Efficient encryption technique for H.264/AVC videos based on CABAC and logistic map

Fatma K. Tabash1 · Muhammed Izharuddin1

Received: 14 January 2018 / Revised: 21 July 2018 / Accepted: 2 August 2018 / Published online: 10 August 2018 © Springer Science+Business Media, LLC, part of Springer Nature 2018

Abstract

Nowadays, the use of real time video communication is growing with a rapid pace. For example, Search and Rescue (SAR) applications like earthquake rescue, avalanche victims, wildfire monitoring in highway and surveillance are some of the applications of real time video communication. Most of these video communications require secure transmission of signals. For real time video communication, the processing time is considered as the most important metric to be optimized. Thus, finding a simple and time efficient encryption technique for securing the transmitted data becomes mandatory. The motive behind this work is to design a system that can process the video signal in real time providing a secure communication. In this paper, we present an efficient encryption technique which has low computation complexity. The proposed technique is based on Context-Adaptive Binary Arithmetic Coding (CABAC) where the bin-string of Intra-Prediction Mode, Motion Vector Difference (MVD) and residue coefficients are encrypted with the random behavior of chaotic systems. The random bit streams used for encryption are generated with chaotic systems using Logistic map. The experimental results show that the proposed technique is very effective for real-time applications and robust against different types of attacks.

Keywords Encryption · Video coding · CABAC · H.264/AVC

1 Introduction

1.1 Research motivation

With the continuous development and innovation of computer networks and wireless communication technologies, digital media data (images, sounds, videos, etc.) are infiltrating more and more in people’s lives. Therefore, the applications of multimedia videos have become more diversified, for example, related video conferencing, video on demand, and real-time

---

* Fatma K. Tabash
  fatma.tabash86@gmail.com

1 Department of Computer Engineering, Aligarh Muslim University, Aligarh, UP 202002, India
video monitoring system. Similar to the other data through the network transmissions, video data without encryption will be very vulnerable to attacks, such as data information interception, data tampering, illegal copying, and pirating. The security of video data is very important to the most of multimedia applications and it is considered widely. Encryption is one of the best technologies in Digital Rights Management (DRM) schemes which are used to protect copyright works. H.264/AVC video is one of the most common formats used for video compression and it spans a long wide range of real-time video communication. Thus, it is mandatory to develop an encryption technique to protect H.264/AVC videos which achieves the requirements of real time nature for videos communication.

For the encryption strategy, CABAC entropy coding technique is selected to be the designated stage for encryption. CABAC is the latest entropy coding technique which is used in the latest video coding standards as H.264/AVC and HEVC (High Efficiency Video Coding) standards. It is a stage through the compression process where the binary codes that represent the video data are generated. The binary codes which are generated by CABAC are expected to have the lowest bitrate load. Thus, doing encryption in CABAC stage in correct way will not increase the bitrate or affecting the compression ratio. On the other hand, CABAC is a complex technique where any simple mistake through the encryption process will lead to break the format compatibility and will impede the function of NAL (Network Abstraction Layer).

1.2 Previous work

H.264/AVC video is one of the most common formats used for video compression and encoding. Its popularity comes from the saving of 50% of the bit-rate required to transmit video stream. Accordingly, it spans a wide range of application starting from low bit-rate internet streaming applications to high bit-rate HDTV broadcast applications. Safeguarding video streams from unauthorized users become an urgent demand. Encryption techniques are the most commonly used ways to protect the information of video stream from the obtrusive users. Encryption techniques for H.264 have many approaches either to encrypt the total bit stream for gaining the maximum security at the cost of high computational cost or with partially encrypting bit stream. Partial/selective encryption approach is commonly used because they do not increase the total bit rate significantly or change the format compatibility. Above all, with partial encryption, total computation cost is also reduced. The full-stream encryption approach is followed when the high security communication is required as in military needs or news scoops, but the drawbacks of these approaches are changing the format compatibility and increasing the computational cost. Many security schemes for images are proposed in various research literatures [4, 8] and [9], but for real time video security, very few schemes are found in the literature. Partial encryption approach can effectively be used to protect multimedia data with low bandwidth taxation. In literature, many techniques have been proposed to partially encrypt H.264/AVC video data: Shahid et al. [14] proposed a selective encryption technique to protect H.264/AVC video based on two entropy coding modules (CAVLC and CABAC) simultaneously. In CAVLC, the signs of trailing ones and the signs of non-zero coefficients are encrypted using AES cipher, whereas in CABAC, the non-zero coefficients of same length bin-stream are encrypted along with the signs of non-zero coefficients with the same cipher. The drawback of this algorithm is the
weakness in perceptual security i.e. it does not conceal the information of the plaintext. Su et al. [15] proposed another encryption scheme based on IPM, MVD and residue data. IPM data are encrypted by doing XOR between IPM codeword and 3 bits of random stream. For encrypting MVD, only flipping the sign of MVD is considered to keep format compliant with reduced complexity. In case of residue, the array of coefficients is divided into units of nonzero levels and zero-run pair i.e. nonzero level associated with its preceding zeros. After dividing, a scrambling technique is applied into these units. The drawback of these algorithms is the increase of bit-rate. Yeung et al. [18] proposed different encryption scheme based on new unitary transforms as an alternative to the common integer DCT transform. The disadvantage of this technique is that it needs new designed H.264/AVC encoder which is cost much. Lian et al. [5] proposed another partial-encryption technique based on CAVLC. The suffix part of Exp-Golomb code of IPM is encrypted with a random stream cipher. The signs of MVD and AC coefficients are also encrypted. DC coefficients are encrypted differently: in CAVLC, the signs trailing ones T1 and levels are encrypted. But encrypting the sign of levels in CAVLC is increasing the bit-rate because they are used in encoding process. Other encryption schemes have been proposed for H.264/AVC standard [1, 3, 6, 10, 11, 16, 17].

2 Background

2.1 Chaotic systems

Recently, the use of chaotic theory is widely used to implement security system for protecting the data. Although the traditional ciphers like DES, IDEA, RSA and AES are secured and most commonly used in encryption algorithms, they are very complex and need a high computational cost. For this, they are more suitable for binary/text applications where the size of the plaintext is considered moderate. For multimedia applications, especially for videos, the size of data for processing is very huge and encrypting them with traditional cipher will produce very significant overhead in the processing time. In addition, the high-redundancy between two consecutive frames makes traditional ciphers fails to hide the significant information of the source file. For these reasons, there has been a significant trend towards other ciphering systems that has low computational complexity and simple implementation. Chaotic systems have been proposed to be an efficient encryption technique for real-time multimedia applications. There are many reasons to use chaotic system as an alternative to traditional ciphers: time evolution of the chaotic signal strongly depends on the initial conditions and the control parameters of the generating functions: slight variations in these quantities yield quite different time evolutions. Consequently, those initial conditions and the control parameters can be used as a shared encryption key.

Different types of chaotic maps are used for encryption process. Logistic Map is one the simplest and very secured functions. Equation (1) shows mathematical form of Logistic Map:

\[ x_{n+1} = r x_n (1 - x_n) \]  

(1)
The chaotic value depends on the initial condition $x_n$ and control parameter $r$. The initial condition $x_n$ can be any real point number between 0 and 1 and the control parameter has range from 3.7 to 4. (Highly chaotic behavior is observed when the value parameter $r$ is selected between 3.9 to 3.999).

$$0 < x_n < 1 \text{ And } 3.7 \leq x_n < 4$$

In our work, we select Logistic Map for encryption process because it is very simple and much suitable for real time applications.

### 2.2 Entropy coding process of IPM, MVD and residue coefficients in CABAC:

#### 2.2.1 Entropy coding for IPM (Intra-Prediction Modes)

The encoding process for IPM is presented in three parts: first, explanation of Syntax Elements (SE) that carries the values of IPMs. Second, different binarization processes in CABAC is taken. Finally, dedication of SEs with their suitable binarization process is achieved.

**Part 1:** In CABAC, IPM for $4 \times 4$, $8 \times 8$ and chroma blocks are encoded in the bit-stream using five syntax elements:

- **prev_intra$4 \times 4$ pred_mode_flag**: this flag is enabled, if IPM value of the current $4 \times 4$ block equals the minimum mode of the upper and left neighboring blocks.
- **rem_intra$4 \times 4$ pred_mode**: this syntax element holds the value of the best IPM mode in the current $4 \times 4$ block.
- **prev_intra$8 \times 8$ pred_mode_flag**: this flag is enabled, if IPM value of the current $8 \times 8$ block is equal the minimum mode of the upper and left neighboring blocks.
- **rem_intra$8 \times 8$ pred_mode**: holds the value of the best IPM mode in the current $8 \times 8$ block.
- **intra_chroma_pred_mode**: holds the value of the best chroma-prediction mode.

In case of $16 \times 16$ block texture, the entire $16 \times 16$ macroblock will be considered as one block for prediction. Therefore, only one value of prediction mode is considered for the entire macroblock. For this reason, there is no designated syntax element to hold the value of Intra_$16x16$ prediction mode. The value of Intra_$16x16$ prediction mode will be included implicitly under the syntax element **mb_type** (that holds the type of macroblock) alongside with the value of Coded Block Pattern (CBP).

**Part 2:** IPMs are binarized in CABAC using three types of binarization processes:

- Fixed-length (FL) binarization process [2]: FL binarization is constructed by using a fixedLength binary representation of unsigned integer of the syntax element value.
fixedLength = \( \lceil \log_2(c_{\text{Max}} + 1) \rceil \) where, c_{\text{Max}} is the maximum value of unsigned integer represented using binary string.

- **Truncated unary (TU)** [2]: is truncated unary binarization. All values of SE which are less than c_{\text{Max}} are binarized using unary binarization. All greater values are binarized with the same as c_{\text{Max}} value.
- **Unary (U) binarization process** [2]: The bin-string is consists of 1s of length equal to the value of SE terminated with zero.

**Part 3:** Each SE is binarized with a specified binarization process:

- **prev_intra4x4_pred_mode_flag**: binarized using FL with c_{\text{Max}} = 1 i.e. one bit binary string to represent SE. If flag is enables The bin-string = “1” else it is “0”. The same manner is to binarize prev_intra8x8_pred_mode_flag.
- **rem_intra4x4_pred_mode**: binarized using FL with c_{\text{Max}} = 7 i.e. it needs 3 bits to represent the value of SE. The same manner for rem_intra8x8_pred_mode.
- **intra_chroma_pred_mode**: binarized using TU of c_{\text{Max}} = 3 i.e. all value smaller than 3 is encoded using Unary binarization, else bin-string will consists of two ones “11”.

2.2.2 **Entropy process for Residue Coefficients:**

Residue coefficients are encoded in CABAC using the next syntax elements:

- **significant_coeff_flag**: indicates if the current coefficient is significant/nonzero or not. This means it draws the map for the positions of nonzero coefficients.
- **last_significant_coeff_flag**: indicates the position of the last nonzero coefficient.
- **coeff_abs_level_minus1**: holds the absolute value of the nonzero coefficients minus one.
- **coeff_sign_flag**: holds the sign of nonzero coefficients.

It is noticed that the encoding process of residual data depends on four different types of SEs which are highly correlated. Accordingly, any encryption process for residual data should be done carefully because any change in these information may lead to break format compatibility of the transmitted source.

Three of SEs: significant_coeff_flag, last_significant_coeff_flag and coeff_sign_flag are binarized using Fixed-length (FL) binarization process with c_{\text{Max}} = 1 because they are flags. So, 1 bit bin-string will be sufficient to hold the value. For coeff_abs_level_minus1, Concatenated unary/k-th order Exp-Golomb (UEGk) binarization process with k = 0 [12] is used.

2.2.3 **Encoding process for motion vector difference (MVD)**

Motion Vector Difference is encoded using two syntax elements: mvd_0[0], mvd_1[0] to curry the values of MVD in X-axis.
mvd_10[][][1], mvd_11[][][1] to curry the values of MVD in Y-axis.

Encryption process for MVD have to be done carefully because any error in encrypting process will lead to format incompatibility and cannot decode the source file. The binarization process for MVD is done using Concatenated unary/k-th order Exp-Golomb (UEGk) binarization process with $k = 3$ [12].

3 Proposed scheme

The proposed scheme is based on the partial encryption of the three encoding parameters: (i) IPM, (ii) MVD and (iii) residue coefficients. Intra Prediction Mode (IPM) is a good choice for encryption process because the current block is predicted from the edges of the neighboring blocks. Therefore, encrypting one intra block will propagate the chaos to the other neighboring blocks. The IDR (Instantaneous Decoding Refresh) frame consists of intra-predicted blocks and any P frame predicted using motion compensation will be encrypted using the chaos propagated from the IDR frame. Encrypting only IPM parameter is unsecure because it is vulnerable to replacement attack [13]. For this reason, encrypting residue data will enhance the security of encryption process. Also MVD data will be encrypted to hide the motion information in the video.

3.1 Initialization process

At the start of encryption process, two steps are performed:

Step 1: Set the encryption key, including the initial value $x_0$ and control parameter $r$, for example $x = 0.123456789555777, r = 3.9$. (Initial value $x_0$ and control parameter $r$ were selected as the encryption key because a very small change in the initial value of the chaotic system will lead to completely different chaotic behavior. Accordingly, any change in the initial key will generate a different pseudorandom number sequences that are used in the encryption process).

Step 2: Do 150 times iteration using formula [14] ($x_1, ..., x_{101}$). In each iteration, one floating point number will be generated. Each number consists of 15 digits after the decimal point.

3.2 Encryption process of IPM

Since prev_intra4x4_pred_mode_flag and prev_intra8x8_pred_mode_flag syntax elements are control parameters, they can’t be encrypted because that will introduce incorrect encoding results. SE intra_chroma_pred_mode which encrypts the chroma planes also will not be selected for encryption to simplify the encryption process and reduce the computational cost. Encrypting chroma planes in each macroblock will not improve the encryption results due to many reasons; the first one is the human eye is much less sensitive to chrominance than luminance. Thus, encryption of chroma plane will not improve the security significantly compared to the overload which is added due to encryption process. The second reason is that chroma plane in each
Macroblock is only one block of size $8 \times 8$ block compared to luma which is composed of four blocks of $8 \times 8$. The third reason is that \texttt{intra_chroma_pred_mode} is binarized by Unary binarization process where each code has different size which require a complicated encryption process compared to other data used in encryption. \texttt{rem_intra4x4_pred_mode} and \texttt{rem_intra8x8_pred_mode} are holding the values of the selected best modes, thus they will be selected for the encryption process. Next is the pseudocode of the encryption process:

```plaintext
if(Up_Macroblock!=NULL && Left_Macroblock!=NULL){
    x = r * x * (1.0-x)
    dec=x*100
    md=dec\%8
    current_IPM=XOR(current_IPM,md)
}
```

First, the availability of neighboring macroblock is checked, to ensure the validity of new ciphered prediction modes. Next, the random floating point number (chaotic number) is obtained by using expression in (1) and it is multiplied by 100. Mod 8 of this number is XORed with the original IPM to get the encrypted IPM.

### 3.3 Encryption process of MVD

To reduce the time and computational complexity, only the sign of MVD will be encrypted. If the random value generated using formula (1) is greater than 0.5, then the sign of MVD will be flipped else it will remain the same.

```plaintext
x = r * x * (1.0-x)
if (x>0.5)
    md=1
else md=0
mv_sign=XOR(mv_sign,md)
```

### 3.4 Encryption process of residue coefficients:

The encryption process for residue coefficients is done based on three stages as follow:

**Stage 1:** This stage encrypts the sign of coefficients. If the random number by formula (1) is greater 0.5 then set the value of SE \texttt{coeff_sign_flag} as 1, else set as 0.

**Stage 2:** In this stage, we encrypt only the suffix part of the bin-string of residue coefficients to avoid format incompatibility and increasing in bitrate. The suffix part is XORed with a random binary string of same length.

**Stage 3:** To enhance the security, the first DC coefficient in each block is encrypted. The first DC coefficient has the most energy value the in energy compaction of the...
Accordingly, encrypting DC coefficient will propagate more ambiguity into the overall video data. The process for encrypting the DC coefficient is shown in Fig. 1.

\[ y = \text{ceil}(\log_2(x + 1)), \quad \text{where } x \text{ is unsigned integer} \]  

4 Experimental analysis

For analyzing the results of the proposed encryption technique, we used H.264/14496-10 AVC reference software JM 19 and raw video sequences in QCIF format. The video sequences are: “Bus”, “City”, “Crew”, “Football”, “Foreman”, “Harbour”, “Mobile”, “Ice”, and “Soccer”. Each of them represents different combinations of motions, colors, contrast and objects. The first 100 frames in each sequence are selected for experiments. The results are compared with the techniques in [5, 14] in section I (any number or sequence of frames may be considered for experiment, but to compare the result with existing work, only these experimental results and values are presented here).

4.1 Perception security

Whenever, it is possible to conceal the information of the plain-text from the human perception, it increases the protection against the interference of unauthorized users. Figure 2 shows the encryption results using the proposed technique for the different video samples. The first column is the first frame in the video which is IDR frame. The second column is the corresponding encrypted frame. From the figure, it is clearly seen that the encrypted videos are not recognizable for the human vision. Table 1 presents the PSNR values for measuring the quality of the encrypted video compared to the original one in the three color planes (Y U V). These statistical results give the quantitative value of the encrypted signal. As it is seen in the Table 1, the PSNR values in the most of the
selected samples are less than 20 dB, which indicates that the encrypted signal is highly opaque. In the same table, the results of our proposed technique are compared with existing scheme, Selected Encryption(SE) in [14]. The results show that our proposed technique gives better value of PSNR of the encrypted video as compared with [14] in all the three color planes. Table 2 gives a comparison between the proposed technique and encryption technique in [5] for PSNR measurements. Figure 3 shows a comparison of the first encrypted frame of foreman video between the proposed technique and SE technique in [14] at different QP values. From the results, it is inferred that the proposed technique can improve the security of the video, since it gives better encryption as compared to the existing one [14]. In the technique proposed in [14] the texture and facial features of the picture can be easily recognized. Table 3 shows the values of PSNR of foreman video at

Fig. 1 Encryption process of DC coefficient
| Original Frame | Encrypted 1st Frame |
|---------------|--------------------|
| ![Image](image1.jpg) | ![Image](image2.jpg) |
| ![Image](image3.jpg) | ![Image](image4.jpg) |
| ![Image](image5.jpg) | ![Image](image6.jpg) |
| ![Image](image7.jpg) | ![Image](image8.jpg) |
| ![Image](image9.jpg) | ![Image](image10.jpg) |
| ![Image](image11.jpg) | ![Image](image12.jpg) |
| ![Image](image13.jpg) | ![Image](image14.jpg) |

**Fig. 2** Encrypted frames using the proposed technique
different QP values and compared with the values presented in [14]. It is shown that the proposed technique has less PSNR values in decibel which means that it more efficiently conceal the information of the video.

4.2 Cryptographic security

Cryptographic security depends on the ciphers adopted by the encryption scheme. In the proposed technique, the adopted cipher is based on chaotic logistic map where its security analysis is clearly proved in [7].

4.2.1 Key space

In the proposed technique, the secret key is combination of the initial parameters of logistic map: the initial condition \( x_n \) and the control parameter \( r \), where \( x_n \) is any floating point number \([0,1)\) and \( r \in [3.5,4)\). Since \( x_n \) is double-precision floating point number then 64 bits are used to present this number in binary. Thus, for decrypting the cipher-text, it requires to try at least \( 2^{64} \) combinations of the key space. In this case, Brute-Force Attack fails to decrypt the interested plaintext.

| Table 1 | PSNR values of the encrypted videos using proposed technique compared with SE in [1] |
|---------|-----------------------------------------------|
|         | PSNR(Y)                                      | PSNR(U)                                      | PSNR(V)                                      |
|         | Proposed Orig. Enc. | SE [14] Orig. Enc. | Proposed Orig. Enc. | SE [14] Orig. Enc. | Proposed Orig. Enc. | SE [14] Orig. Enc. |
| Bus     | 44.3 7.1 43.7 7.7 | 45.1 25.5 45.1 25.4 | 46.4 26.5 46.4 27.0 |
| City    | 44.1 11.3 43.8 11.1 | 46.5 29.7 45.7 30.2 | 47.4 31.7 46.8 31.7 |
| Crew    | 44.9 9.6 44.5 10.0 | 45.5 21.1 45.8 22.0 | 45.3 17.8 45.7 20.2 |
| Football| 44.8 10.4 44.2 11.3 | 46.0 16.0 45.7 14.6 | 46.3 22.3 46.1 24.3 |
| Foreman | 45.0 8.1 43.9 10.4 | 46.2 22.8 45.5 23.9 | 47.7 22.5 47.6 23.2 |
| Harbour | 44.1 9.8 43.7 9.8 | 45.1 20.7 45.4 22.9 | 45.9 31.0 46.6 31.7 |
| Ice     | 47.2 9.9 46.1 10.4 | 48.8 25.3 48.6 25.3 | 49.2 19.4 49.1 19.7 |
| Mobile  | 44.3 8.5 43.8 8.8 | 44.3 14.4 44.2 12.5 | 44.3 11.4 44.1 11.8 |
| Soccer  | 44.9 10.4 43.6 10.6 | 46.8 17.4 46.5 20.8 | 47.8 21.8 47.8 22.2 |

| Table 2 | Comparison of PSNR between the proposed technique compared and technique in [5] |
|---------|-----------------------------------------------|
|          | PSNR(Y)                                      |
|          | Proposed Orig. Enc. | Technique [5] Orig. Enc. |
| Salesman | 44.8 6.4 | 36.2 9.5 |
| Foreman  | 45.0 8.1 | 36.7 11.4 |
| Mobile   | 44.3 8.5 | 33.7 7.4 |
4.2.2 Replacement attack

One of the common attacks in video encryption area is the replacement attack. This type of attacks is trying all the possible combinations of the parameters (Intra-Prediction modes, MVD and residue coefficients) that are used in the encryption process in trying to reveal more information about the cipher-text. Accordingly, we will count the number of trials that the attacker can do to get more clear decrypted frames. To break on macroblock, The attacker needs to break 16 sub-block of $4 \times 4$ size. Thus to guess one sub-block, the attacker needs $2^3 \times 2^3 \times 2^2 \times 2 \times P(16, 16)$ number of trials to decrypt one $4 \times 4$ sub block. To decrypt the entire macroblock, the attacker needs to multiply the number by 16. Thus, to attack one macroblock using replacement attack, a huge number of trials needed to be applied, so decrypting the entire frame will be almost impossible.

4.3 Computational cost

4.3.1 Computational complexity

The proposed technique is based on the chaotic systems which are very simple and consuming very small computational power. The mathematical form of Logistic map in Eq. (1) shows that

| PSNR(Y) | PSNR(U) | PSNR(V) |
|---------|---------|---------|
| Proposed | SE | Proposed | SE | Proposed | SE |
| 18 | 8.1 | 10.4 | 22.8 | 23.9 | 22.6 | 23.2 |
| 24 | 8.9 | 9.7 | 22.1 | 24.9 | 24.3 | 25.0 |
| 30 | 8.9 | 9.2 | 22.8 | 24.9 | 24.1 | 24.0 |
| 36 | 8.0 | 8.2 | 23.7 | 24.3 | 23.0 | 23.3 |
| 42 | 7.3 | 8.6 | 24.9 | 26.8 | 25.0 | 24.6 |
the hardware implementation of Logistic map consists of just two multiplication operations and one subtraction. This exhibits that the computational complexity and hardware implementation for our proposed technique are much efficient than the traditional ciphers which are very complex and they require a huge number of computational power. By comparing the proposed technique with technique in [14], it is shown that the encryption process in [14] is based on AES technique which is very complex and it needs much computational power. Regarding the technique in [5], it is based on encrypting the sign of levels in CAVLC but this increases the bitrate of the stream.

### 4.3.2 Computational time

A key feature to assess the efficiency of the proposed technique is to optimize the reduction of processing time. In the proposed technique, we follow many steps to minimize the computational time. The first step is to use a simple encryption system based on chaotic logistic map. Next, we reduced the universe of discourse data which will be encrypted. Moreover, we perform encryption in CABAC stage through compression process which will not affect the bitrate. All these factors help to reduce the processing time significantly. Table 4 gives the rate of the increasing of the computational time due to the encryption process. Since the computer gives different execution at each test time, the average of five runs is considered for each value in the table. It is observed, that the impact of encryption taxation on the encoding process is very small. The same table also presents a comparison between the proposed technique and SE [14] for the increase of the computational time. It is clear that, the proposed technique consumes very small computational time compared to SE technique in [14]. Table 5 gives another comparison between the proposed technique and the technique in [5]. The proposed technique has much less time impact compared to [5] for the same types of video sequences.

| Sequence | Proposed % | SE [14] % |
|----------|------------|-----------|
| Bus      | 0.056      | 0.25      |
| City     | 0.038      | 0.23      |
| Crew     | 0.121      | 0.14      |
| Football | 0.052      | 0.18      |
| Foreman  | 0.032      | 0.20      |
| Harbour  | 0.028      | 0.26      |
| Ice      | 0.052      | 0.17      |
| Mobile   | 0.026      | 0.33      |
| Soccer   | 0.110      | 0.18      |

| Sequence | Proposed % | Technique [5] % |
|----------|------------|-----------------|
| Foreman  | 0.032      | 0.9             |
| Akiyo    | 0.170      | 1.1             |
| Mother   | 0.058      | 0.7             |
| Silent   | 0.155      | 1.0             |
| News     | 0.245      | 0.5             |
| Salesman | 0.056      | 0.5             |
5 Conclusion

This paper presents an efficient encryption technique to encrypt H.264/AVC video data. This algorithm is applied through the different modules of the compression pipeline which are: CABAC entropy coding, MVD, and residues transformation. The experimental results show that the proposed technique is highly secured against different types of attacks. The proposed technique obtains high time efficiency compared to the previous algorithm used for encryption. In addition, the encrypted video is highly chaotic where the information can be completely opaque.

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

1. Bodke N, Khule J, Shinde P, Kapse S, Kumavat K (2015) A Novel Approach for Codeword Substitution using Encrypted H.264/AVC Video Streams for Data Hiding. Int J Comput Appl 128(4):6–10
2. ITU-T Recommendation H.264 (2012) Series H: Audiovisual and multimedia systems. Infrastructure of audiovisual services - Coding of moving video. Advanced video coding for generic audiovisual services
3. Li Y, Liang L, Su Z, Jiang J (2005) A new video encryption algorithm for H.264. In: Proc. 5th ICICS, pp. 1121–1124
4. Li X, Ma J, Wang W, Xiong Y, Zhang J (2013) A Novel smart card and dynamic ID based remote user authentication scheme for multi-server environment. Math Comput Model ELSEVIER 58:85–95
5. Lian S, Liu Z, Ren Z, Wang H (2006) Secure advanced video coding based on selective encryption algorithms. IEEE Trans Consumer Electron 52(2):621–629
6. Lian S, Sun J, Liu G, Wang Z (2008) Efficient video encryption scheme based on advanced video coding. Multimedia Tools Applicat 38(1):75–89
7. Lian S, Sun J, Wang Z et al (2005) Security analysis of a chaos-based image encryption algorithm. Physica A: Statistical Mechanics and its Applications 351(2–4):645–661
8. Liao X, Guo S, Yin J, Wang H, Li X, Sangaiah AK (2018) Cubic reference table based image steganography. Journal of Multimedia Tools and Applications, Springer 77:10033–10050
9. Liao X, Zheng Q, Ding L (2017) Data embedding in digital images using critical functions. Journal of Signal Processing: Image Communication, Elsevier 58:156–156
10. Liu Y, Yuan C, Zhong Y (2007) A new digital rights management system in mobile applications using H.264 encryption. Proc 9th Int Conf Adv Commun Technol 1:583–586
11. Magli E, Grangetto M, Olmo G (2006) Conditional access to H.264/AVC video with drift control. In: Proc. IEEE ICME, pp. 1353–1356
12. Marpe D, Schwarz H, Wiegand T (2003) Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard. IEEE Trans Circuits Syst Video Technol 13(7):620–636
13. Podesser M, Schmidt H, Uhl A (2002) Selective bitplane encryption for secure transmission of image data in mobile environments. In CD-ROM Proceedings of the 5th IEEE Nordic Signal Processing Symposium (NORSIG 2002), Tromso-Trondheim, Norway
14. Shahid Z, Chaumont M, Puech W (2011) Fast Protection of H.264/AVC by Selective Encryption of CAVLC and CABAC for I and P frames. IEEE Trans Circuits Syst Video Technol 21(5):565–576
15. Su P-C, Hsu C-W, Wu C-Y (2011) A practical design of content protection for H.264/AVC compressed videos by selective encryption and fingerprinting. Multimedia Tools Applicat 52(2–3):529–549
16. Tong L, Dai F, Zhang Y, Li J (2010) Prediction restricted H.264/AVC video scrambling for privacy protection. Electron Lett 46(1):47–49
17. Wang Y, O’Neill M, Kurugollu F (2012) The Improved sign bit Encryption of Motion Vectors for H.264/AVC. EURASIP, 20th European Signal Processing Conference (EUSIPCO 2012), Bucharest, pp. 1752-1756
18. Yeung S-KA, Zhu S, Zeng B (2009) Partial video encryption based on alternating transforms. IEEE Signal Process Lett 16(10):893–896
Fatma K. Tabash is currently doing PhD in Computer Engineering in Aligarh Muslim University, Aligarh, India. She received her master degree in Computer Engineering from the same institution. Her main research area is multimedia security. She is also interested in the security issues of biomedical signals like ECG and EEG. She has been published many research papers in different international conferences and journals. She worked as a research assistant in the department of information security at United Arab Emirates University, UAE.

Muhammed Izharuddin did his B.Sc. Engg. and M.Sc. Engg. and Ph.D from Z. H. College of Engineering and Technology AMU, Aligarh, India, in 1988, 1991 and 2014, respectively. He worked for two years as a Hardware Engineer in Morsad Computer, in Oman, till 1997. In 1997 he joined the Department of Electronics Engineering at AMU, Aligarh, as an Assistant Professor. Currently, he is Associate Professor in the Department of Computer Engineering, AMU Aligarh 202,002. His main area of research interests is low-power VLSI design (ASIC and FPGA based Design) for signal processing specifically security and surveillance applications and security of Medical signals, other area of interest is Device Modelling. His worked are published in various reputed journals. He is a member of Institution of Electrical & Electronics Engineers (IEEE) (U.S.A.) and Institute of Electronics and Telecommunication Engineers, IETE, India.