Novel Selective Video Encryption for H.264 Video

Dinesh Goyal*‡, Naveen Hemrajani**

* Department of Computer Science and Engineering, Suresh Gyan Vihar University, Jaipur, India

** Department of Computer Science and Engineering, JECRC University, Jaipur, India

‡ Corresponding Author; Address: Suresh Gyan Vihar University, Jaipur, India Tel: +91 141 6450389

e-mail: dinesh8dg@gmail.com

Abstract- : Internet video services has widespread application in areas such as digital television, mobile TV, video-conferencing and video on demand. Once a video stream goes beyond simple public communications, then various factors have to be considered. One important factor to consider is that data security. Selective Video Encryption has evolved in past few years because they need less time and ensures security of video content too. There has been many selective Video Encryption Techniques, which are complex and dependent on various factors like key sharing and inter and intra frame difference etc. In this paper we propose a selective video encryption algorithm, which is fast, more compact and independent of key sharing factor for H.264 video format.

Keywords- Video, Selective Encryption, Video Encoding, Key sharing, H.264.

1. Introduction

Information sent or transmitted over the public networks must have reliable protection. The protection for video streaming can be achieved by using cryptography.

Cryptography can protect video streaming in different ways. The video is subjected to encryption and decryption so that it can be read only by authorized receivers. The use of cryptography also ensures that the video reaches its destination without change (not tempered with). It verifies the identity of the communicating parties, and ensures that none of them can deny that he/she has sent or received a specific video (non-repudiation).

2. Using Cryptography to Secure Video Streaming

Security and privacy issues in multimedia technology have become an important concern. Many multimedia applications require secure transmission; the level of security required depends on the sensitivity of the information in these applications.

In order to overcome the problem of processing overhead and to meet the security requirements for real-time video applications with high quality video compression, several encryption algorithms to secure video streaming have been proposed. Most of these algorithms attempt to optimize the encryption process with respect to the encryption speed, and the display process. Some of the proposed video encryption schemes are reviewed in the section below.

2.1 Naïve Algorithm

Naïve algorithm is the most straight-forward method to encrypt every byte in the whole Moving Picture Experts Group (MPEG) (MPEG, 1988) video stream using standard encryption schemes such as DES or AES. The concept behind the Naïve algorithm is to treat the MPEG bitstream as text data and does not use any of the special structure [1] [2] [3].

Naïve algorithm ensures the security of the entire MPEG stream by using standard encryption schemes because, to date, no effective algorithm can break the encryption schemes such as AES or

triple DES. However, this algorithm cannot be applied for big video, because it is very slow, especially, when using triple DES. Because of the encryption operation, the delay increases, and overhead will be unacceptable for real-time video encryption.

2.2 Pure Permutation Algorithm

The pure permutation algorithm works by simply scrambling the bytes within a frame of an MPEG stream by permutation. It is extremely useful in a situation where the hardware decodes the video, but decryption must be done in the software.

Authors in [4] demonstrated that the pure permutation algorithm is vulnerable to known-plaintext attack, and hence, its use should be carefully considered. This is because by comparing the cipher text with the known frames, an adversary or hacker could easily figure out the secret permutation list. Once the permutation list is figured out, or becomes known, all frames could be easily decrypted. It must be noted that knowing one I-frame of an MPEG stream is enough to decrypt the permutation list, based on Shannon's Theorem.

2.3 Zig-Zag Permutation Algorithm

In the Zig-Zag permutation approach [5], instead of mapping the 8x8 block to 1x64 vector in "Zig-zag" order, it maps the individual 8x8 block to a 1x64 vector by using a random permutation list (secret key). Zig-Zag permutation algorithm consists of three main steps:

1. Generate a list of 64 permutations.

2. Complete the splitting procedure. Assume that the DC coefficient is denoted by 8- digit binary numbers then it is split into two numbers and then, the number of it is placed to DC coefficient and the number of is placed to AC coefficient. The splitting procedure is based on the following observations:

I. The value of DC coefficient is much larger than the value of AC coefficient.

II. After splitting, extra space is needed to store one of the splitting numbers, and this will increase the size of the MPEG stream. However, it must be noted that the last AC coefficient is the least important value in the block which can be set to zero with no significant visual degradation.

3. Apply the random permutation to the split block.

As mapping Zig-Zag order and mapping according to the random permutation list have the same computational complexity, the encryption and decryption processes add very little overhead to the video compression and decompression processes. However, this method decreases the video compression rate because the random permutation distorts the probability distribution of Discrete Cosine Transform (DCT) coefficients and makes the Huffman table used less than optimal. Authors at [6] introduced two types of attacks on Zig-Zag permutation, a cipher text only attack, and a known-plaintext attack.

The Zig-Zag permutation algorithm is vulnerable to the cipher text only attack, the attack relies on the fact of the statistical properties of the DCT coefficient, where non-zero AC coefficients are gathered in the upper left corner of the I-block. Statistical analysis by **[6]** counted the number of non-zero ACs and DC coefficients from all blocks in an I-frame with the following observations:

- DC coefficients always have the highest frequency of non-zero occurrences.
- The frequency of AC1 and AC2 are among the top 6.
- The frequency of AC3 to AC5 is among the top 10.

The second problem is that the Zig-Zag permutation algorithm cannot withstand the known-plaintext attack. Assuming that we know certain frames of the video in advance (known-plaintext), the secret key could easily be figured out by simply comparing the known-plaintext with the corresponding encrypted frames. To solve this problem, a method, called binary coin flipping sequence method, together with two different permutation lists, could be used. For each 8x8 block, a coin is flipped. If it is a tail, the permutation list 1 (key1) is applied to the block. If it is a head, the permutation list 2 (key2) is applied to the block. This method is vulnerable to known-plaintext attack as well, because non-zero AC

coefficients have the tendency to gather in the upper left corner of the block. Thus, it would be easy for an adversary to determine which key is used **[7]**

2.4 Selective Encryption Techniques

In order to reduce the amount of processing overhead **[8]** and to meet the security requirements for real-time video applications, selective encryption techniques have been proposed **[9]**. This scheme is aimed to encrypting different levels of selective parts of the MPEG stream by using the feature of the MPEG layered structures (e.g. encrypting all headers and I-frames, encrypting all I-frames and all I-blocks in P and B-frames). The basic selective encryption is based on the MPEG I-frame, P-frame, and B-frame structure. It encrypts the I-frame only because, conceptually, P- and B-frames are useless without knowing the corresponding I-frame.

2.4.1 Video Encryption Algorithm (VEA)

Authors at [6] suggested a new video encryption algorithm called VEA. Video encryption algorithm works upon the statistical properties of MPEG video standard and symmetric key algorithm standard to reduce the amount of data that is encrypted.

Researchers at **[10]** in have introduced four different video encryption algorithms Algorithm I, Algorithm II (VEA), Algorithm III (MVEA), and Algorithm IV (RVEA).

A. Algorithm I: First algorithm makes use of the permutation of Huffman code words in I-frames. This algorithm assimilates encryption as well as compression in a single procedure. The permutation p is the secret part of the algorithm. This secret part is extensively used in permuting the standardized MPEG Huffman code word list. Hence, in order to save the ratio of compression, only those code words having exact bit count should be permuted. This can be done only by selecting the permutation p to such that it does so. [**12**] Showed that this algorithm has higher vulnerability towards both known-plaintext attack, and cipher-text-only attack. The attacker could possibly sense and cause reconstruction of the permutation p that is kept secret, only if some of the video frames are known in advance. This can be done by analyzing and creating a comparison between the frames that are known and the frames that are having encryption done on it.

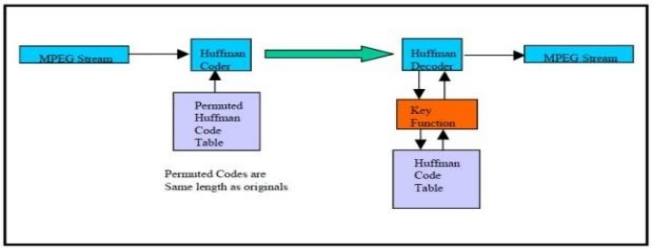

Fig. 1. Algorithm-I Huffman permuted coder [10]

B. Algorithm II (VEA): We know that the most integral information about the MPEG video is carried along by I-frames. Thus, encrypting only the necessary sign bits of the DC coefficients of the blocks of I frame is done by simply XOR-ING them with an m-bit binary key, k=k_{1},k_{2}...k_{m} that is secretly building this algorithm. Randomly changing of the sign-bit of the DC coefficients of an MPEG video stream which happens to belong to the same GOP of w_{1},w_{2}…w_{n} combined in a function is the intrinsic effect seen after applying the aforementioned technique to encrypt.

When 0 is the found value of the key bit k_{i}(mod m), a wi bit will remain unchanged. The same value will be in a state of flipping if they found out value of the key bit k_{i}(mod m) is calculated as 1. Ultimately, reusability of the secret key would be done by the upcoming GOP. This helps in resynchronizing of the values. The resynchronization capability for bit streams of video is required in the case of transmission errors. They are rewinding, and the opposite. The level of secure transmission of this scheme is dependent vitally on the using key's length. The writers provide for the knowledge that a binary-key long enough should be used. But, key with quite large size might prove impractical as well as infeasible. But, using a key with small size, the breakage of any system might take place and could be easily turned down.

When stream of video and the size of the key is same and also is specific and only one and is being used at most one-time then that is in correspondence to Vernam cipher that is

also known with the name of one-time pad. This cipher provides complete security. But, it is not possible practically done for applications of the mass media like services of Video on Demand and other like applications. Although, when the size of the key is quite small, the complete technique is simplified to known as Vigenere-like cipher.

C. **Algorithm III (MVEA):** In [11], the authors have made an improvement to the Algorithm II (VEA). A lot of advancement has been incorporated into the Algorithm-II (VEA). The sign bits of the differential values of DC coefficient as well as motion vectors in P-frames and B-frames be encrypted using XORs using the confidential key, in-place of encrypting just the sign bits of DC coefficient in I-frame block. Such kind of enhancement makes a video playback further much random as well as much more non-viewable.

The position of the motion vector gets altered as soon as the sign bits of the differential values of the motion vector are altered. Along with this there is also an observable alteration in the magnitude of the motion vector which makes or leads to creation of the entire video being a lot chaotic or hectic. It has also been found during studies that the encryption of sign bits of motion vectors causes to the encryption of sign bits of DCT coefficient in B- as well as P- frames useless or to say redundant.

Further, the Algorithm-III (MVEA) was made in order of encrypting only the sign bits of DC coefficients in the I-pictures of sequence of video in MPEG standard, although it leaves the coefficients of AC not encrypted. Doing this makes an increment in risks factors of secure transmission but with significance it deteriorates the overhead of computation.

Namely, because the DC coefficient and the sum of all AC coefficients within the block are related, an attacker might make usage of the non-encrypted AC coefficients to make out the unknown DC coefficients that are already in encrypted form. For this treason, any of the application the authors recommend encrypting all DCT coefficients in the I-frames for applications that need higher level of security. However this type of improvement makes the video playback more random and more obscure (non-visual). With similar conduct of VEA, this algorithm, i.e. MVEA is dependent on the size related with the secret key.

The Algorithm-III (MVEA) heavily relies on the m-bit secret key, k just the same is done in the Algorithm-II (VEA). Moreover, the task of resynchronizing the bits of key is done just in the starting of any GOP. Unluckily, the basic issues related with secure transmission relevant to VEA are again also applicable to MVEA.

D. **Algorithm IV (RVEA):** The concept of Algorithm-IV (RVEA) was recommended by [11]. The difference between Algorithm-IV (RVEA) and that to Algorithm-III (MVEA) is that Algorithm-IV (RVEA) makes use of a traditional symmetric key cryptography for encrypting the sign bits of DCT coefficient and the sign bits of motion vectors. The process of encryption is speeded up by the algorithm by only encrypting definite or specific sign bits in MPEG stream. Henceforth we can very well say that this particular algorithm is far superior in comparison to the previous mentioned three algorithms that is to say Algorithm-I, Algorithm-II (VEA), and Algorithm-III (MVEA) in parlance of security. In the Algorithm-IV (RVEA), the sign bits of DCT coefficients and motion vectors are merely pulled out from the MPEG video sequence, and encrypted by means of a fast conventional cryptosystem such as AES, and after this it is then reinstated back to its original position in the encrypted form. The outcome of this is alike to VEA/MVEA whereby the sign bits are made to either reverse or left as it is without making any changes. The number of bits for encryption is restricted to at most 64 by the authors, for each MPEG stream macro block, with the vital and ultimate intention of plummeting and bounding the time taken for computation. Subsequent, we define exactly as to how these sign bits are carefully chosen for encryption.

2.5 Other Selective Encryption Algorithms

Selective encryption intends to encrypt only some parts of the entire bit stream to reduce the

overall computational requirement, and hence the cost, introduced by encrypting large volumes of video data stream in a limited period of time. That is the multimedia data stream to protect *P* is partitioned into subsets: P_A and P_B, $P = P_A \cup P_B$, where P_A is the subset to be encrypted while *P B* is left in the clear.

$$C = E\{select_K Enc(P)\} = E_K\{Enc(P_A)\} \quad (1)$$

For instance, the I-frames or I-frames plus the I-blocks in P and B frames of a MPEG video are encrypted. Another simple light weighted algorithm is to encrypt only the sign bits and leave the rest in the clear.

In order to reduce the amount of processing overhead and to meet the security for real time video applications, selective encryption techniques have been proposed. The idea of this scheme is to encrypt different levels of selective parts of MPEG stream by using the feature of MPEG layered structures (e.g. encrypting all headers and I frames, encrypting all I frames and all I blocks in P and B frames).

1) *AEGIS, (Encrypt I-Frame Only)*: **[13]** have introduced a new secure MPEG video mechanism called Aegis. Aegis method encrypts only the I-frame of all MPEG groups of frames in MPEG video stream, while B-frame and P-frame are left unencrypted.

2) *Sign-Bit of DCT Coefficients*: **[10]** used a secret key to transform the sign bits of the DCT coefficients of MPEG video data. The secret key (k_{1}, k_{2}, k_{3},….., k_{2m}) is randomly generated with length 2m, where the number of key and the length of key is not limited. If the sign bits of DC and AC coefficients are represented by S, (s1, s2, s3,., Sm), then the encrypted data is Ek(Si) = bi xor si of length 2m. The encryption operation randomly changes the sign bits of DCT coefficients.

3) *Byte-Encryption:* [14], have proposed to randomly encrypt bytes in an MPEG stream for free distribution, while the original bytes at the corresponding positions are transferred in encrypted form to legitimate users. This is actually equivalent to encrypting byte at random positions.

3. Summary and Motivation

In previous section we have discussed many techniques of Video Security and one can conclude many theories which can be described as follows:

- Video Security (Encryption) has been a serious issue for real time or on demand video services especially in this modern era of high bandwidth data connectivity and Ubiquitous devices with high storage capacity.
- Video Encryption has evolved from conventional cryptographic techniques using symmetric key or asymmetric key cryptography to now a day's popular selective video encryption.
- All of them are more or less dependent on key sharing amongst end user and their encryption quality varies with the length of the key used.
- All these video encryption techniques especially selective video encryption though provide good encryption but at the same time they involve high cost of key sharing and encryption time.

4. Proposed Selective Video Encryption and Decryption Model

As discussed in previous sections the existing video encryption technologies though evolved over a long period still have many issues of cost of encryption and key sharing and speed of encryption. Thus we require a model of video encryption which encrypts the video selectively but at the same time it reduces the burden of key sharing and takes lesser time.

Here we propose a method of Video Encryption which is free from key sharing by the end user, which is fast and selective. The major features or working of the encryption and decryption processes is as follows:

A. Encryption Process

The encryption process can be under stood in the simpler way in the given flow chart, which can be illustrated as follows:

a. I frame is not Encrypted of the video only the P and B Frames are Encrypted.
b. I frame is used for generating the key for Encryption of the remaining video
c. I frame behaves as a key for encryption process of later frames.

d. In case of P and B frames only those frames are encrypt where motion Vector (M) is not Zero. Or in other words the Motion Vector is encrypted only which helps in compression of the video too.
e. If Motion Vector is not zero then the key is XORed with the Motion Vector to provide selective encryption of the same.
f. Then the encrypted video frame sequences are buffered in a pre defined sequence.
g. The encrypted video s then compressed to ensure high speed communication to the client.
h. The encrypted video can further be compressed or communicated to the receiver as per the need of the user.

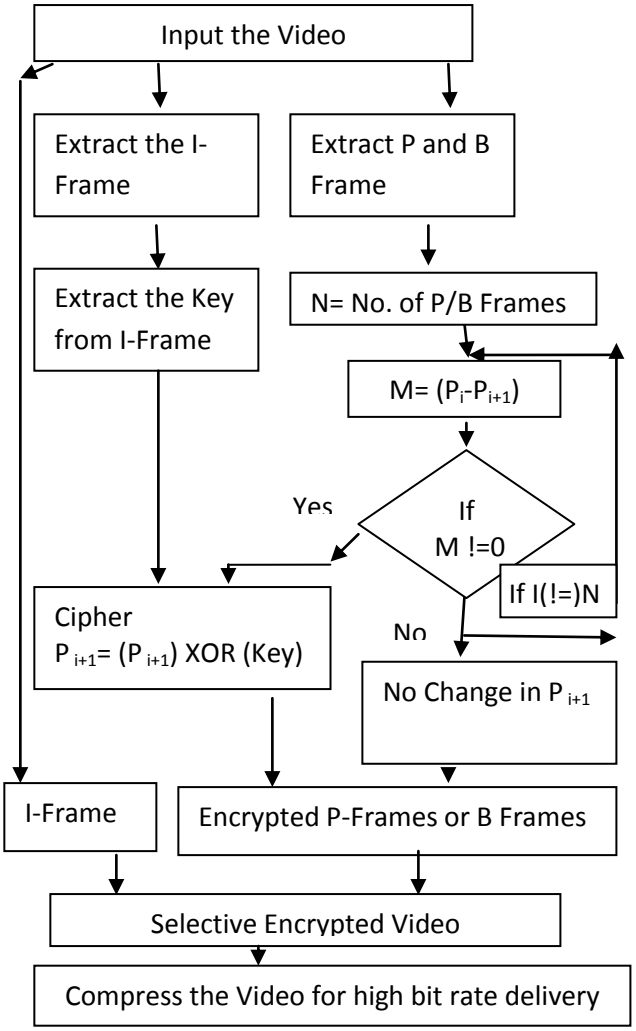

Fig. 2. Video Encryption Process

B. Decryption Process

The decryption process can be under stood in the simpler way in the given flow chart, which can be illustrated as follows:

a. User receives the encrypted video.
b. The I-Frame and P and B frames are extracted for decryption of video.
c. I frame behaves as a key for encryption process of later frames.
d. In case of P and B frames only those frames are decrypted where motion Vector (M) is not Zero.
e. If Motion Vector is not zero then the key is XORed with the Motion Vector to provide decryption of the same.
f. Then the decrypted video frame sequences are buffered in a pre defined sequence.
g. The encrypted video is then compressed to ensure high speed communication to the client.

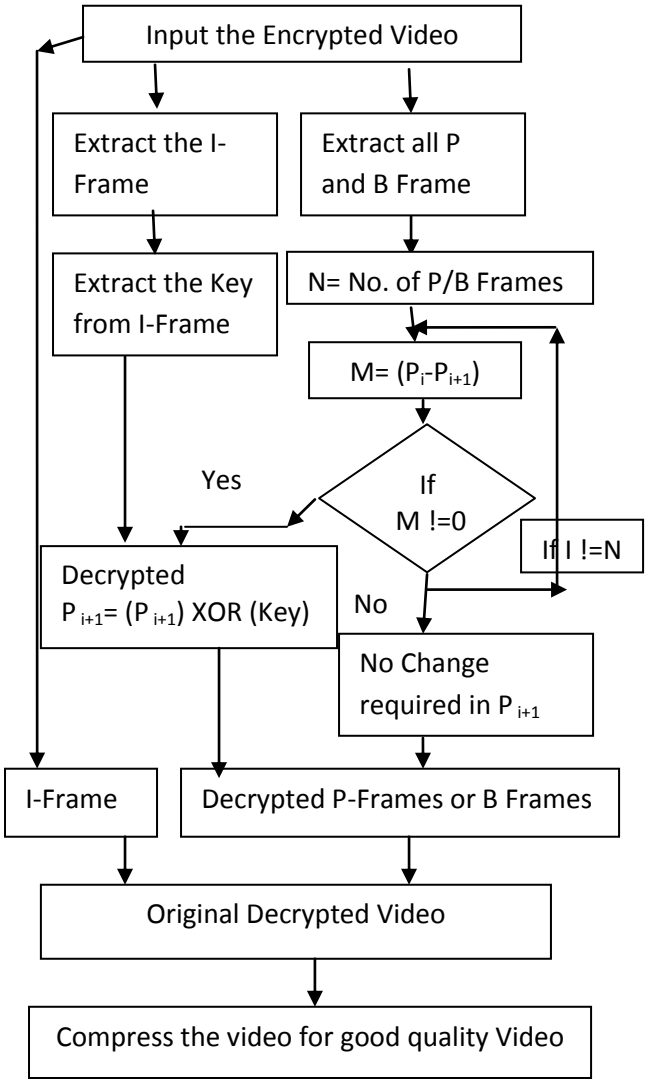

Fig. 3. Video Decryption Process

5. Analysis of Proposed Model

We analyse the performance of encryption and decryption of two different video formats and frame sizes and find out the quality and speed of encryption and decryption of the same. We analyse them on below given two formulas:

Encryption Rate=no.of FrameX size/time (2)
Bitwise encryption rate=size of video/time (3)

a. AVI Video Input

In this case we input uncompressed AVI video of 352 frames of 480X640 pixels of 2.39 MB

Fig. 4. Original Video Frames

Fig. 5. Encrypted Video Frames

Fig. 6. Decrypted Video Frames

b. H.264 Video Format

In this case we input a compressed H.264 video of 313 frames of 240X320 frame size of 3.36 MB.

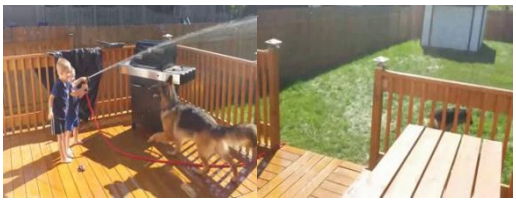

Fig. 7. Original Video Frames

Fig. 8. Encrypted Video Frames

Fig. 9. Decrypted Video Frames

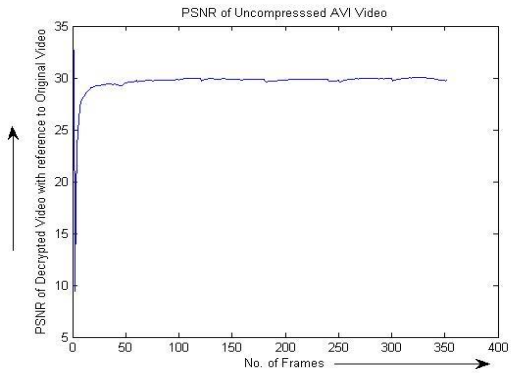

Fig. 10. PSNR values of AVI decrypted video with reference to Original Video

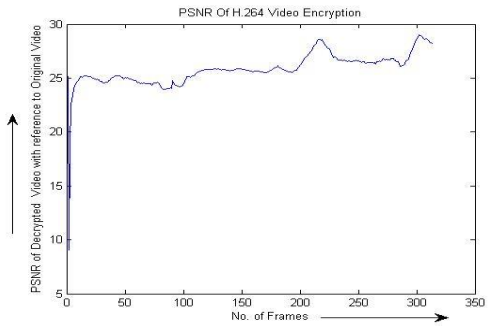

Fig. 11. PSNR values of H.264 decrypted video with reference to Original Video

Table 1. Comparison of Performance of both the videos in terms of speed

	AVI Video	H.264 Video
No. Of Frames	352	313
Size of Frames	480*640	240*320
Encryption Time (sec)	88.128099	26.005758
Decryption Time (sec)	75.670624	24.725279
Pixels Encryption Rate (Pixels/Sec)	1227013.872	924349.138
Pixels Decryption Rate (Pixels/Sec)	1429014.245	972219.565
Bitwise encryption rate (KB/Sec)	27.7705	132.303
Bitwise decryption rate (KB/Sec)	80.2467	50.9457

The graph in figure 12 compares the numbers of pixels of AVI and H.264 video being encrypted in a second during the process of encryption. Here more number of pixels of the AVI video is encrypted in a second than that of H.264 Video.

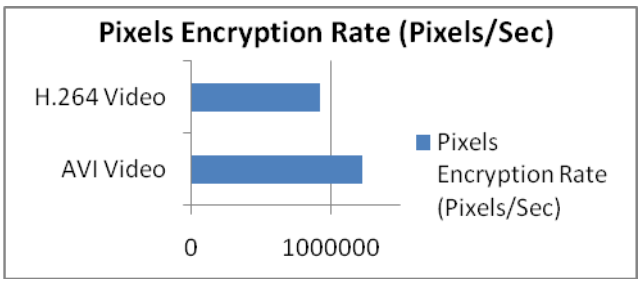

Fig. 12. Comparison of Pixel wise encryption rates

The graph in figure 13 compares the numbers of pixels of AVI and H.264 video being decrypted in a second during the process of decryption. Here more number of pixels of the AVI video is decrypted in a second than that of H.264 Video.

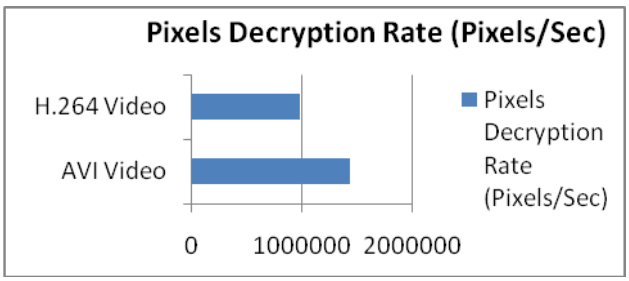

Fig. 13. Comparison of Pixel wise decryption rates

The graph in figure 14 compares the numbers of bytes of AVI and H.264 video being encrypted in a second during the process of encryption. Here less number of bytes of the AVI video is encrypted in a second than that of H.264 Video.

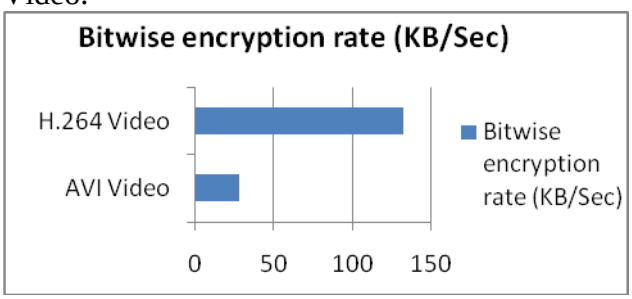

Fig. 14. Comparison of Bitwise encryption rates

The graph in figure 15 compares the numbers of bytes of AVI and H.264 video being decrypted in a second during the process of decryption. Here more number of pixels of the AVI video is decrypted in a second than that of H.264 Video but at the same time video in H.264 is compressed too much during process of encryption, thus decryption also involves time for extraction of all the frames.

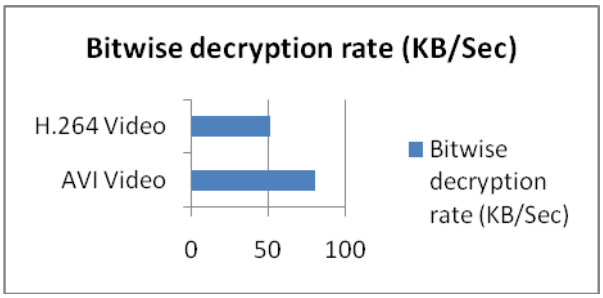

Fig. 15. Comparison of Bitwise decryption rates

6. Comparison of Proposed Model with Existing MVEA Algorithm

Now we will like to analyze the performance of the proposed technique with the one of the most popular existing technique namely MVEA, which has been explained prior to this.

The tables and graph given below will quantify the performance and efficiency of over algorithm. Here the same video, AVI and H.264 used earlier are encrypted using MVEA and proposed technique.

Table 2. Comparison of Performance of AVI Video for MVEA and proposed Technique

	AVI Video	
Encryption Type Applied	Proposed Technique	MVEA Encryption
No. Of Frames	352	352
Size of Frames	480*640	480*640
Encryption Time (sec)	88.128099	91.771253
Decryption Time (sec)	75.670624	82.193215
Pixels Encryption Rate (Pixels/Sec)	1227013.872	1178303.624
Pixels Decryption Rate (Pixels/Sec)	1429014.245	1315612.244
Bitwise encryption rate (KB/Sec)	27.7705	24.12135
Bitwise decryption rate (KB/Sec)	80.2467	75.74258

Table 3. Comparison of Performance of H.264 Video for MVEA and proposed Technique

	H.264 Video	
Type of Encryption Applied	Proposed Technique	MVEA Encryption
No. Of Frames	313	313
Size of Frames	240*320	240*320
Encryption Time (sec)	26.005758	31.94112
Decryption Time (sec)	24.725279	28.19415
Pixels Encryption Rate (Pixels/Sec)	924349.138	752584.756
Pixels Decryption Rate (Pixels/Sec)	972219.565	852602.4016
Bitwise encryption rate (KB/Sec)	132.303	124.798
Bitwise decryption rate (KB/Sec)	50.9457	45.1043

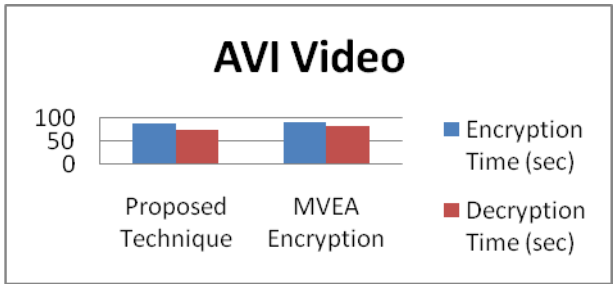

Fig. 16. Comparison of Encryption time of MVEA and Proposed Technique

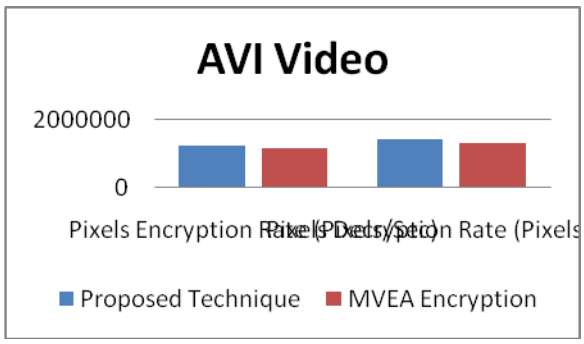

Fig. 17. Comparison of Pixel wise Encryption and Decryption Rate of MVEA and Proposed Technique

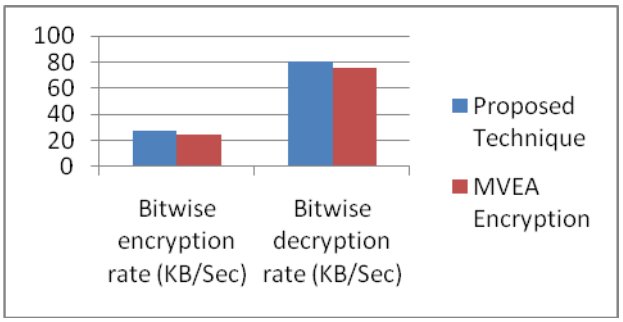

Fig. 18. Comparison of Bit wise Encryption and Decryption Rate of MVEA and Proposed Technique

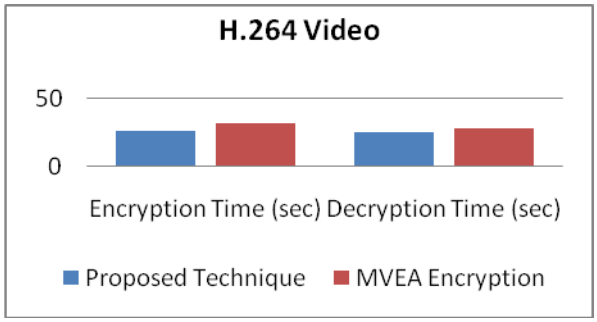

Fig. 19. Comparison of Encryption time of MVEA and Proposed Technique

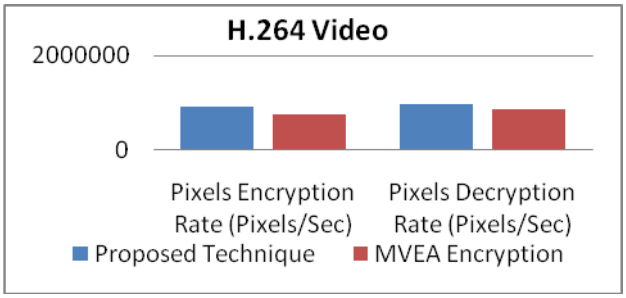

Fig. 20. Comparison of Pixel wise Encryption and Decryption Rate of MVEA and Proposed Technique

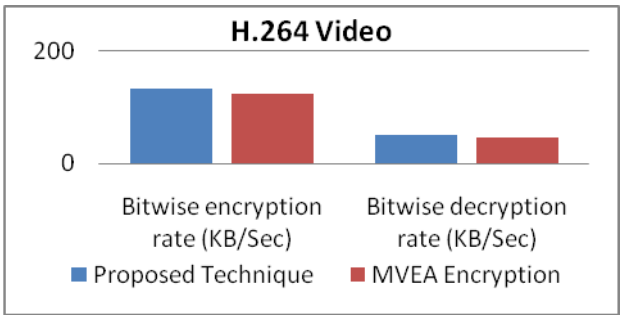

Fig. 21. Comparison of Bit wise Encryption and Decryption Rate of MVEA and Proposed Technique

7. Conclusion

In the above work we have proposed a technique of selective video encryption which is thoroughly key independent and fast. The analysis of above is performed on two different types of video formats one is compressed one and other one is uncompressed of different frame sizes and almost similar no. of video frames. By this analysis it can be proven that the encryption performed, by using I frame as key reduces the responsibility of key sharing.

The encryption and decryption time of compressed video is quite less, whilst the same for the uncompressed video is quite more though the frame size of uncompressed video size is high but the encryption rate of compressed video is almost 6 times of uncompressed video.

In terms of quality of video quality after decryption for both of the video formats is almost similar though it is varying more in case of H.264 format because of compression of video too but the PSNR are almost of 30 db, thus proving that quality of video is also maintained in both the cases.

Next we compared the performance of proposed technique with one of the most popular selective video encryption technique called MVEA and as the result (tables and graphs) shows the speed of encryption for proposed technique is much faster than that of MVEA Algorithm both for AVI and H.264 Video.

7.1 Future Work

In future using key sharing, independent video encryption technique can further be more enhanced with complex encryption techniques, also the researchers may attempt to use zig-zag and selective encryption to ensure fast and secure video encryption.

References

[1]. I. Agi , L. Gong, An Empirical Study of Secure MPEG Video Transmissions, Proceedings of the 1996 Symposium on Network and Distributed System Security (SNDSS '96), p.137, February 22-23, 1996

[2]. A. Salah, A Light-Weight Encrypting For Real Time Video Transmission. Retrieved Nov 2008, 22, from http://www. cdm. depaul.edu/legacy/checksite. aspx?oldUrl=http://www.cdm.depaul. edu/research/ Documents/ TechnicalReports/2004/TR04-002.Pdf

[3]. M. Habib and T. P. Mee Encryption of MPEG Video Streams. 2006 IEEE Region 10 Conference TENCON 2006. 1-4. Hong Kong, China.

[4]. J.A. Slagell. Known-Plaintext Attack Against a Permutation Based VideoEncryption Algorithm. Available http://eprint.iacr.org/2004/011.pdf .

[5]. L. Tang, For encrypting and decrypting MPEG video data e±ciently. in Proceedings of The Fourth ACM International Multimedia Conference (ACM Multimedia'96), (Bosten, MA). 219-230.

[6]. L. Qiao and K. Nahrstedt. A new algorithm for MPEG video encryption. In Proc. of First International Conference on Imaging Science System and Technology, pages 21–29,1997.

[7]. T. Lookabaugh et al., Selective encryption of MPEG-2 video, in Proceedings of the SPIE Multimedia Systems and Applications VI, (Orlando, FL), September 2003.

[8]. D.W. Redmill, M. Nithin, D. Lefol A novel secure H.264 transcoder using selective encryption. IEEE 2007. IV-85 - IV-88.

[9]. L. Qiao and K. Nahrstedt, "Comparison of MPEG encryption algorithms", Int. J. Comput. Graph., Special Issue on Data Security in Image Communications and Networks, vol. 22, no. 3, 1998

[10]. C. Shi., S.Y. Wang, and B. Bhargava, "MPEG Video Encryption in Real-Time Using Secret key Cryptography," 1999 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA'99), Las Vegas, NV, June 28 - July 1, 1999

[11]. B. Bhargava, C. Shi and S. Y. Wang "MPEG video encryption algorithms", *Multimedia Tools Appl.*, vol. 24, no. 1, pp.57 - 79 2004

[12]. T. Seidel, D. Socek, and M. Sramka, "Cryptanalysis of Video Encryption Algorithms ," to appear in Proceedings of The 3rd Central European Conference on Cryptology TATRACRYPT 2003, Bratislava, Slovak Republic, 2003.

[13]. G.A. Spanos, and T.B. Maples, "Performance Study of a Selective Encryption Scheme for the Security of Networked, Real-time Video," Proceedings of 4th International Conference on Computer Communications and Networks, Las Vegas, NV, September 20-23, 1995.

[14]. C. Shi, and B. Bhargava, "A Fast MPEG Video Encryption Algorithm," Proceedings of the 6th International Multimedia Conference, Bristol, UK, September 12-16, 1998.

[15]. C. Griwotz, O. Merkel, J. Dittmann, and R. Steinmetz, Protecting VOD the easier way," in Proceedings of Multimedia and Security Workshop at the 6th ACM International Multimedia Conference, (Bristol, England), pp. 21-28, 1998.

[16]. S.R. Ely, MPEG Video coding- A simple Introduction in EBU Technical Review Winter 1995 Ely.

[17]. R.Schafer and T. Sikora, "Digital Video Coding Standards and Their Role in Video Communications", Proceedings of the IEEE Vol. 83, pp. 907-923,1995.

[18]. T.Sikora, "MPEG Digital Video Coding Standards",In Digital Electronics Consumer Handbook, McGraw Hill Company,

[19]. T.Sikora, "The MPEG-1 and MPEG-2 Digital Video Coding Standards",IEEE Signal Processing Magazine, to be published

[20]. Liu, X., and Eskicioglu, A.M., Selective Encryption of Multimedia Content in Distributed Networks: Challenges and New Directions, presented at IASTED International

Conference on Communications, Internet and Information Technology(CIIT 2003), Scottsdale, AZ, 2003

[21]. Cheng, H., and Li. X., Partial encryption of compressed images and video, IEEE Trans, Signal Process., 48(8), 2439-2451, 2000

[22]. Meyer, J. and Gadegast, F., Security Mechanisms for Multimedia Data with the Example MPEG-1 Video, Project Description of SECMPEG, Technical University of Berlin, 1995

[23]. Alattar, AM Al-Regib, G.l., and Al-Semari, S.A., improved Selective Encryption Techniques for Secure Transmission of MPEG Video Bitstreams, in Proceedings of the 1999 international Conference on image Processing (ICIP '99), Kobe, 1999, Vol. 4. pp. 256-260.