A New Approach for Hiding Data Using B-box

Dr. Saad Abdual azize AL_ani
Associate Prof. Computer Science Department
Al_Mamon University College
Baghdad, Iraq

Bilal Sadeq Obaid Obaid
Associated Researcher Computer Science (MSc)
Amman Arab University
Amman, Jordan

***Abstract*—Digital Images and video encryption play an important role in today's multimedia world. Many encryption schemes have been proposed to provide a security for digital images. This paper designs an efficient cryptosystem for video. Our method can achieve two goals; the first goal is to design a height security for hiding a data in video, the second goal is to design a computational complexity cryptosystem**.

***Keywords*—ASCII; Binary; Cryptosystem; Decimal; Decryption; Encryption; Image; Plaintext; Video**

I. INTRODUCTION

The grand challenges of the data that is carried and stored over the network is making the data safe and disclosed to illegal users. The use of computer networks - especially during the last decade – has grown dramatically. For this reason, creating and developing the security systems and encryption techniques should take wide focus in the field of information security [2]. Parallel with the information transition evolution time, systems and techniques have big success against the threats, but still there are faults [1]. One of these techniques is using the image and multimedia encryption, because the digital image is becoming important carrier of information for people [3]. With the advance of information security requirement, the encryption technology of digital image is applied widely on multimedia communications. Conventional encryption arithmetic could be used to develop or initiate new techniques for best specification and satisfactions. Because some encryption algorithms still used where there there disadvantages and faults are included, such as the structure complexity, the secret key singleness and the encryption speed slowly, and it is difficult to satisfy the encryption requirement of the image that has lots of data. So using the conventional encryption solely is not enough [4]. Anarchic mapping idea could be applied into video to encrypt, because it has the sensitivity and big qualifications of initial values and the randomness [6]. The method which is adopting combined conventional encryption technology. Complex anarchic mapping can overcome the single conventional encryption's disadvantage effectively. The pixels' values in original image of video frames can be changed ultimately via encrypting, in order to realize the aim of encryption [5].

For powerful and advanced quality of encryption effectively, the method of position scrambling can be used before and during encrypting steps [5]. The classical algorithms are Arnold cat map, affine transformation, magic square transformation, and knight-tour transformation, etc. Through these transformations, the change of the image pixels' position can be realized by keeping the secret of parameters and by normalizing heavy complex random boxes and numbers and with iteration times to reach the aim of encryption. A method which based on Arnold cat map and S-DES has been proposed and developed strongly and efficiently in this paper by applying the particular character of logistic anarchic map. The key numbers of S-DES are increased and the key can be changed in real-time [7]. The experiments results have shown that the good methods that integrate encryption/decryption process on the image or video produce well security and fast executing with no visible contorting or reconstructing on the image

II. THE METHOD

The method of encryption process divides into two sections. First section is to encrypt the data. The second is to augment the encrypted data on a video file.

A. Encryption Algorithm

With first section step, we will initiate the input size of Plaintext such 8 – character and handle such a binary (0s, 1s).

1) Choose eight characters as Plaintext such one block.

2) Calculate the weight of the plaintext characters by subtracting 64 from the ASCII code of each character

*3) Convert the weights to binary mode with 6 bits. Here, output includes 48 binary bits (8char * 6bits).*

4) Build the B – Table, where B – table which consists of 4 rows and 15 columns. The table made up by generating a particular function to generate the number 15 randomly from 1 to 15, where the function will be generated for four times.

5) For each 6 bits from the 48 bit that referred to the selected Plaintext, consider first and last bits as row number of the B – table that have been evaluated, and every 4 bits in between is for the column number.

6) From B – table, take value of the cell that have been matched by (Row no, Column no).

7) Convert it to its 4 bits of binary mode. Here, the result will be changed and shrank from 48 bits to 32 bits.

8) Generate a particular function specialized for generating random numbers to build P – table.

9) Broadcast the 32 bits in P – table, where P – table used to reshape the sequence of the 32 bits depending on P – table numbers, these numbers are the assumed new location of the 32 bits.

10) Write the output row by row to get new 32 bits. The output is the encrypted data bits and the next steps for augmenting the encrypted data into video file

11) Initiate a free object of video file to prepare for constructing a new video and giving the name of the object.

12) Define the video that should be encrypted.

13) Find the number of frames of the defined video.

14) Find the dimensions of video frames that defined.

15) Create a figure that would hold the video frames.

16) Configure the created figure shape, dimensions and location to be adjusted with appearance of video frames properties.

17) Generate a random number by applying a particular algorithm, select random number from the interval [1, number of the video frame that we have inputted].

18) Sequentially, read the data of video frame and save it on the memory.

19) Present the frame data as image onto a special figure.

20) Create frame by getting the data that is presented onto the figure.

21) Find a frame which holds the random number of the video frames.

22) Convert the frame data type to image data type.

23) Find the dimension of the converted image.

*24) Hag the image array to sub - image arrays, where each sub –image size is 8*8. With considering image is colored, that mean it's three dimensions (R, G, B), so it's run over the sub – image dimensions (8*8*3).*

25) Record the location of each sub-image array depending on the image.

26) Calculate the number of sub – images that have been produced.

27) Apply the particular random number generator from the interval [1, number of sub – image].

28) Pick-up the sub – image array that holds the random number.

29) Calculate the values (minimum, mean, maximum) of sub-image array pixels from arrays (red, green, blue) respectively.

30) Conserve the locations of those values from sub-image array.

31) Convert the values from decimal to binary mode.

32) Bring first three bits of encrypted data bits and put them rather last three binary bits of minimum value, and next three bits of encrypted data bits rather than last three binary bits of mean value, and next three bits of encrypted data bits key rather the last three binary bits of maximum.

33) Convert each changed binary back to decimal mode.

34) Carry each value as new pixels back to the origin locations of the new values (locations of min, mean and max) of sub-image array which is the converted image.

35) Convert the converted image back to frame data type.

36) Fulfill the video object file by adding frame by frame to the video object file that has been constructed, respectively, depending on the frame order from that came out of the inputted video file.

37) Close the figure that prints the data of frames to initiate the new figure to present the next frame to add them onto video object file.

38) Generate special algorithm on the video file that is done to remove – for some time – some delays or to increase of video size that could occur.

Here, the algorithm has finished encrypting the data and hiding it inside a video file by a very strong algorithm that aims to hide and save data or to maintain property rights.

B. Decrytion Algorithm

Now, the output is the encrypted video will be decrypted by the inverse algorithm to testify and return the original video – that has been encrypted - file back.

1) Define the video that should be decrypted.

2) Find the number of frames of the defined video.

3) Find the dimensions of video frames that defined.

4) Create a figure whose would hold the video frames.

5) Configure the shape of created figure, dimensions and location to be adjusted with appearance of video frames properties.

6) By applying a particular algorithm to generate a random number, select random number from the interval [1, number of the video frame that we have inputted].

7) Sequentially, read the data of video frame and save it in the memory.

8) Present the frame data as image of a special figure.

9) Create frame by getting the data that presented onto the figure.

10) Find a frame which holds the random number of the video frames.

11) Convert the frame data type to image data type

12) Find the dimension of the converted image

*13) Divide the image array to sub - image arrays, where each sub –image size is 8*8. With considering image as colored, that mean three dimensions (R, G, B), so it's run over the sub – image dimensions (8*8*3).*

14) Record the location of each sub-image array depending on the image.

15) Calculate the number of sub – images that have been produced.

16) Apply the particular random number of generator from the interval [1, number of sub – image].

17) Pick-up the sub – image array that holds the random number

18) Calculate the values (minimum, mean, maximum) of sub-image array pixels from arrays (red, green, blue) respectively.

19) Conserve the locations of those values from sub-image array

20) Pickup each LSB from minimum, maximum, mean of sub image.

21) Convert the values from binary mode to decimal.

22) Broadcasting the bits of decrypted data array on the P^{-1} table, where the P^{-1} is evaluated by special generation

numbers which were generated by complex arithmetic algorithm.

23) Divide the output to 4-bit, convert to decimal number.

24) Search each decimal number in b-table column by column until matching, the intersection of row and column.

25) Align the bits for each address of the cells; where the 2-bit of row addresses of each cell that will be located between the 4-bit of column.

26) Convert these 6 bits decimal number.

27) Add threshold (64), then convert it to ASCII character.

With showing these characters as one line, the output is the original data.

C. Video Encryption Implementation

THE FOLLOWING FIGURE SHOWS HOW DOES THE METHOD ENCRYPT THE VIDEO BRIEFLY CONCLUSION

In this paper before hiding data in audio, convert the 8 characters to 48 bits, compression the output to 32 bits by using B-table, broadcast in P-table to get different location for bits. Hiding the 32 bit in random frame and random pixel will be difficult to attack the data, and also a smallest space is used in hiding data that each 8 character is in 5 pixels.

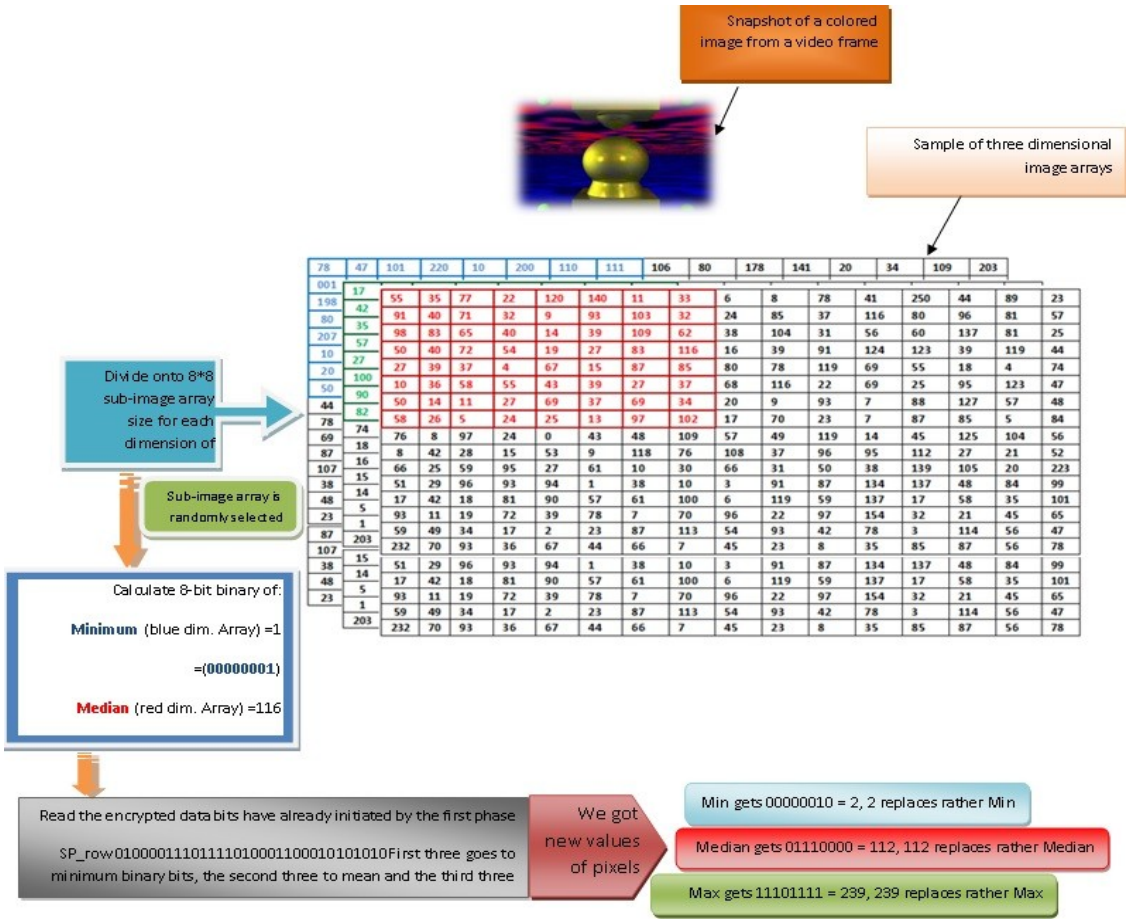

Fig. 1. Video encryption flow

REFERENCES

[1] Fridrich J 1998 Symmetric ciphers based on two-dimensional chaotic maps J. Bifurcat Chaos 8 1259-84

[2] Scharinger J 1998 Fast encryption of image data using chaotic kolmogorov flows J.Electron Imageing **7** 318-325

[3] Li. Shujun, X. Zheng "Cryptanalysis of a chaotic imageencryption method," Inst. of Image Process. Xi'anvJiaotong Univ., Shaanxi, This paper appears in: Circuits and Systems, ISCAS 2002.

[4] Li S J, Zheng X, Mou X and Cai Y 2002 Chaotic encryption scheme for real-time digital video Proc SPIE on Electronic Imaging 4666 149-166

[5] W. Lee, T. Chen and C. Chieh Lee, "Improvement of an encryption scheme for binary images," Journal of Information and Technology. Vol. 2(2), 2003, pp. 191-200.

[6] I. Ozturk, I.Sogukpinar, "Analysis and comparison of image encryption algorithm," Journal of transactions on engineering, computing and technology December, vol. 3, pp.38, 2004.

[7] X Y Yu, J Zhang ,H E Ren ,G S Xu and X Y Luo, " Chaotic Image Scrambling Algorithm Based on S-DES ", International Symposium on Instrumentation Science and Technology- Journal of Physics: Conference Series 48 (2006) 349–353\