v-CSS: A Video Encryption Algorithm Based on Conversion, Shuffling and Substitution using Randomly Generated Grayscale Image

[1]Kanagaraj Narayanasamy, [2]Padmapriya Arumugam
[1]Research Scholar, [2]Associate Professor, Alagappa University, Karaikudi, Tamilnadu, India.
[1]kanagaraj.n.in@ieee.org, [2]mailtopadhu@yahoo.co.in

Abstract **With the enormous development of network and communication technology, the need of video encryption algorithms have more research focus from the past decade. Privacy protection of videos is a significant concern because of the sensitiveness information resides in the video. In this research paper, a new video encryption methodology has been proposed which is based on Conversion, Shuffling and Substitution using a randomly generated grayscale image. Cryptanalysis and Brute force attack have been done to assess the strength. The reliability of the proposed image and video algorithms are assessed through the results of Histograms, Mean values, Entropy measure, Peak Signal-to-Noise Ratio (PSNR), Mean Square Error (MSE), Number of Pixels Change Ratio (NPCR), and Unified Average Changing Intensity (UACI). The results reveal that the video algorithm is good enough to withstand attacks like statistical, differential and other attacks.**

Keywords —Video Encryption, Pixel Transformation, Substitution, Shuffling

I. INTRODUCTION

Videos are widely used in this digital Era. Each and every online users are sharing their videos enormously through internet. Videos are usually personal, they may be related to the factors; broadcast, post production, medical, public safety, and defense. So, privacy has to be preserved. The shared videos need to be secured from the preying eyes. Just like encrypting text or image, it is possible to encrypt the videos to avoid the unintended or unauthorized access.

To keep the video safe, it can be protected or encrypted or both. The encryption of a video can be done in two ways. The first way is to encrypt the whole video, and the second way is to encrypt the Region of Interest (RoI). Traditional cryptographic algorithms were modified slightly to handle video data, example, AES, DES and RSA. However, the traditional algorithms are not the best to handle videos efficiently. Due to the high computational cost those are not suitable to use for the same purpose.

Traditional image encryption algorithms such as private key encryption standards (DES and AES), public key standards such as Rivest Shamir Adleman (RSA), and the family of Elliptic-Curve-based encryption (ECC), as well as the International Data Encryption Algorithm (IDEA), may not be the most desirable candidates for image encryption, especially for fast and real-time communication applications [2].

An object-based unequal encryption method for H.264 compressed surveillance videos was proposed in [10]. Based on the H.264 bit sensitivity analysis, the bits with the highest sensitivity are encrypted by AES. It is noted that AES is not suitable for videos encryption due to its high computational cost.

A Layered Cellular Automata (LCA) is a highly parallel system, not only has the inherent advantages of traditional CA but also has more complex and flexible neighborhood structures. LCA is firstly introduced in [6] and [1] by which to generate pseudo random sequence. Then LCA has been applied to design block cipher along with reversible CA in [4]. It has been found that the proposed block cipher is better than AES with respect to the requirements of confusion and diffusion. Video contain frames [7]. The quality of the video is based on the number of frames per second (*fps*). If the fps is high, the video will be legibly played. The frames are basically made up of pixels. In this work, a new video Encryption Algorithm is introduced which uses the substitution and shuffling techniques. Pixel values (R/G/B) are converted to binary form and then binary values undergo the process of shuffling and substitution. These phases make the Encryption Algorithm tough to break.

MPEG4 (MP4) and MKV are the two popular video containers. MP4 contains H.264 video codec and AAC for audio codec. In our work, MP4 is taken to the experimental study.

 DOI : 10.18231/2454-9150.2019.0097

In this research paper, second section describes about the proposed video encryption methodology. The next sections consist of algorithm and work flow to explain deeply about the proposed encryption. In the fifth and sixth section, the experimental study, and results and discussion are presented in the respective sections. In the next section consists of performance evaluation of proposed algorithm. In the final section, the conclusion of the proposed methodology is elaborated.

II. PROPOSED METHODOLOGY

The proposed Encryption Algorithm (*v-CSS*) consists of four phases. They are Key Generation, Conversion, Shuffling and Substitution. The overall process of the proposed methodology is shown in the below *figure 1*. These phases are elaborated in this section.

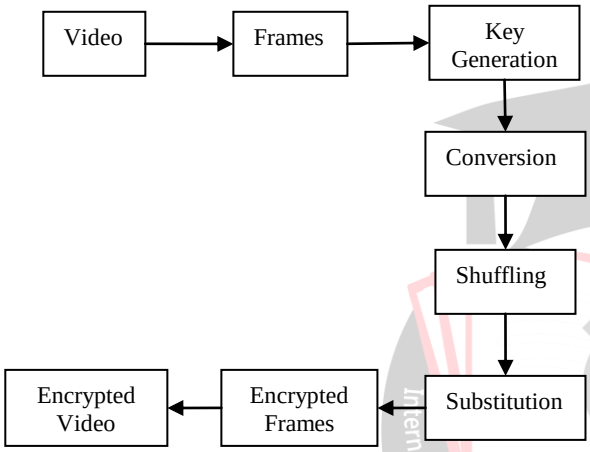


Figure 1 Overall processes of v-CSS

A. *Key Generation*

In the first phase, a gray scale image (*GSi*) is randomly generated according to the size of the input frame of the video. The Key (*K*) will be generated from the *GSi* using a random value called 'Chosen Value (*CV*)'. The result of (Sum of digits of *K*) mod 10 will be prefixed to K and its binary value (BK_{Sh}) will be used for shuffle process.

B. *Conversion Process*

In the second phase, the pixel values are converted to binary form, that is, bits. Pixel values of red, green and blue band falls between 0 and 255. Values are converted as 8 binary digits as shown in *figure 2*.

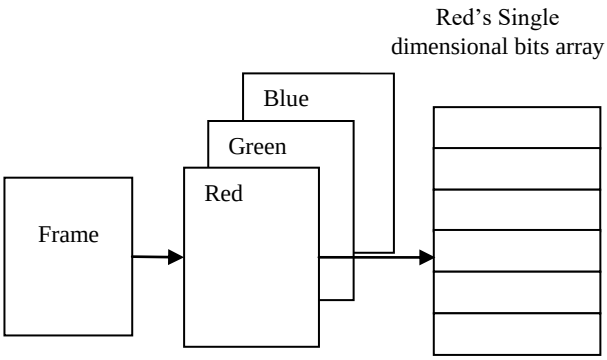


Figure 2 Process of conversion of frame to bits

C. *Shuffling Process*

In the third phase, the binary digits are shuffled in even or odd order (*figure 3(a)*) which will be based on the BK_{Sh}. If the count of one's in BK_{Sh} is zero, the odd order shuffling will be chosen; otherwise the even order shuffling will be chosen. In the odd order shuffling (*figure 3(b)*), the first, third, fifth and seventh bits are moved to third, fifth, seventh and first respectively. In the even order shuffling (*figure 3(c)*), the second, fourth, sixth and eighth bits are moved to fourth, sixth, eighth and second respectively. After the shuffle process, the bits are converted to decimal values.

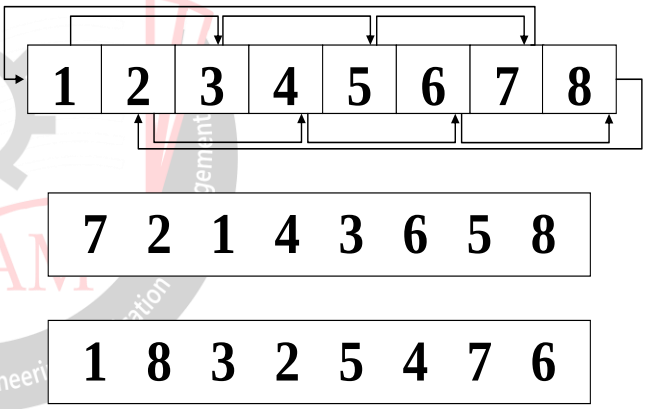


Figure 3 (a) Shuffling process (b) Odd order Shuffling and (c) Even order Shuffling

D. *Substitution Process*

In the last phase, using the *GSi*, the pixel values are encrypted using XOR operator. By index mapping method, the *GSi* XORed with shuffled pixel values to produce the encrypted frame.

In the decryption phase, the three encryption phases are processed in the order conversion, substitution and shuffling to get the decrypted video. The video is splitted into frames and then the pixel values of red, green, and blue band are retrieved. Using the received key file, the bits are substituted with the help of XOR and then the resultant values are converted into bits and then reshuffled to retrieve the original video.

 DOI : 10.18231/2454-9150.2019.0097

III. WORK FLOW

The plain video is converted into frames according to the *fps*. As per the size of the frame, a Gray Scale image (*GSi*) is generated with random values. From the *GSi*, the Chosen Value (*CV*) and key is formed and BK_{sh} is determined. Meanwhile the Frame is converted in R/G/B array and the values are converted into BITS. By using BK_{sh} the order of shuffling is determined. After the shuffling process, the BITS are substituted using *GSi* using XOR operator. After encrypting all the frames, the frames are formed to create encrypted video.

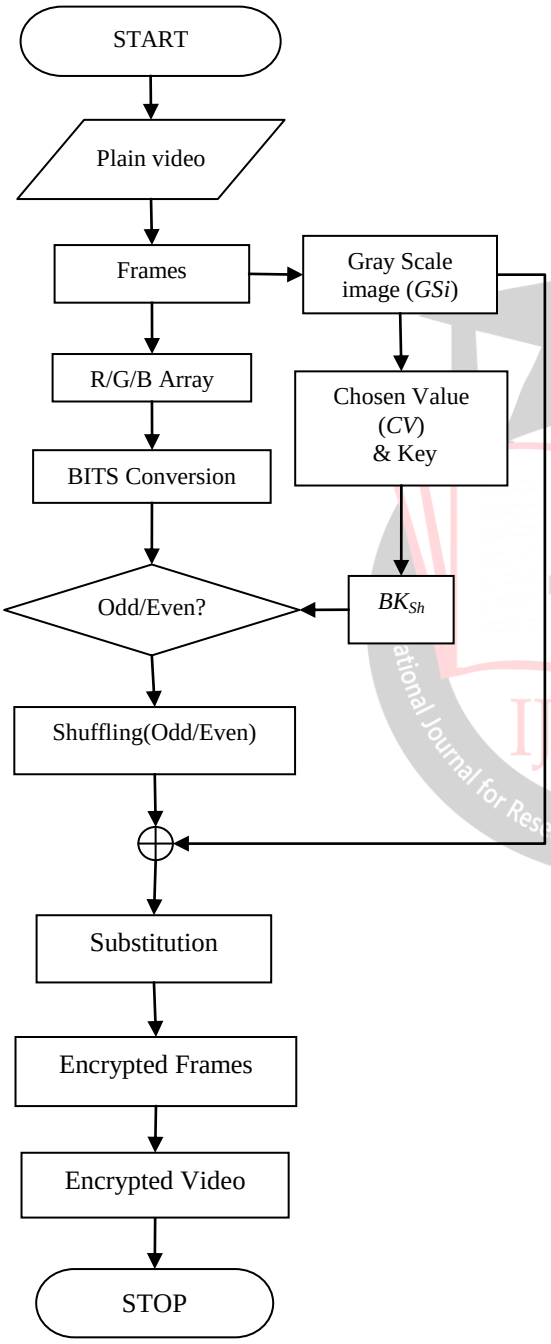


Figure 4 Work flow of *v*-CSS

IV. ALGORITHMS FOR *v-CSS*

Algorithm for encryption:

Input : Plain video (in .mpeg4 format)

Output : Encrypted video and key file (.txt format)

Step 1: Plain video splitted into frames.

Step 2: Generation of *GSi* and determine BK_{Sh} using *K* and *CV*.

Step 3: Do step 4, 5 and 6 until all the frames are encrypted

Step 4: Frame's Red/Green/Blue band values are converted to Binary form.

Step 5: Binary values are shuffled in odd order or even order according to the BK_{Sh}.

Step 6: The *GSi* is XORed with the shuffled values of each band.

Step 7: Encrypted frames are combined to form encrypted video.

Algorithm for decryption:

Input : Encrypted video (.mpeg4 format) and Key file (in .txt format)

Output : Decrypted video (in .mpeg4 format)

Step 1: Encrypted video fragmented to frames.

Step 2: Do step 3, 4 and 5 till the last encrypted frame

Step 3: Encrypted frame's band values are decrypted using *GSi*.

Step 4: The decrypted values are transformed to binary form.

Step 5: The binary values are reshuffled to retrieve the decrypted frame.

Step 6: The decrypted frames are reformed into decrypted video.

V. EXPERIMENTAL STUDY

When a video is fed into the encryption system; the video splitted into frames according to the quality of input video. Each and every frames goes through the process of conversion phase, R/G/B values are converted into eight bits as shown in the below *table 1*.

Table 1 Conversion of Pixel values into bit form

Pixel Value	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
45	0	0	1	0	1	1	0	1
70	0	1	0	0	0	1	1	0
150	1	0	0	1	0	1	1	0
210	1	1	0	1	0	0	1	0
255	1	1	1	1	1	1	1	1

The 'key' is chosen from Red or Green or Blue band of the gray scale image (*GSi*) according to a value which is called 'Chosen Value' (*CV*). *CV* is taken randomly from the frame's red value array, which is also used as 'Position Notifier'. For instance, if the frame is 128x128 pixels, then the total available pixels are 16,384. Now, from 4915 pixels which is the result of thirty percent of 16,384; a pixel will

 DOI : 10.18231/2454-9150.2019.0097

be chosen at random and that random pixel value will be the *CV*. If the *CV* falls between 0 to 9 percent, >9 to 19 percent, and >19 to 29 percent then Red, Green, and Blue value will be chosen respectively at the position of *CV*, as key (*K*).

If count of one in BK_{Sh} is odd, then the bits of pixel values are shuffled in odd order; otherwise shuffled in even order. If the value of *K* is 72, |7+2| mod 10 = 9 then 97 is obtained which is used for shuffling purpose. The binary form of 97 is '01100001' and count of one in BK_{Sh} is odd. So the values are shuffled in odd order and the result is tabulated in below *table 2*. If *K* has three digits, the least valued digit will be omitted.

After the shuffle process, the pixel values are totally irrelevant to the original. This shows the transformation of the frame to be unpredictable. But for the pixel value 255, there is no change; since all bits positions are '1'. This will be same for the pixel value 0. In the next phase, this will be eradicated by performing substitution process.

The pixel values of *GSi* of each band are XORed with pixel values of each band to form encrypted frame. Eventually makes the encrypted video. The resultant value after substitution process is shown in the *table 2*.

In summary, the videos are handled as frames. First, the frames are converted to bits, then they are shuffled in odd or even order and finally substitution takes place for better encryption purpose.

Table 2 Pixel values before and after the encryption processes

Pixel Value	Phase 2: Shuffle	Phase 3: After XOR
45	**15**	**22**
70	**196**	**221**
150	**180**	**173**
210	**240**	**233**
255	**255**	**230**

The table 2 shows the variation between original pixel value and after the last phase (Substitution phase). It shows the unpredictability of the proposed video encryption algorithm.

Since this algorithm follows the Symmetric key cryptographic technique. The key (*K*) is used for both encryption and decryption process. In the decryption phase, the value of *K* is used to decrypt the encrypted video. The two stages of the encryption are processed in the order of substitution and then shuffling to obtain the decrypted plain video.

VI. Performance Evaluation

Execution speed is a deciding factor of the applicability of the algorithm in the real world. The average encryption (*AET*) and decryption time (*ADT*) is determined using various resolution of video from 64x64 to 1024x1024 on personal computer equipped with an Intel processor (Core *i3*) with clock speed of 1.7GHz, 2GB of RAM and 520GB of Hard disk capacity and tabulated in *table 3*.

Table 3 Encryption and Decryption time of *v*-CSS.

Frame Size	*v*-CSS	
	AET (ms)	ADT (ms)
64x64	178	164
128x128	218	202
256x256	325	318
512x512	470	451
1024x1024	1023	987

VII. Results and Discussion

PSNR and MSE are the traditional image quality metrics used to assess whether the algorithms are capable of withstand the differential attacks. NPCR and UACI are the new metrics which are used in recent days. These metrics are already discussed in lot of research articles [3],[5],[8],and [9]. Histograms and mean values are used to study the ability of the proposed algorithm against statistical attacks. The results are tabulated in the below *table 4*. Various images with various resolutions are used in the study. For simple understanding purpose, the values are tabulated for the lena image with 512x512 size in *table 5*.

Table 4 Histogram and Mean values of the original image and encrypted image

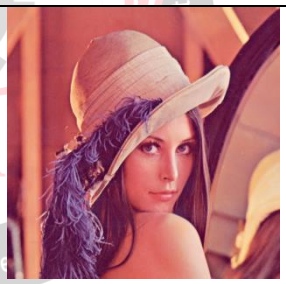
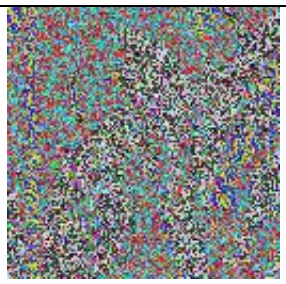
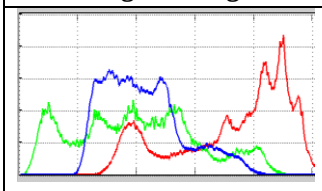
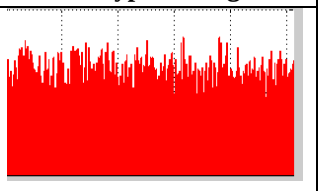
 **Original image**	 **Encrypted image**
 **Histogram**	 **Histogram**
Red : 180.22 Green : 99.05 Blue :105.41 **Mean values**	Red : 127.78 Green : 127.42 Blue : 127.20 **Mean values**

Table 5 Results of various image quality metrics

PSNR	20.2478
MSE	4.226
NPCR (%)	99.6196
UACI (%)	33.50
Entropy	7.9998

 DOI : 10.18231/2454-9150.2019.0097

VIII. CONCLUSION

Traditional cryptosystems are not suitable for encrypting video due to the high computational cost. A new symmetric key based video encryption methodology (*v-CSS*) is presented in this paper which encrypts the video with the help of shuffling and substitution techniques. These two processes satisfy both the confusion and diffusion requirements. The results of PSNR, MSE, NPCR, UACI and Entropy reveal that the encrypted video would withstand statistical and differential attacks easily and it is also able to resist brute-force (Trial and Error) attack due to high randomness in the key generation. The proposed algorithm can be used in digital archive, live stream and saved videos.

REFERENCES

[1] A. Moosavi, ‘‘Two-layer cellular automata based cryptography,’’ Trends Appl. Sci. Res., vol. 7, pp. 68–77, Jan. 2012.

[2] A. Pande, P. Mohapatra, and J. Zambreno, ‘‘Securing multimedia content using joint compression and encryption,’’ IEEE Multimedia Mag., vol. 20, no. 4, pp. 50–61, Oct. 2013.

[3] Ali Soleymani, Md Jan Nordin, and Elankovan Sundararajan,"A Chaotic Cryptosystem for Images based on Henon and Arnold Car Map", The Scientific World Journal, Volume 2014, Article ID 536930

[4] C. S. Rao, ‘‘Implementation of object oriented encryption system using layered cellular automata,’’ Int. J. Eng. Sci. Technol., vol. 3, no. 7, pp. 5786–5795, 2011.

[5] National Instruments, “Peak Signal-to-Noise Ratio as an Image Quality Metric”, White paper published by ‘National Instruments, China’ on Sep 11, 2013. Available online at http://www.ni.com/white-paper/13306/en/

[6] R. Ayanzadeh, K. Hassani, Y. Moghaddas, H. Gheiby, and S. Setayeshi, ‘‘Multi-layer cellular automata for generating normal random numbers,’’ in Proc. 18th Iranian Conf. Elect. Eng. (ICEE), 2010, pp. 495–500.

[7] Rafael C.Gonzalez, Richard E. Woods, “Digital Image Processsing” Second ed., Pearson Education, ISBN 978-8178086293.

[8] Wang, Z., Bovik, A.C., “Mean Squared Error: Love It or Leave It?”, IEEE Signal Processing Magazine ,January 2009.

[9] Wu, Y., Noonan, J.P., Agaian, S. “NPCR and UACI randomness tests for image encryption”. Cyber J., Multidiscip. J. Sci. Technol. J. Sel. Areas Telecommun., 2011 pp. 31–38

[10] Y. Zhao, L. Zhuo, N. Mao, J. Zhang, and X. Li, ‘‘An object-based unequal encryption method for H. 264 compressed surveillance videos,’’ in Proc. IEEE Int. Conf. Signal Process., Commun. Comput., Aug. 2012, pp. 419–424

 DOI : 10.18231/2454-9150.2019.0097