "Chaos based partial encryption of spiht compressed images," *Int. J. Wavelets, Multiresolution Inf. Process.*, vol. 9, no. 2, pp. 317–331, Mar. 2011, doi: 10.1142/S0219691311004092.
[68] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 2nd ed. Upper Saddle River, NJ, USA: Prentice-Hall, 2002.
[69] X. Yang and X. Jiang, "A hybrid active contour model based on new edge-stop functions for image segmentation," *Int. J. Ambient Comput. Intell.*, vol. 11, no. 1, pp. 87–98, Jan. 2020, doi: 10.4018/IJACI.2020010105.

[70] A. K. Jain, *Fundamentals of Digital Image Processing*. Upper Saddle River, NJ, USA: Prentice-Hall, 1988.
[71] A. Jawad and A. Fawad, "Efficiency analysis and security evaluation of image encryption schemes," *Int. J. Video Image Process. Netw. Secur.*, vol. 12, no. 4, pp. 18–31, 2012.
[72] A. Kaur, L. Kaur, and S. Gupta, "Image recognition using coefficient of correlation and structural SIMilarity index in uncontrolled environment," *Int. J. Comput. Appl.*, vol. 59, no. 5, pp. 32–39, Dec. 2012, doi: 10.5120/9546-3999.
[73] M. N. Asghar, M. Ghanbari, M. Fleury, and M. J. Reed, "Confidentiality of a selectively encrypted H.264 coded video bit-stream," *J. Vis. Commun. Image Represent.*, vol. 25, no. 2, pp. 487–498, Feb. 2014, doi: 10.1016/j.jvcir.2013.12.015.
[74] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*. Boac Raton, FL, USA: CRC Press, 1996.
[75] N. K. Pareek, V. Patidar, and K. K. Sud, "Diffusion–substitution based gray image encryption scheme," *Digit. Signal Process.*, vol. 23, no. 3, pp. 894–901, May 2013, doi: 10.1016/j.dsp.2013.01.005.
[76] L. Gutteridge. *What's the Deal With Encryption Strength is 128 Bit Encryption Enough or do you Need More?* [Online]. Available: https://medium.com/@drgutteridge/whats-the-deal-with-encryption-strength-is-128-bit-encryption-enough-or-do-you-need-more-3338b53f1e3d
[77] W. Hamidouche, M. Farajallah, N. Sidaty, S. E. Assad, and O. Deforges, "Real-time selective video encryption based on the chaos system in scalable HEVC extension," *Signal Process., Image Commun.*, vol. 58, pp. 73–86, Oct. 2017, doi: 10.1016/j.image.2017.06.007.
[78] N. Kanwal, M. N. Asghar, M. S. Ansari, M. Fleury, B. Lee, M. Herbst, and Y. Qiao, "Preserving chain-of-evidence in surveillance videos for authentication and trust-enabled sharing," *IEEE Access*, vol. 8, pp. 153413–153424, 2020, doi: 10.1109/ACCESS.2020.3016211.
[79] M. N. Asghar, N. Kanwal, B. Lee, M. Fleury, M. Herbst, and Y. Qiao, "Visual surveillance within the EU general data protection regulation: A technology perspective," *IEEE Access*, vol. 7, pp. 111709–111726, 2019, doi: 10.1109/ACCESS.2019.2934226.

RIZWAN ALI SHAH received the master's degree in computer science from the Virtual University of Pakistan and the M.Phil. degree in computer science from the Department of Computer Science & IT (DCS&IT), The Islamia University of Bahawalpur (IUB), Pakistan, where he is currently pursuing the Ph.D. degree. He is an active Research Member of the Multimedia Research Group, DCS & IT, IUB. He is working as a Computer Instructor in Federal Government Educational Institutes (Cantt. /Garrison). He has more than 10 years of teaching and R&D experience. His research interests are the security aspects of multimedia (audio and video), video codecs, compression, encryption, secure transcoding, and secure multimedia transmission.

MAMOONA N. ASGHAR received the Ph.D. degree from the School of Computer Science and Electronic Engineering, University of Essex, Colchester, U.K., in 2013. She is currently a Marie Sklodowska-Curie (MSC) Career-Fit Research Fellow with the Software Research Institute, Athlone Institute of Technology (AIT), Ireland, since June 2018. As an MSC Principal Investigator (PI), her research targets the proposals and implementation of technological solutions for General Data Protection Regulation (GDPR) compliant CCTV Surveillance Systems. She is also a regular Faculty Member with the Department of Computer Science and Information Technology (DCS&IT), The Islamia University of Bahawalpur, Punjab, Pakistan, where she is currently on postdoc leave. She has more than 15 years of teaching and R&D experience. She has published several ISI indexed journal articles along with numerous international conference papers. She is also actively involved in reviewing for renowned journals and conferences. Her research interests include security aspects of multimedia (image, audio and video), video encoders, compression, visual privacy, encryption, steganography, secure transmission in future networks, video quality metrics, and key management schemes.

SAIMA ABDULLAH received the Ph.D. degree from the Department of Computer Science and Electronic Engineering, University of Essex, U.K. She is currently working as an Assistant Professor with the Department of Computer Science and Information Technology, The Islamia University of Bahawalpur, Pakistan. Her main research interests include wireless networks/communications, future Internet technology, and network performance analysis. She has published around 10 articles in the above research areas. She serves as a reviewer for international journals. She is a member of the Multimedia Research Group in DCS & IT, and has been working on efficient and secure communication of multimedia data over future generation network technologies.

NADIA KANWAL (Senior Member, IEEE) received the M.Sc. and Ph.D. degrees in computer science from the University of Essex, Essex, U.K., in 2009 and 2013, respectively. She is currently a Marie Sklodowska-Curie Career-Fit Postdoctoral Fellow with the Software Research Institute, Athlone Institute of Technology (AIT), Ireland. The primary objective of this fellowship is to propose technological solutions for privacy protection of humans as per GDPR guidelines. She is associated with the Lahore College for Women University, Pakistan, where she is also an Associate Professor and is also on leave to pursue Postdoctoral Fellowship. She is applying deep learning methods to improve the performance of vision algorithms for detection and matching tasks which can help to develop robust solutions for different vision-related applications. Her research interests include machine learning, image/video processing, medical imaging, and privacy. She remained as a Student Member of the IEEE Computer Society, the Institution of Engineering and Technology, and the British Machine Vision Association. She has been actively involved in reviewing for reputed conferences and journals.

MARTIN FLEURY (Member, IEEE) received the degree in modern history from Oxford University, U.K., the degree in maths/physics from the Open University, Milton Keynes, U.K., the M.Sc. degree in astrophysics from QMW College, University of London, U.K., in 1990, the M.Sc. degree in parallel computing systems from the University of South-West England, Bristol, in 1991, and the Ph.D. degree in parallel image-processing systems from the University of Essex, Colchester, U.K. He was a Senior Lecturer with the University of Essex, after which he became a Visiting Fellow. He is currently associated with the School of Engineering, Arts, Science, Technology and Engineering (EAST), University of Suffolk, Ipswich, U.K. He is also a Free-Lance Consultant. He has authored or coauthored around 295 articles and book chapters on topics such as document and image compression algorithms, performance prediction of parallel systems, software engineering, reconfigurable hardware, and vision systems. He has also published or edited books on high performance computing for image processing and peer-to-peer streaming. His current research interests include video communication over wireless networks and multimedia network security.