Confidentiality of a Selectively Encrypted H.264 Coded Video Bit-stream

Mamoona N. Asghar[1], Mohammed Ghanbari, Martin Fleury, and Martin J. Reed

University of Essex, School of CSEE, Colchester CO4 3SQ, U.K.

Abstract

It is an assumption that selective encryption does not strongly protect confidentiality owing to the partial visibility of some video data. This is because, though encryption keys may be difficult to derive, an enhanced version of selectively encrypted video sequence *might* be found from knowledge of the unencrypted parts of the sequence. An efficient selective encryption method for syntax elements of H.264 encoded video was recently proposed at the entropy coding stage of an H.264 encoder. Using this recent scheme as an example, the purpose of this paper is a comprehensive cryptanalysis of selectively encrypted H.264 bit-streams to *contradict* the previous assumption that selective encryption is vulnerable. The novel cryptanalysis methods presented in this paper analyze the ability of an attacker to improve the quality of the encrypted video stream to make it watchable. The conclusion is drawn that if the syntax elements for selective encryption are chosen using statistical and structural characteristics of the video, then the selective encryption method is secure. The cryptanalysis is performed by taking into account the probability distribution of syntax elements within the video sequence, the relationship of syntax elements with linear regression analysis and the probability of successfully attacking them in order to enhance the visual quality. The results demonstrate the preservation of distorted video quality even after considering many possible attacks on: the whole video sequence; each video frame; and on small video segments known as slices.

Keywords: cryptanalysis, entropy coding, perceptual attacks, selective encryption

[1] Email: masghaa@essex.ac.uk, tweety.mees@yahoo.com
Contact no. : 00447909391129

1. Introduction

Multimedia content confidentiality during network transmission is an important issue, due to the need to preserve a commercial advantage by not allowing access to a high-quality version of the original. Video, in particular, is normally compressed as a video bit-stream before transmission (or storage), as otherwise the bandwidth (and memory) demands could be prohibitive. Many algorithms have been proposed for video content protection, including selective encryption [1] [2] and watermarking [3], though the latter does not provide confidentiality. Selective encryption (SE) is often required for real-time streaming [4], as otherwise any delay involved in performing full encryption, especially on mobile devices, may prevent a continuous display at the receiver. With the move to High Definition (HD) video and video on mobile devices, the delay and computational cost of full encryption may become prohibitive. There are also issues of synchronization for full encryption if two-way, interactive video streaming takes place.

In fact, because video streams are normally compressed there are many opportunities to encrypt selected syntax elements of the video bit-stream, though which elements are selected must be carefully considered. For example, the size of the bit-stream should not be increased and the statistical properties of the original bit-stream should be preserved [5]. This has led to a focus on the out-of-the-predictive-loop stage of video compression [6], namely entropy coding; this paper contains cryptanalysis of one such SE scheme [7] [8]. SE is applied to the codeword and bin-strings (see Section 2.2) of the two forms of H.264 entropy coding Context Adaptive Variable Length Coding (CAVLC) [9] and Context Adaptive Binary Arithmetic Coding (CABAC) [10] present in the H.264 Advance Video Coding (AVC) [6] (as well as its extension Scalable Video Coding (SVC) [11]). Compared to other methods of SE, the encryption of entropy coding syntax elements can be made fully format compliant, and compression friendly, as well as providing good perceptual security to the video. However, the purpose of this paper is the cryptanalysis of such an SE scheme and *not* the features of this particular scheme itself. Indeed the features of the scheme itself have already been considered in [8].

Cryptanalysis [12] is the science of breaking and analyzing secure data and judging the efficacy of such cryptosystems against cryptographic attacks. In modern cryptosystems, the security of a cryptographic system is

not only measured in terms of the time and resources to recover the plaintext but also the strength of SE is analyzed in terms of improvement in visual quality. However, traditional cryptanalysis concentrates on the difficulty of extracting encryption keys. Based on an open description of the cipher, an attacker will either test the effect of a candidate key upon a known plaintext or consider using a candidate key on a captured ciphertext. Therefore, the resources expended are the (often) exhaustive application of candidate keys. However, in SE a well-known strong cipher may be used and, therefore, the principal weakness is not attacks upon the cipher key(s) but use of partially visible or unencrypted parts of the compressed bit-stream. An attacker may also have available a statistical model of the media source. Whereas in the traditional form of cryptanalysis, a zero-distortion decrypted version of the original is aimed at, the strength of an SE scheme should be judged [13] by the perceptual quality of the video after an attack. Therefore, the priority of the present paper is to analyse the probability of a successful attack based on access to the selectively encrypted version. This paper proposes a unique methodology for investigating the resilience of selectively encrypted video to attacks aimed at improving the video quality without access to any encryption keys. As mentioned in [13], SE poses a different problem to that of traditional cryptanalysis. In the latter, it is 'only' necessary to establish that access to secret keys is prohibited, whereas there are many different ways that SE can take place. Thus, cryptanalysis of SE must establish how difficult it would be to gain access to a video of sufficient quality and whether the effort would be worthwhile in any practical sense. Although, the specific analysis very much depends on how SE is carried out, the analysis techniques considered in this paper are transferable, with modification to the specific parameters used by a particular SE technique. Because the analysis is parameter-specific, this explains why, in this paper an SE method recently published by the authors [8] is used to illustrate the cryptanalysis technique, as the parameters used are naturally well-known to the authors. We have, therefore, avoided applying the cryptanalysis techniques to somebody else's algorithm, as the risk is that the finer details and parameters of that other algorithm might be misunderstood.

This paper is a primary research contribution in the cryptanalysis field by proposing the novel methods aimed at analyzing the distribution of syntax elements within the video sequence and the probability of attacks on the encrypted syntax elements. Experimental results are presented on the: whole video sequence; the video frame;

and the video slice basis of H.264/AVC. Before beginning that task this Section now briefly reviews recent related work in the field of SE, along with cryptanalysis.

1.1 Recent research on H.264 selective encryption and its cryptanalysis

To enhance the security of H.264 coded video streams, many video content encryption methods have been proposed [5] [14] [15]. In [16], the Network Abstraction Layer (NAL) units of H.264/AVC and SVC were individually encrypted, in order to have no side-effect upon format compliancy. By setting the NAL unit type of encrypted NALs to be outside the defined range, the decoder is forced to reject them, unless decryption is enabled. The work in [17] was an earlier scheme for SE of scalable video. It identified the entropy coder as most suitable for SE, as it has least side-effects. The base-layer receives more attention than enhancement layers. The intra prediction modes (if an intra frame), Motion Vector Difference (MVD) values and texture sign bits are all selectively encrypted in the base layer. However, for the enhancement layers the encrypted bits are determined by the form of scalability (spatial/SNR or temporal). SE of scalable video suffers from the need to transmit separate keys for each layer. In [18], the keys are embedded as watermarks output. However, the video layers are created through wavelet coding prior to compression using H.264, rather than by the standard-compatible SVC extension to H.264. SE is achieved by scrambling the wavelet scan order of the intra quantized transform coefficients. In [19], a low-quality free preview application was developed by performing transparent encryption on the H.264/SVC layers, resulting in non-format compliant enhancement layers. The algorithm encrypts the scalable enhancement layers, while leaving the base layer in plain format. On the other hand, there is a conviction in some quarters that if the base layer is protected then no one can get the data from the enhancement layers and the whole SVC bit-stream is secured. Although it is a useful technique to reduce the computational cost, research shows [20] that if objects are encrypted in this way their content can be easily guessed without decryption. Consequently, Algin et al. [21] proposed the idea of SE on SVC with three security levels. The idea of [21] involves the encryption of signs of coefficients, signs of motion vectors and the alteration of DC values. Sign encryption has no effect on the bit-rate and compression efficiency (as the signs are equally likely) but DC value alterations can change the video statistics and affect the compression efficiency, which consequently increases

the bit-rate. That is to say the bit-rate overhead increases in [21], while the video quality remains the same. Another promising technique applied in [22] to non-scalable H.264/AVC also uses SE on the codewords/bin-strings of the entropy coder. The research in [22] bears comparison with that of the authors. In preliminary work [7], the authors of this paper identified those syntax elements in H.264/AVC CABAC bin-strings that could disrupt video statistics including video motion and texture information for all frame types i.e. I, P and B frames, while preserving the bit-rate and format compliance of the SVC layers. In a further conference paper, the authors of this paper in [8] went on to apply the same approach to CAVLC codewords.

Returning to comparing the work in [8] with that in [22], in [22] the motion data are not considered and only the codewords or bin-strings (depending on the type of entropy coding) of residual data are selectively encrypted for I and P frames, while the results are not analyzed on B frames. Therefore, because of the links between the approach in [22] and that of [8], the cryptanalysis in the current contribution is relevant to both these state-of-the-art advances in SE. Work continues in this field as [23] illustrates for non-scalable H.264/AVC. In [23], a single tuneable control factor allowed the extent of encryption to be traded-off against the level of security. The components that are encrypted are similar to those of [17], except instead of MVDs, the sign of the MVDs is encrypted. The authors also analysed the SE method of [22] and found that only encrypting the subsuffix of the suffix of the non-zero transform coefficients without the sign-bits reduces the perceptual scrambling effect, while encrypting just the sign bits has a similar effect but reduces the computational cost. Cryptanalysis of a replacement attack is briefly considered in [23]. A replacement attack replaces encrypted components by constant values. However, the authors of [23] were unable to detect any change to the objective video quality after applying their replacement scheme.

In [5], there is already an extensive recent survey of encryption methods for H.264/SVC, which includes a tabulated comparison of those methods. The main subject of our paper is not our method but is actually a way of performing cryptoanalyis of SE methods,. However, to increase the utility of this paper, Table 1 is a summary of the main points for and against the methods described in the previous discussion.

Table 1: Summary of recent research on H.264 SE methods

SE method	**Pros**	**Cons**
[5][14][15][20]	- Survey the field	Do not make a direct contribution
[7][8]	- Uses codewords or bin-strings for sufficient encryption — decoder format compliant (permitting transcoding and other compression-domain processing), compression efficient (bandwidth and storage conservation).	Some loss in compression speed due to wider selection of encrypted syntax elements.
[16]	Modifies NAL headers rather than perform a full encryption of complete bitstream — format compliant as encrypted NALs are rejected at a decoder.	Not transcodable and no compression-domain processing, as video payload is conventionally encrypted. Will increase bandwidth as payload is encrypted after compression rather than within the compression process. Reduces error resilience.
[17]	- Uses CAVLC features for base layer — decoder format compliant, compression efficient at base layer	- Enhancement layers are less secure than the base layer, implying a perceptual attack using enhancement layer data may be possible; uses separate keys for each SVC layer implying organizational complexity.
[18]	- Compression efficient, includes keys as watermarks reducing organizational complexity,	- Not H.264/SVC format compliant as enhancement layers are wavelet encoded — requires proprietary codec at end devices.
[19]	- Unencrypted base layer as preview (transparent encryption) but conventionally encrypted enhancement layers.	- The enhancement layers are not decoder format compliant and increase the bitrate. - Error resilience is reduced.
[21]	- Encrypts selected syntax elements, including sign bits.	Encryption of values of integer DCT coefficients results in an increase in bitrate
[22]	- Uses codewords or bin-strings — decoder format compliant, compression efficient	- Does not use MVDs or B-frames — possibly vulnerable to guessing attack (see [23]), allowing reconstruction of sufficient quality.
[23]	-Provides some control of visual scrambling effect and showed that subsuffix encryption did not enhance security	-Requires full control to guard against replacement attack.

Cryptanalysis is an important field, which analyses the confidentiality of encrypted data. However, in recent studies of video-coded bit-stream encryption not much attention has been paid to cryptanalysis. The previously mentioned methods for the SE of H.264 video standard bit-streams do not provide a method to practically analyse their strength during cryptanalysis, and this is generally true of the literature on SE of H.264 bit-streams. A

limited amount of work on cryptanalysis is found in the literature on early video standards which is now briefly discussed. In [24] a cryptanalysis was applied on several SE schemes for MPEG-1/2 coded video (MPEG 1 and 2 are early hybrid video codecs [25]). The first set of SE algorithms [26] was designed to enable re-encryption of an already encrypted video stream with a new key, if the original key had been compromised. Unfortunately, this property was shown to provide no defence when both cipher-texts were known. Besides the encryption algorithm applied to transform coefficients was known to be vulnerable to known-plaintext cipher-only attacks. Cryptanalysis of a second scheme [27], involving permutation of the entropy coding coefficients, was also performed. Unfortunately, by a heuristic assumption, a successful attack succeeded by brute force exploration of the reduced permutation space.

In [28] there is a further examination of SE algorithms for MPEG compressed video streams, which concludes that adequate security is provided for the risks encountered. An interesting feature of that survey is that, at that time, SE of entropy-coded bits was not considered. However, the goal of selecting syntax elements that are statistically independent was defined, leading to the present interest in entropy coding elements. Of course, as perfect compression has yet to be accomplished, no codec removes all predictability. The work in [28] also examines the possibility of a perceptual attack in which corrupted parts of a video frame image are replaced by trial replacements. Such an attack can result in an acceptable or watchable video. In [29], the security of encrypted entropy coding other than through SE of entropy coded syntax elements was considered. That form of encryption was applied to the coding tables used with entropy coding. However, in [29] a number of such algorithms are shown to be vulnerable to known-plaintext attacks.

The remainder of this paper is organized as follows. Section 2 outlines the context of this work on SE for H.264/AVC coded video along with cryptanalysis techniques. Section 3 briefly discusses the input data for the analysis, before examining the possible guessing attacks on the encrypted syntax elements in Section 4. Concluding remarks are made in Section 5.

2. Context

This Section supplies essential background information needed to understand the rest of the paper.

2.1 H.264 video coding standard

The contemporary H.264, otherwise known as MPEG-4 part 10 [6] has been standardized by the International Telecommunications Union (ITU-T) Video Coding Experts Group and the ISO/IEC Motion Picture Experts Group (MPEG) [30]. H.264/AVC provides significant improvement in compression efficiency of up to 50% over a broad range of bit rates and video resolutions compared to earlier standards. H.264 is a hybrid video codec in that various algorithms are involved in de-correlating the video data and removing various forms of redundancy. For the purposes of this paper, attention is focused on the final output stage, entropy coding, which, as mentioned in Section 1, can take one of two forms, CAVLC, or CABAC.

H.264/SVC [11] permits devices to send and receive multi-layered bit-streams. H.264/AVC and SVC use the same entropy coding modalities: CAVLC, with lower computational requirements, and CABAC which can result in a 5-15% reduction in bit-rate over the CAVLC. Thus, the analysis in this paper is relevant to both variants of H.264. Whether CAVLC or CABAC is employed, the codewords or bin-strings are the subject of selective encryption, upon which cryptanalysis takes place in this paper.

2.2 Selective Encryption of Syntax Elements

Naïve encryption (NE) is a full video content encryption encompassing the media header and payload. NE normally uses a block cipher such as Advanced Encryption Standard (AES) as part of a Digital Rights Management (DRM) [32]. NE is applied [5] because many content providers assume that without full encryption their content will be vulnerable. This is why they are willing to sacrifice bitrate efficiency in the interests of total security. SE, also called sufficient encryption [32], encrypts the partial, most influential, video data to make the video viewable but not watchable. As SE retains format compliance, it allows potential viewers to preview a distorted video stream to tempt viewers into paying for subscription. SE also does not require a set-top to be adapted or replaced to view these distorted video streams [33]. Thus, SE has received considerable attention from researchers [5] but has not met with a similar take-up in commerce compared to NE. Further cryptanalysis of SE is one way to remedy that situation.

This paper now considers the scheme which is used as a case study for the cryptanalysis that follows. As previously emphasized, the objective of this paper is the cryptanalysis of an SE scheme that the authors have ready access to and not the re-presentation of a scheme already presented in [8]. Thus, to fulfill the cryptanalysis pre-requisites, the selected codeword and bin-strings, chosen for SE over CAVLC and CABAC encoded bit-streams respectively, are outlined here.

CAVLC syntax elements for SE are:

i) Signs of MVD in *se(p)* mapping,

ii) Signs of Trailing Ones (T1s), and

iii) *UEG0* suffix of non-zero (NZ) levels including signs of the NZ levels.

CABAC syntax elements for SE are:

i) *UEG3* suffix (signs of MVD),

ii) *UEG0* suffix, and

iii) Signs of the NZ- Transform Coefficients (TC) levels.

The SE is implemented on an H.264 bit-stream by using the Advanced Encryption Standard in Cipher Feedback mode (AES-CFB) with a 128-bit key length. The impact of SE upon codewords or bin-strings (depending on which entropy coder is used) upon a *Foreman* video sequence with quantization parameter (QP) set to a low value (higher video quality) of 18 can be seen in Figure 1. Two objective quality measures, i.e. Peak Signal-to-Noise Ratio (PSNR) (dB) [34] and Structural Similarity (SSIM) [35], were applied. SSIM, with a range from 0 (poor quality) to 1 (high quality), is said to more closely reflect human perception than PSNR. Therefore, SSIM is an objective measure of the likely subjective response. The results of the given SE after decoding are illustrated, where the raw video YUV values include color weightings. Figure 1 shows that, though the original subject can be guessed at, if the rest of the frames are like frame 84, the video sequence is certainly unwatchable.

(a) (b) (c)

Figure 1: SE of frame 84 of *Foreman* for CABAC and CAVLC (a) Original video frame 84-QP18 PSNR = {Y=40.4, U=45.2, V=46.6} dB, SSIM = 0.981, (b) Selective encryption (CAVLC) PSNR = {Y=8.3, U=24.0, V=20.5} dB, SSIM = 0.225, (c) Selective encryption (CABAC) PSNR = {Y=8.4, U=20.6, V=24.1} dB, SSIM = 0.219.

2.3 Security Considerations

In modern cryptography, the security of selectively encrypted multimedia content is considered in terms of confidentiality and preventing improvements to the visual quality. For commercial applications, preventing improvements to the video quality of an encrypted video takes priority over privacy [5]. In other words, it is more important to prevent any improvements to video quality than it is to maintain the complete privacy of the content. The two security criteria (Confidentiality and Quality resilience) are dependent on various factors which include:

1) Selection of a cipher algorithm;
2) Strength of an encryption key;
3) Importance of selected encryption parameters for the visual quality/perception of an encoded video; and
4) Distribution of selected encryption parameters within the video.

Items 1 and 2 above relate to the confidentiality of the video, while items 3 and 4 relate to video quality resilience.

The treatment of video in the case of SE is quite different from the treatment afforded to a still image. Video represents the motion of objects and/or camera zooms, pans, and so on. Hence, to provide quality resilience, attention to the encryption of the MVD is valuable (item 3 above), as it can restrict any kind of motion within the video. Motion restriction alone is not enough to provide good visual security to the video. Therefore, other parameters such as the signs of transform coefficients (TC) and suffixes are considered in order to disrupt the

luminance and chrominance statistics of the video, consequently, making it un-watchable. Regarding items 1 and 2, AES with an 128-bit key for encryption is an excellent option, as it is estimated that the time required for breaking a 128 bit key by applying all possible keys at 50 billion keys/s takes 5×10^{21} years [36]. Item 4 above is significant for the cryptanalysis of a given SE; as if the selected syntax elements are widely distributed and are in large number then it is indeed impossible to guess them all correctly to significantly enhance the visual quality. Details of item 4 are described in Section 3.

In the following Sections, there is a comprehensive analysis of the security of the case study SE scheme for the CAVLC and CABAC syntax elements.

3. Video Data Analysis

For the cryptanalysis, twenty different Common Intermediate Format (CIF) (352 × 288 pixels/frame) video clips were used to find the number of occurrences of each chosen syntax element within different sizes of video sequences. These well-known video sequences are commonly employed to test the effectiveness of compression algorithms within the video coding community. The test video clips were encoded with [24] a QP value of 18 (from an H.264 range from 0–51), a Group-of-Pictures (GOP) size and Intra-refresh period equal to 16. The counts of chosen syntax elements were extracted from the decoder immediately after de-compression of an H.264 bit-stream. The statistical results of H.264 encoded video were taken with the H.264/AVC reference software JM 18.3 version encoder [37].

Table 2 shows the counts of MVDs, suffixes and Non-Zero (NZ)-TCs for both CAVLC and CABAC. CAVLC and CABAC each has different patterns of encoding for an H.264 bit-stream. CAVLC processes the NZ-TCs in the form of Trailing Ones (T1s) and non-zero levels. Consequently, the value of the total number of non-zero transform coefficients in CAVLC is the sum of T1s and the NZ-levels. For example, the total number of NZ-TCs in the *Foreman* sequence encoded with CAVLC is 1356179+1169855 = 2526034, which is larger in number than the number of coefficients encoded in CABAC, which is 2187867 (shown in Table 2). The CAVLC encoded bit-streams are around 14% larger in number of NZ-TC's than CABAC, owing to the poorer compression efficiency of the dynamic Huffman variable-length-coding (VLC) used by CAVLC. (CAVLC

selects from a number of pre-formed look-up tables rather than the adaptively formed tables used by CABAC.) The number of MVDs is dependent on the motion within sequences (motion of objects and/or camera) and is nearly equal for both CAVLC and CABAC. In CABAC, suffixes are used according to a set of rules [10] but in CAVLC suffixes are used for every NZ-TC. Therefore, suffix counts in CAVLC are equal to the total number of T1s and NZ-levels. Suffix lengths are variable for both CABAC and CAVLC and, consequently, only their counts and not their lengths are reported in Table 2.

4. Video Cryptanalysis

Cryptanalysis [14] is the investigation of the weaknesses of encrypting systems that might enable retrieval of secret information as well as improve the visual quality of image or video in the case of SE. By keeping in mind the security requirements of commercial applications, novel cryptanalysis methods are proposed in this paper to analyze the strength of SE in the case of reconstruction of encrypted video with enhanced visual quality.

The possibility of attacking the SE through a guessing attack to reconstruct the original video quality is examined through a statistical analysis of the data given in Table 2. Any attacker has access to the bit-stream sent to a decoder. Therefore, the signs of the non-zero TCs and MVDs are not a secret and are known to everyone. The sign can be a positive or negative sign. It might be claimed that it is easy to guess the signs and make the video watchable. To consider this aspect we have performed an analysis of the SE used by determining what the probability of guessing sufficient numbers of any particular syntax element is to enhance the video quality up to the level of being watchable. Moreover, the cryptanalysis mechanism is based on the following scenarios that are elaborated in the following Sections of this paper:

1) The distribution probability of each selected syntax element throughout the video.
2) Determine what the probability is of successful guessing attacks on: slices; frames; and the whole video clip.
3) Determine the strength of the overall SE scheme against the attacks.

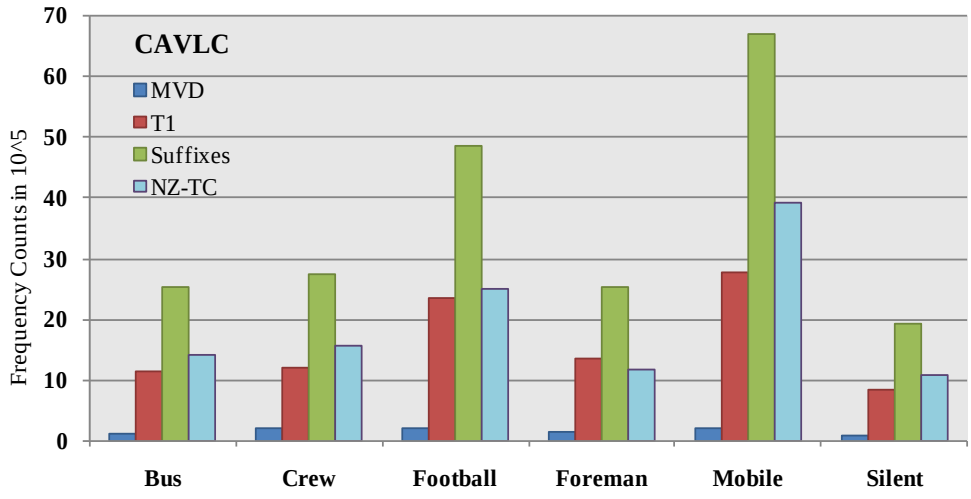
Table 2: Frequency counts for test videos for selected syntax elements.

Sequence (CIF)	File Size (MB)	Number of encoded frames	CAVLC				CABAC		
			Signs of MVD	T1s	Suffixes	Signs of NZ-level	Signs of MVD	Suffixes	Signs of NZ-TC
Bus	21.7	150	119904	1121693	2540974	1419281	120526	24343	2301381
City	43.5	300	110490	1040270	3212042	2171772	110899	9740	1928953
Coastguard	43.5	300	137225	2075792	4680750	2604958	137668	18398	4374389
Container	43.5	300	30235	705826	1509011	803185	30618	22502	1357796
Crew	43.5	300	194532	1186750	2732956	1546206	195575	5415	3289294
Flower	36.3	250	203606	1849957	5211348	3361391	206532	86015	4839292
Football	37.7	260	194247	2360672	4861193	2500521	194587	24688	4234493
Foreman	43.5	300	131377	1356179	2526034	1169855	132151	8898	2187867
Hall	43.5	300	44461	1739684	2416053	676369	44778	11762	1777465
Harbor	43.5	300	196237	2676961	5353922	2676961	198473	27687	5523826
ICE	34.8	240	117761	634997	1204046	569049	118537	7648	992637
Mobile	43.5	300	194344	2777287	6694372	3917085	196352	102930	6057747
Mother	43.5	300	58605	598521	952695	354174	58929	1090	741001
News	43.5	300	54545	617443	1265783	648340	54459	19675	1082922
Paris	43.5	300	103587	941295	2593375	1652080	104612	42368	2366256
Silent	43.5	300	73830	830202	1918376	1088174	74113	5924	1722413
Soccer	43.5	300	164730	1541657	3115842	1574185	164138	10426	2737290
Stefan	13.1	90	57310	798684	1642859	844175	57902	18508	1474421
Tempete	37.7	260	173200	2085299	4794428	2709129	174220	37823	4357297
Waterfall	37.7	260	80102	922264	2120420	1198156	80583	9520	1895015

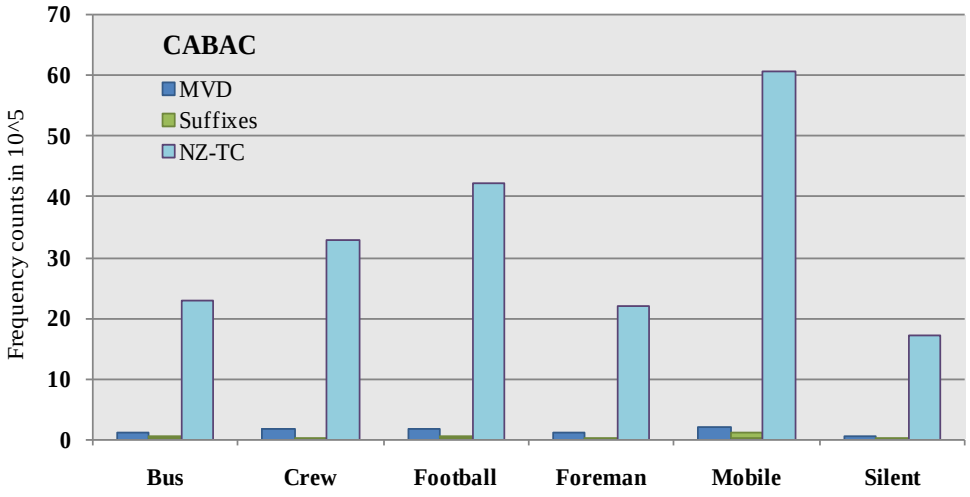
4.1 Distribution of Syntax Elements

This Section elaborates item 4 of Section 2.3 by determining the distribution of each syntax element within a compressed video file. The extracted data from twelve different sequences with the same file size of 43.5 MB (Table 2) was the source for this distribution analysis. The Poisson distribution served to represent the probability distribution of the numbers of each syntax element within the tested sequences. Of alternative candidate discrete distributions, the Binomial is unsuitable, because the counts are *not* closely bunched around their means. The negative Binomial distribution is also unsuitable as to be applied it requires the range of the count sizes to be unbounded. Therefore, we selected the Poisson distribution as the most likely distribution for the random data, as, through the nature of the encoding process, the range of the count sizes is not unbounded. Figure 2 is a visual representation of the counts for an arbitrary selection of clips and syntax elements. Notice that the vertical axis is in units of 10000 counts. Though Figure 2 gives the reader an idea of the data involved in this research, as this paragraph outlines, the main reason why the Poisson distribution was selected in this research is the unsuitability of

the alternative candidate discrete distributions. Selecting a distribution in this sense means that there is reasonable expectation that the sample data selected (the syntax element counts) will match the distribution of the statistical population of such syntax element counts. Because the population consists of all available video sequences, it is impossible to show definitively that the Poisson distribution applies to this population.



(a)



(b)

Figure 2: Counts of syntax elements for five video clips using (a) CAVLC and (b) CABAC

The idea of the analysis is to show that large numbers of encrypted syntax elements can arise within a file with moderate probabilities. This is a first step in showing the difficulty of correctly guessing syntax elements or at least enough syntax elements to make a video clip watchable. The assumption is made that the values of the signs within a file are uniformly distributed, which is a common assumption, for example in [10]. For syntax elements with a continuous range of values, it is assumed that it would be very difficult to guess the correct value of the encrypted element, at least over a sufficient number of elements.

The number of non-zero MVDs is determined by the file size and the degree of motion within a video sequence. Notice that all non-zero MVDs have a sign associated with them. On the basis of these factors, the occurrence of non-zero MVDs within the video was analyzed with counts taken from the CAVLC and the CABAC encoded streams. The *Container* video sequence has around 30,000 MVDs (Table 2), so this video was not included in probability calculations (Table 3). The other eleven sequences with file size 43.5 MB were used for the calculations of MVD distribution in the video. In Table 3, the counts of the non-zero MVD syntax elements were arranged into a number of classes. For example, there is the class with a range (0.5-1) consisting of all those MVD counts with a value in the range 5,000 to 10,000 (as all values in Table 3 are in units of 10,000 syntax elements). The characteristic value for each class was conservatively estimated as the start value. For example, the start value for this class is 0.5 or a count with a value of 5,000. The column *Frequency* shows the frequency of occurrence of non-zero MVDs within that range. Thus, there are three counts in the range (0.5-1). The final column is the product of the other two columns e.g. the row 1 and column 3 of Table 3 has a value 0.5×3 = 1.5. These values enable the sample mean of the MVD values to be calculated as μ = 1.13

Using this value of μ in the well-known Poisson distribution ($P\ (\mu;\ n)\ =\ \frac{e^{-\mu}\mu^{n}}{n!}$) with $n = 3,$ gives in (1):

$$P(\mu;\ n) = 0\ .07752 \qquad (1)$$

which shows that there is a 7.75% probability that the total number of MVD signs will be as many as 30,000 while for n = 4 there would be a 2.19% probability that the MVD sign count will be around 40,000 signs distributed throughout the video sequence.

This analysis can be repeated for the TCs. The total number of TCs was found separately for CAVLC and CABAC encoded bit-streams. The total number of TCs for CAVLC is comprised of both T1s and non-zero levels, while the CABAC total is comprised of just non-zero levels. Equivalent results to Table 3 for MVD values are given in Table 4. This gives:

$$\mu = 2.75 \text{ (CAVLC-TC)} \quad (2)$$

$$\mu = 2.79 \text{ (CABAC-TC)} \quad (3)$$

which, by setting n=2, yields:

$$P(\mu; n) = 0.241 \text{ (CAVLC)} \quad (4)$$

$$P(\mu; n) = 0.239 \text{ (CABAC)} \quad (5)$$

This shows that for CAVLC and CABAC there is a 24.1% and 23.9% probability respectively that the total number of TC signs will be around two million.

The residual transform coefficients are encoded in two different ways in CAVLC: firstly, the number of T1s are encoded in the form of positive and negative signs (if there are more than three), while secondly the other coefficients are encoded as non-zero levels. Therefore, the number of T1s as well is required in CAVLC encoded video.

Table 3: Frequency of MVD values.

Class (x)	**Frequency (f)**	**Product (f.x)**
0.5 (0.5-1)	3	1.5
1 (1-1.5)	4	4
1.5 (1.5-2)	2	3
2 (2-2.5)	2	4

All counts are in units of 10,000 syntax elements

Table 4: Frequency of TC values.

CAVLC			**CABAC**		
Range (x)	**Freq. (f)**	**Product (f. x)**	**Range (x)**	**Freq. (f)**	**Product (f. x)**
1 (0-1)	3	3	0.5 (0.5-1.5)	1	1
2 (1-2)	4	8	1.5 (1.5-2.5)	4	6
3 (2-3)	2	6	2.5 (2.5-3.5)	3	7.5
4 (3-4)	0	0	3.5 (3.5-4.5)	1	3.5
5 (4-5)	2	10	4.5 (4.5-5.5)	1	4.5

6 (5-6)	1	6	5.5 (5.5-6.5)	2	11

All counts are in units of millions of syntax elements

Table 5: Frequency of T1 values.

Range (x)	**Frequency (f)**	**Product (f. x)**
0.5 (0.5-1)	5	2.5
1.0 (1-1.5)	3	3
1.5 (1.5-2)	1	1.5
2.0 (2-2.5)	1	2
2.5 (2.5-3)	2	5

All counts are in units of millions of syntax elements

Using Table 5 with n =1 gives

$$P(\mu; n) = 0.363 \text{ (CAVLC)} \quad (6)$$

Thus, there is a 36.3% probability that the number of trailing ones (T1s) will be around one million, while there is a 21% probability that the number of T1s will be around two million in the CAVLC encoded sequences.

Just as for the MVD and TC counts, the distribution of the number of suffixes in an entire video sequence was calculated. The suffix occurrences are found to be very different in CAVLC and CABAC. In CAVLC, the suffixes exist for every non-zero level of coefficient but in CABAC the suffixes occur when the condition abs_level > 14 holds. Therefore, the number of suffixes in CAVLC is much larger than in CABAC. CAVLC results in numbers of suffixes in the millions, while the CABAC numbers are only in the thousands.

Repeating the analysis showed that there is a 33% probability that the number of suffixes will be around one million in the video clip encoded with CAVLC and that there is a 22.4% probability that the number of suffixes will be around ten thousand in a CABAC encoded video sequence.

4.2 Linear regression of numbers of syntax elements

We also investigated the linear regression of every encrypted parameter against the independent variable 'file size' to find the relationship of numbers of encrypted parameters with the file size, as in (7):

$$y = \alpha + \beta x \quad (7)$$

where y is any dependent variable (an encrypted syntax element), x is the independent variable (file size) and α and β are the coefficient values calculated on the basis of the data. Linear regression testing was performed for all

variables by taking 95% confidence intervals for upper and lower ranges. The twenty test sequences from Table 2 with different file sizes were used in this analysis. The results are for MVD, non-zero TC, suffixes and trailing ones as dependent variable *y*. Table 6 shows sample linear regressions results for the numbers of MVD signs within the 20 video clips.

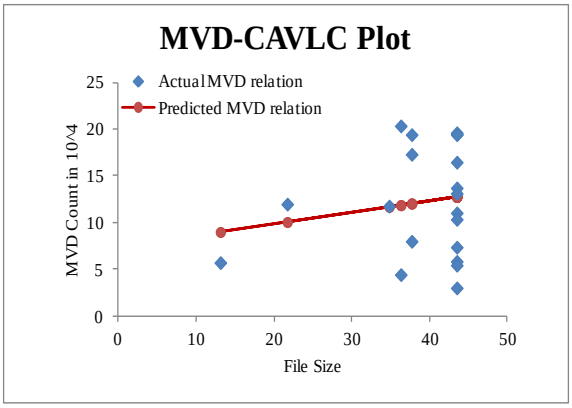
Table 6: Results of linear regression analysis for MVD signs.

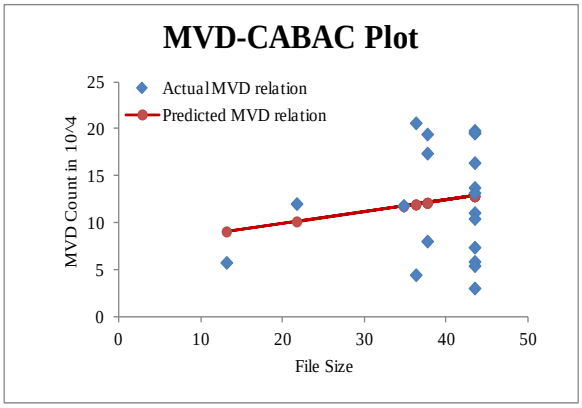
	Coefficients	**Values**	**Lower 95%**	**Upper 95%**
CAVLC	α	73874.47	-64987.8	212736.8
	β	1238.69	-2262.5	4739.9
CABAC	α	74681.34	-65340.6	214703.3
	β	1237.64	-2292.8	4768.1

Thus, the linear regression equation for the number of MVDs against video file sizes for CAVLC and CABAC are as follows:

$$y = 73874.47 + (1238.69)\,x \qquad (8)$$

$$y = 74681.34 + (1237.64)\,x \qquad (9)$$





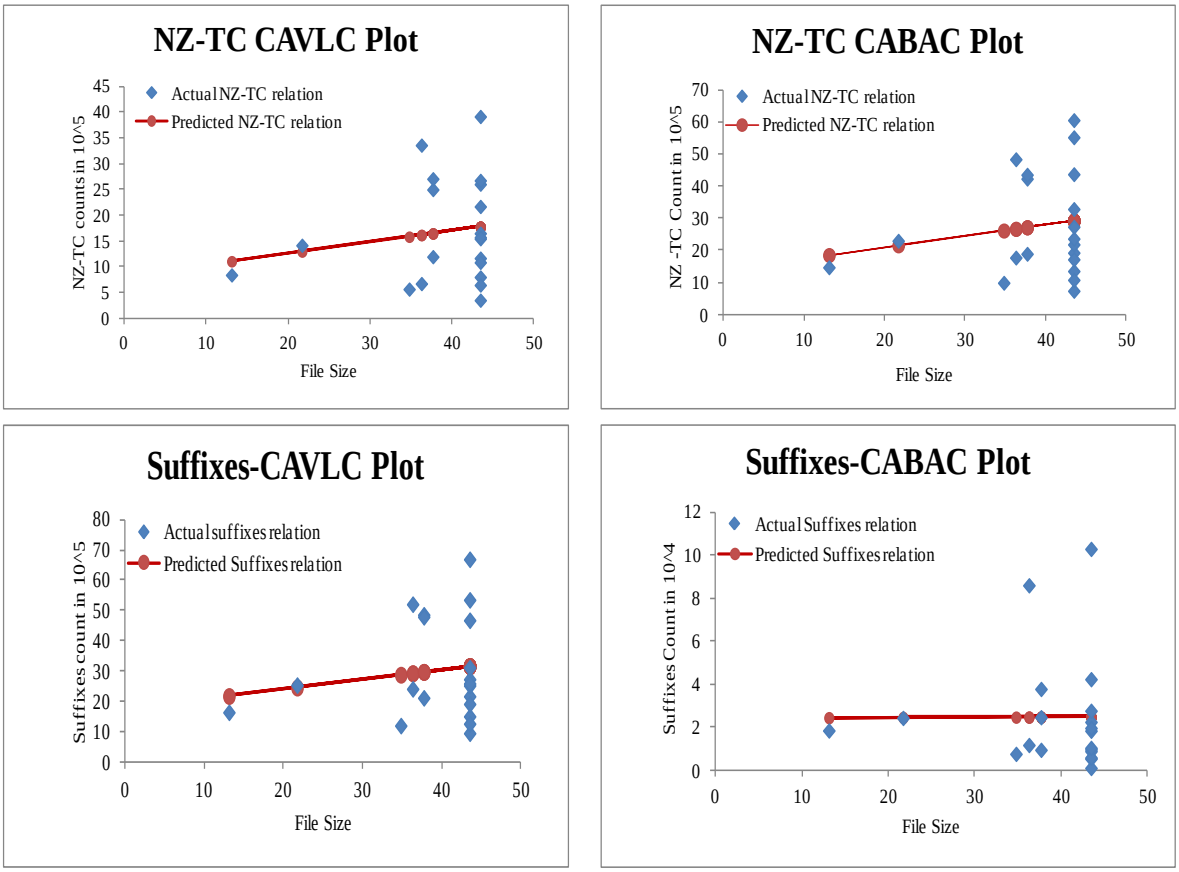


Figure 3: Graphical representation of linear regression analysis for encrypted CAVLC and CABAC syntax elements.

The results of the linear regression analysis for all the selected encrypted syntax elements are shown in Figure 3. It can be observed from all the plots that, as the file sizes become larger, there is little relationship between compressed video file size and the number of encrypted elements. The other conclusion that can be drawn is that the signs of MVD and TC counts are always a very large number and guessing them would be very difficult as a result.

4.3 Probability of Guessing

To find the probability of guessing the MVD and TC signs, we have examined video sequences with shorter durations and, hence, small file sizes. For example, the *Bus* sequence (150 frames, 5 second video clip) has the following characteristics:

Video file size = 21.7 MB

CAVLC-MVD signs count (n) = 119904

Assume that an attacker must be able to guess at least 80% correct MVD signs to be able to make the video watchable. This percentage can be reduced if the person watching the video is willing to accept a lower quality of video. The probability of guessing can be found from the standard formula for a combination:

$$^{n}C_{k} = \binom{n}{k} = \frac{n(n-1)\ldots(n-k+1)}{k!(n-k)!} \qquad (10)$$

If one can guess all the correct MVD signs with complete accuracy the total number of possibilities is 2^119904. Thus the probability of getting all 119904 correct is 1/(2^119904), which is obviously very small.

The probability of getting one correct MVD sign is found from putting the values of n and k into (11), where n denotes the non-zero MVD count and k denotes the guessing combinations:

$$^{n}C_{k} = {}^{119904}C_{1} = 119904 \qquad (11)$$

Guessing 80% non-zero MVDs correctly requires the following number of guessing combinations:

$$^{119904}C_{95923} = 119904 \times 119903 \times \ldots (119904 - 95923 + 1) / 95923! \qquad (12)$$

This results in a very small probability of guessing all 80% of the coefficients correctly. Moreover, these calculations involve very large numbers which are beyond the normal computational limit of a computer, unless a special large-number library is used.

Encoding at large QP values

The frequency counts of selected syntax elements given in Table 2, as previously remarked are taken with a QP value of 18 (from a QP range in H.264 from 0–51), which is a typical value for broadcast quality video. High quality video with a small QP value (finer quantization), results in a maximal number of DCT coefficients. However, when the video is encoded with larger QP values (coarser quantization), some of the DCT blocks are either coded with just a few coefficients, or none at all, and the corresponding macroblock may even be 'Skipped' (no residual data sent to the decoder, which uses an estimate of the MVs to directly select and insert a macroblock from a previous frame). Thus, in general the number of DCT coefficients per frame is greatly reduced and, hence, the number of their signs is also reduced, as noted in Table 7. Table 7 shows the number of NZ-TC signs of four

different videos for three different QP values, i.e. QP 8, 18 and 48; thirty frames are encoded in each test video. As the number of coded blocks (blocks with DCT coefficients) is reduced (through skipping), column number 4 of Table 7 ensures that the number of TC signs is greatly reduced for large QP value of 48 for both CABAC and CAVLC. Therefore, if the video is encoded with larger QP values, the probability of correctly guessing the signs of the MVDs/ TCs is increased. Although Table 7 confirms that the number of TC signs are still in thousands even for a short video of 30 frames, for further investigation let us assume, with the encoding of video at relatively large QP value, the number of MVD/TC signs has been considerably reduced to 100 or even just 50, as a result of coarser quantization and block skipping. The following calculations confirm the close-to-zero probabilities of guessing 80% correct from totals of 50 and 100 MVD/TC signs. The probability of guessing 80% correct out of 50 MVD/TC signs is 0.00001193, while the probability of guessing 80% correct for 100 MVD/TC signs = $3.98e^{-27}$ (almost zero). However, for watchable quality at least 80% of the MVD/TC signs must be correctly guessed. Therefore, with as few as 50 signs remaining, the probability of guessing sufficient of these is very low.

Table 7: NZ-TC signs count at different QP values.

Test Videos	**QP Values**	**No. of Encoded frames**	**NZ-TCs (CABAC) and NZ-levels (CAVLC) signs count**
Paris encoded with CAVLC	8	30	689622
	18	30	247545
	48	30	20948
Silent encoded with CAVLC	8	30	459480
	18	30	162572
	48	30	2356
City encoded with CABAC	8	30	909431
	18	30	227407
	48	30	2071
Mobile encoded with CABAC	8	30	1415332
	18	30	597840
	48	30	42224

For further clarity and the convenience of readers, visual results of two videos were taken at QP values 8 and 48 after an 80% successful guessing attack on the NZ-TC signs. The PSNR and SSIM values given in Figure 4 confirm that the visual quality of a short video having 30 frames (one second video clip) is below the watchable level if 80% TC signs are guessed correctly at the lowest QP value of 8 and the largest QP value at 48.

(a) [Y=42.8, U=49.0, V=49.7] dB
SSIM = 0.9955

(b) [Y=5.7, U=21.3, V=28.6] dB
SSIM = 0.0363

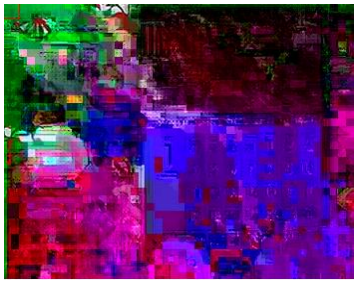
(c) [Y=16.4, U=26.2, V=31.2] dB
SSIM = 0.6539

(d) [Y=28.2, U=35.1, V=36.3] dB
SSIM = 0.7515

(e) [Y=6.1, U=19.5, V=21.2] dB
SSIM = 0.0955

(f) [Y=14.7, U=28.0, V=27.4] dB
SSIM = 0.6208

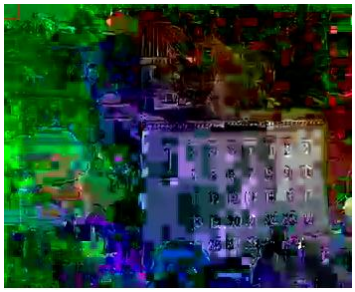
(g) [Y=39.5, U=45.0, V=44.3] dB
SSIM = 0.997



(h) [Y=6.3, U=12.9, V=11.3] dB
SSIM = 0.0742

(i) [Y=39.5, U=45.0, V=44.3] dB
SSIM = 0.997

(j) [Y=25.2, U=31.3, V=30.4] dB
SSIM = 0.790



(k) [Y=7.4, U=16.8, V=12.7] dB
SSIM = 0.1165

(l) [Y=11.2, U=20.7, V=18.4] dB
SSIM = 0.5282

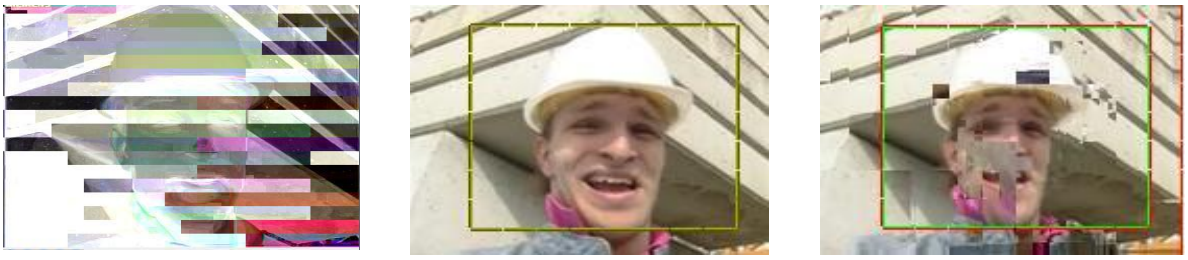
Figure 4: The effect of correct guessing up to 80% TC signs (a-f) *Silent* video (Frame # 13) encoded with CAVLC and (g-l) *Mobile* video (Frame # 16) encoded with CABAC. (a)(g) Test videos encoded at QP 8, (b)(h) Full TC signs encryption only with QP 8, (c)(i) 80% TC signs guessed correct at QP 8, (d)(j) Test videos encoded at QP 48, (e)(k) Full TC signs encryption only with QP 48, (f)(l) 80% TC signs guessed correct at QP 48.

Effect of slicing

The following calculations were performed to verify the strength of the encryption system if each video frame is encrypted as a number of small slices. In H.264, a single frame can be encoded in a number of independently decodable slices, thus reducing the impact of the loss of a slice during transmission. Each slice might have a minimum of 10 to 100 encrypted MVD or TC signs. In order to guess 80% of these correctly from 10 encrypted signs the analysis is as follows. The total number of possibilities is 2^{10} = 1024. There is just one way of guessing all 10 values correctly. To get 9 values correct one has 10/1 = 10 possibilities. To get 8 values correct, the possibilities are 10 × 9/2 = 45. Therefore, the total possibilities are 1+10+45 =56. Thus, the probability of guessing 80% correctly is 56/1024 = 0.054, i.e. 5.4%. Figure 5 (a) shows an encrypted frame of *Foreman* video encoded with slices of size 200 bytes with only TC signs encryption. It can be seen that encryption of just a few signs within a slice can still significantly obscure the video frame.

Effect of regions of interest

Regions of Interest (ROI) can be created using H.264's Flexible Macroblock Ordering (FMO) type 2 [38]. Consequently, encryption can be performed solely on a specific region. If one further delves down to the slice level then it is possible that the video might even be segmented into very small slices with at most ten MVD or TC signs or signs for both. (Small slices may have 10 motion vectors, and, hence, 10 MVD signs. However, these slices have plenty of TC blocks, where each block can have several DCT coefficients. Hence, the number of TC signs is much larger than 10, even well over several 100.) Then probability calculations show that there is a 5.4% probability of getting 80% correct signs, which is a low probability of getting barely watchable video sequence. As an illustration, Figure 5 (b) and (c) show an ROI before and after encryption. For the sake of simplicity, only MVD signs are encrypted in an ROI.

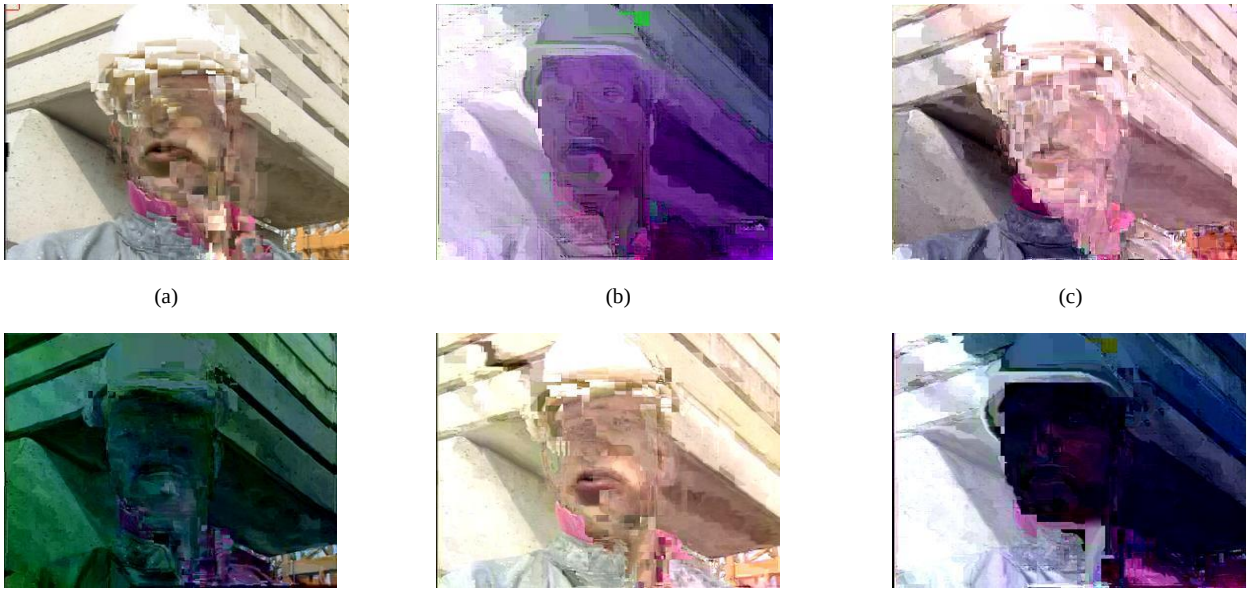


(a) (b) (c)

Figure 5: *Foreman* frame (a) Example encryption using TC signs of small (200 B) slices (b) With ROI, (c) With MVD signs encryption of ROI.

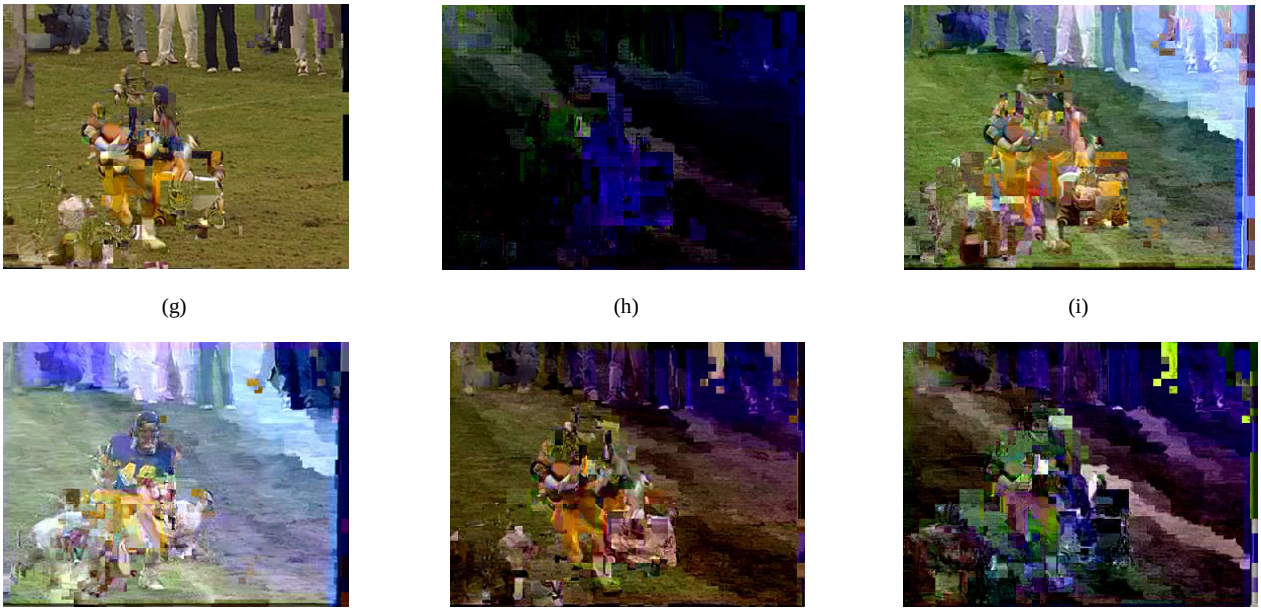
Results of successful guessing

The MVDs and the coefficients are separately encrypted. Thus, guessing the MVD signs will only improve the motion characteristics of the sequence while the luminance (luma) and color (chroma) values will still be incorrect, resulting in obscured video pictures. Two videos were selected for the experiments and visual results; the *Foreman* sequence is selected due to the visibility of the human face and the inclusion of moderate motion, along with a camera pan towards the end of the sequence, while the *Football* sequence is selected as it includes fast motion. Figure 6 shows the extent of any quality improvement after correctly guessing various relatively high percentages (as much as 50 or 80%) of MVD and TC signs. As will be observed from the *Foreman and Football* video sequences, any gain in visual appearance is limited, even at high percentages of correctly guessed MVD signs and TC values. The quality measures PSNR and SSIM for the guessed videos (Figure 6) are given in Tables 8(a) and 8(b); measured values are below the range of watchable video.



(a) (b) (c)

(d) (e) (f)



(g) (h) (i)

(j) (k) (l)

Figure 6: The effect of correct guessing up to a given percentage on (a-f) *Foreman* video (Frame # 84) and (g-l) *Football* video (Frame # 140). (a)(g) Full MVD signs encryption only, (b)(h) Full TC signs encryption only, (c)(i) 50% MVD and 80% TC signs (CAVLC) guessed correct, (d)(j) 80% MVD and 50% TC signs (CAVLC) guessed correct, (e)(k) 50% MVD and 80% TC signs (CABAC) guessed correct, (f)(l) 80% MVD and 50% TC signs (CABAC) guessed correct.

Table 8(a): PSNR and SSIM results for *Foreman* video after guessing attack.

SE with guessing percentages	**PSNR (dB)**			**SSIM**
	Y	U	V	
Only MVD signs encryption	17.0	33.4	32.5	0.534
Only TC signs encryption	7.9	20.2	25.3	0.235
CAVLC				
50% MVD and 80% TC signs guessed correct	12.1	27.7	23.3	0.377
80% MVD and 50% TC signs guessed correct	9.3	23.9	20.6	0.290
CABAC				
50% MVD and 80% TC signs guessed correct	11.7	29.8	27.1	0.381
80% MVD and 50% TC signs guessed correct	9.9	25.1	25.2	0.371

Table 8(b): PSNR and SSIM results for *Football* video after guessing attack.

SE with guessing percentages	**PSNR (dB)**			**SSIM**
	Y	U	V	
Only MVD signs encryption	19.4	27.2	33.4	0.336
Only TC signs encryption	8.8	18.4	23.5	0.114
CAVLC				

50% MVD and 80% TC signs guessed correct	16.5	21.3	30.1	0.356
80% MVD and 50% TC signs guessed correct	14.0	13.7	23.0	0.259
CABAC				
50% MVD and 80% TC signs guessed correct	14.3	23.8	27.7	0.375
80% MVD and 50% TC signs guessed correct	11.9	21.4	25.7	0.358

Table 9 shows the occurrences of MVD and TC signs within typical frames of the *Foreman* sequence. As the Table shows, the Intra Decoding Refresh (IDR) anchor frame has no MVD signs, while the predicatively-coded P- and bi-predicatively-coded B-frames have a significant number of MVD signs. From the Table 9, it is apparent that CAVLC and CABAC coding results in different frame sizes, though coding the same sequence with the same QP value.

Table 9: Frame type characteristics for *Foreman* video.

	CABAC			**CAVLC**		
Frame Type	**Frame Size (bits)**	**MVD count (1000s)**	**Signs of TC count**	**Frame size (bits)**	**MVD count (1000s)**	**Signs of TC count**
IDR	211456	0	39203	213432	0	41021
P	40376	667	4145	41040	659	5001
B	32144	457	6688	35432	441	8089

4.4 Strength of SE scheme

From the experiments of Section 4 regarding the distribution of syntax elements, and the investigations into the possibility of guessing the values of the encrypted syntax elements, it can be verified that a known-plaintext attack on signs of MVD and TCs cannot be successful against such an SE scheme [8], whether it is applied to CAVLC or CABAC entropy coding.

Chosen cipher-text and plaintext attacks are usually successful due to weak cipher algorithms, and cannot be successful against an entropy-coding encryption system for four reasons:

1. AES-CFB is a very strong algorithm and does not produce repeated cipher patterns, which an attacker might otherwise observe. Moreover, the 128-bit key length also makes the cipher un-breakable in all practical terms. Thus, brute force attacks will be unsuccessful.

2. The syntax elements are independently encrypted and have no correlation with each other. Motion vectors only deal with motion of the video, while transform coefficients only affect the spatial resolution of the video. Therefore, MVD and TC sign encryption have no relation to each other. Consequently, correlation attacks have little hope of success.
3. Suffixes exist for non-zero coefficients only. The strong point in respect to their encryption is that they are comprised of a variable number of bits, unlike the single sign bit. This scheme makes an attacker's task very difficult.
4. The probability of guessing 80% correct known signs, from the total of ten signs in a single slice is as low as 5.4%. The results on slices are assumed to be based on 10 to 50 sign bits; but in reality a large number of TC signs exist within each slice. This makes it likely that the probability of a successful attack using this technique is very low. Moreover, combining the probability of guessing the three parameters of signs of MVDs and TCs, and bits for suffixes, makes the probability of watching a good quality slice indeed very low. Considering that each video frame has several slices and a video sequence is made up of several video frames, the probability of getting a watchable video clip will be virtually zero.

5. Conclusion

Partially visible video data are at risk of guessing attacks that can render the video 'watchable' if not fully satisfactory. Balanced against the risk of an attack on broadcast video must be the gain, as networked TV (i.e. Internet Protocol TV or IPTV) packages are relatively cheap, costing around £2 a day [39], perhaps making it easier to purchase a subscription than launching an attack to enhance the visual quality. The strength of any encryption system is measured in terms of the time and resources required to recover the plaintext. It is important that the time and resources required by an attacker are quantifiable.

The work of this paper has proposed a methodology of determining the probability that an attacker can carry out an attack on SE using a guessing approach. What has been studied is that it is possible that there will be a large number of encrypted syntax elements even in a short video clip of few seconds. Moreover, even if as many as 80% of the bits encrypting signs are guessed, the video quality is still significantly impaired. It is also apparent (from

the numbers in Table 2) that CAVLC generates more TCs and suffixes than CABAC, which makes the CAVLC encoded bit-stream even stronger against guessing attacks than CABAC. Nevertheless CABAC provides more compression by producing less residual data. Hence, there is a trade-off between the number of syntax elements and compression efficiency. Finally, it is concluded that the proposed cryptanalysis method is sufficient to demonstrate that SE in the entropy coding stage, which exploits all the important video characteristics such as motion and texture characteristics, is secure. This is especially relevant for real-time commercial services, which, on the evidence of this paper, are sufficiently secure after the application of state-of-the-art selective encryption algorithms.

References

[1] A A. Massoudi, F. Lefebvre, C. De Vleeschouwer, B. Macq, J.-J., Quisquater, Overview on selective encryption of image and video: Challenges and perspective, EURASIP Journal on Information Security 2008, article no. 5, (2008) 1-18.

[2] B. Furht, D. Socek, A. M. Eskicioglu, Fundamentals of multimedia encryption methods, in: B. Furht and D. Kirovski (Eds.) Multimedia Security Handbook, CRC Press, Boca Raton, FL, 2005, pp. 95–132.

[3] E. Muharemagic, B. Furht, Survey of watermarking techniques and applications, in: B. Furht and D. Kirovski, (Eds.), Multimedia Security Handbook, CRC Press, Boca Raton, FL, 2005, pp. 221–260.

[4] H. Wu, Streaming media encryption, in: B. Furht and D. Kirovski (Eds.), Multimedia Security Handbook, CRC Press, Boca Raton, FL, 2005, pp. 197–217.

[5] T. Stütz, A. Uhl, A survey of H.264 AVC/SVC encryption, IEEE Transactions on Circuits and Systems for Video Technology 22 (3) (2012) 325-339.

[6] I.E.G. Richardson, H.264 Advanced Video Compression Standard, Wiley & Sons, Chichester, U.K, 2010.

[7] M. N. Asghar, M. Ghanbari, M. Fleury, M. Reed, Efficient selective encryption with H.264/SVC CABAC Bin-strings, in: International Conferance on Image Processing, 2012, .

[8] M. N. Asghar, M. Ghanbari, M. Reed, Sufficient encryption with codewords and bin-strings of H.264/SVC, in: IEEE International Conference on Trust, Security and Privacy in Computing and Communications.(Trustcom), 2012, pp. 443-450.

[9] T. C. Chen, Y. W. Huang, C. Y. Tsai, B. Y. Hsieh, L. G. Chen, Architecture design of context-based adaptive variable-length coding for H.264/AVC, IEEE Transactions on Circuits and Systems II, Exp. Briefs 53 (9) (2006) 832–836.

[10] D. Marpe, H. Schwarz, T. Wiegand, Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard, IEEE Transactions on Circuits and Systems for Video Technology 17 (9) (2003) 620-636.

[11] H. Schwarz, D. Marpe, T. Wiegand, Overview of the scalable video coding extension of the H.264/AVC standard, *IEEE Transactions on Circuits and Systems for Video Technology* 17 (9) (2007) 1103-1120.

[12] D. Kahn, The codebreakers: The comprehensive history of secret communication from ancient times to the Internet, 2nd ed., Simon and Schuster Publ., New York, NY, 1997.

[13] A. Said, Measuring the strength of partial encryption schemes, in: International Conference on Image Processing, 2005, vol. II, pp. 1126-1129.

[14] B. Furht, E. Muharemagic, D. Socek (Eds.) Multimedia Encryption and Watermarking, Springer Verlag, New York, NY, 2005.

[15] A. Uhl, A. Pommer, Image and video encryption: From Digital Rights Management to secured personal communication. Springer-Verlag, New York, NY, 2005.

[16] T. Stűtz, A. Uhl, Format-compliant encryption of H. 264/AVC and SVC, in: IEEE International Symposium on Multimedia, 2009, pp. 446 – 451.

[17] S. W. Park, S. U. Shin, Efficient selective encryption scheme for the H.264/Scalable Video Coding (SVC), in Fourth International Conference on Networked Computing and Advanced Information Management, 2008, pp. 371-376.

[18] Z. Shahid, M. Chaumont, W. Puech, Selective and scalable encryption of enhancement layers for dyadic scalable H.264/AVC by scrambling of scan patterns, in IEEE International Conference on Image Processing, 2009, pp. 1273-276.

[19] E. Magli, M. Grangetto, G. Olmo, Transparent encryption techniques for H.264/AVC and H.264/SVC compressed video, Journal of Signal Processing 91 (5) (2011) 1103-1114.

[20] B.B. Zhu, M.D. Swanson, S. Li, Encryption and authentication for scalable multimedia: Current state of the art and challenges, in: SPIE Internet Multimedia Management Systems, 5601 (2004) 157–170.

[21] G.B. Algin, E.T. Tunali, Scalable video encryption of H.264/AVC codec, Journal of Visual Communications and Image Representation 22 (4) (2011) 353-364.

[22] Z. Shahid, M. Chaumont, W. Puech, Fast protection of H.264/AVC by selective encryption of CAVLC and CABAC for I and P frames, IEEE Transactions on Circuits and Systems for Video Technology 21 (5) (2011) 565-576.

[23] Y. Wang, M. O'Neill, F. Kurugollu, A tunable encryption scheme and analysis of fast selective encryption for CAVLC and CABAC in H.264/AVC, IEEE Transactions on Circuits and Systems for Video Technology, 23 (9) (2013) 1476 - 1490.

[24] T.E. Seidel, D. Socek, M. Šrámka, Cryptoanalysis of video encryption algorithms, Tata Mountain Mathematics Publications 29, (2004) 79-87.

[25] M. Ghanbari, Standard codecs: Image compression to advanced coding, IET Press, Stevenage, UK, 2003.

[26] B. Bhargava, C. Shi, S.Y. Wang, MPEG video encryption algorithms, Multimedia Tools and Applications, 24 (2004) 57-79.

[27] C. Shi, B. Bhargava, A fast MPEG video encryption algorithm, in International Conference on Multimedia, 1998, pp. 55-61.

[28] T. Lookabaugh, D.C. Sicker, D.M. Keaton, W.Y Guo, I. Vedula, Security analysis of selectively encrypted MPEG-2 streams, in: Multimedia Systems and Applications VI, SPIE Proceedings vol. 5241, 2003, pp. 1-12.

[29] G. Jakimoski, K.P. Subbalakshmi, Crytoanalysis of some multimedia encryption schemes, IEEE Transactions on Multimedia 10 (3) (2008) 330-338.

[30] J. Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockhammer, T. Wedi, Video coding with H. 264/AVC: tools, performance, and complexity, IEEE Circuits and Systems Magazine 4 (1) (2004) 7-28.

[31] L. Harte, Introduction to Digital Rights Management (DRM); Identifying, tracking, authorizing and restricting access to digital media, Althos Publishers, 2 Fukua Varina, NC, 2007.

[32] H. D. Engel, R. Kutil, A. Uhl, A symbolic transform attack on lightweight encryption based on wavelet filter parameterization, in: ACM Multimedia and Security Workshop, 2006, pp. 202-207.

[33] T. Lookabaugh, D.C. Sicker, Selective encryption for consumer applications, IEEE Communications Magazine 42 (5) (2004) 124-129.

[34] Q. Huynh-Thu, M. Ghanbari, The accuracy of PSNR in predicting video quality for different scenes and frame rates, Telecommunication Systems, 49 (1) (2012) 35-48.

[35] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli, Image quality assessment: From error measurement to structural similarity, IEEE Transactions on Image Processing 13 (4) (2004) 600–612.

[36] B. Esslinger, The CrypTool script: Cryptography, mathematics and more, (10th ed.) {distributed with CrypTool version 1.4.30}, 2010.

[37] HHI (2008-2011), JM reference software. http://iphome.hhi.de/suehring/tml/ (last accessed 24th Nov. 2012).

[38] P. Lambert, W. De Neve,Y. Dhondt, R. Van de Walle, Flexible macroblock ordering in H.264/AVC, Journal of Visual Commun. and Image Representation 17 (2) (2006) 358-375.

[39] Sky TV subscription charges webpage: http://www.sky.com/shop/ (last accessed Feb. 23, 2013)