ISSN XXXX XXXX © 2018 IJESC

Research Article Volume 8 Issue No.4

Real Time Video Encryption for Secure Multimedia Transfer: A Novel Approach

Neha Dilkash[1], Anku Gupta[2], Arpita Jain[3]
B.Tech Student[1, 2, 3]
Department of Computer Science and Engineering
Galgotias College of Engineering and Technology, Greater Noida, India

Abstract:
Advances in digital content transmission have increased in the past few years. Security of multimedia data is an imperative issue because of fast evolution digital data exchanges over an unsecure network. Image and video processing applications are becoming widely used in many domains including industrials, medical imaging, manufacturing, and security systems. Real time image and video processing is a very demanding task as it needs to perform high computations for a big amount of data represented by the video, and the complex operations, which may need to be performed on the video. Currently, video encryption is important to ensure its confidentiality during its transmission on insecure networks or its storage. Many algorithms have been suggested for the purpose of video encryption but, almost all of them have some or the other security flaw. In this paper we propose a novel approach for real time video encryption by using pixel encryption. Pixel encryption is performed by shuffling and manipulating pixel values. The positions of the pixel values are changed according to a random sequence generated. At the receiver end decryption is performed by first de shuffling the pixel values and then performing reverse manipulation of individual pixel values to get the original video file.

Key words: key exchange, Pixel shuffling, RGB pixel displacement, Video decryption, Video encryption.

I. INTRODUCTION

The use of multimedia files for the purpose of communication is increasing rapidly .Video transmission from sender to receiver should be in a secure way. Video surveillance, video conferencing, digital video broadcast, distance learning these are some of the day to day life widely used applications using video content. Encryption is technique which can be used to make secure transmission of videos. The data becomes unreadable when it is encrypted. Till now many encryption algorithms have been proposed like RSA, Diffie Hellman, DES, AES etc. Most of these algorithms have been used for text encryption and using these algorithms directly for video encryption is not feasible. This is because of the use of large volume of video data and the need of real time operations. The main purpose of our approach for video encryption is to provide security as well as reduced time complexity. Many methods are being used nowadays for video encryption, which includes Permutation encryption, Selective encryption, fully encryption, RGB displacement etc. In this paper we propose a method for video encryption that is based on shuffling of pixel values, encrypted key exchange and RGB displacement. Section II contains the Literature Survey of our work. The proposed work and result analysis is discussed in Section III. Lastly conclusion is given in Section IV.

II. LITERATURE SURVEY

RGB Color Model

The name of this color model comes from the initials of the three additive primary colors, red, green, and blue. The RGB color model is a color model in which red, green, and blue light are added together in a way to produce a broad array of colors.

Three colored light beams, each of red, green and blue must be superimposed, for forming a color with RGB. Each of the three beams is called a component of that color, and each of them can have an arbitrary intensity, from fully off to fully on. In computers, the component values are often stored as integer numbers in the range of 0 to 255, the range that a single 8-bit byte can offer. These are represented as either decimal or hexadecimal numbers

Permutation Based Encryption For Video Files

Permutation Based approach for video encryption [3]-[4] is a technique in which the contents of the video file are shuffled. This is done so as to make it difficult for the adversary to know the contents of the video file. Shuffling can be applied to either some or whole part of the file, depending on the algorithm used. The permutation list serves as key in most algorithms. At the receiver end this key is used to retrieve the original file. Therefore it is important to provide for the security of this key as well.

Fully Layered Encryption

Fully Layered Encryption is a technique in which the whole video content is first compressed and then encryption is performed on it using AES or DES algorithms. This technique is not possible for real time encryption because of increased time complexity. At the same time, there can also be a loss of video quality because of compression.

Selective Encryption Technique

Selective Encryption Technique overcomes the drawback of the fully layered approach. In this technique, instead of encrypting the whole video content, selected content is only encrypted .This reduces the time taken to perform encryption and at the same time also provide sufficient security.

RGB Pixel Displacement

In the RGB pixel displacement technique [1],[7], the numerical value of the RGB pixel is displaced from its original position. In this method there is no change in the original size of the file on encryption. At time of decryption, reverse displacement of the pixel values is done to get original file.

III. PROPOSED WORK

In this paper we have discussed a hybrid approach for video encryption which uses shuffling of pixel values and RGB pixel manipulation.

Algorithm for Encryption

In the first phase, the frames from the video are extracted as it is being recorder using a webcam. The video captured is in AVI format. Fetch the resolution of the first frame (image) captured. Now, since our objective is to permute the pixel values of each frame of the video, we need a random sequence. The random sequence should be equal to the number of elements in a single frame. Therefore, a random sequence is generated. The next step is pixel displacement. This would provide a partial encryption to the video file and it also serves as an added layer of security. An RGB image stored in MATLAB has an m×n×3 data array, each of which defines red, green, blue color components for every pixel. Graphics file formats store RGB images as 24 bit images. Each of the Red , Green and Blue component are of 8 bit. Their values range from 0 to 255. In order, to manipulate each pixel value a median of the range (0 to 255) is calculated so that we can scale up or scale down the pixel values according to the median, as is shown in Figure. All the pixel values less than median are added to 128.Similarly, median is subtracted from all the pixel values greater than or equal to median. Such that the manipulated image pixel values remains in the range (0 to 255).The final values as shown in Figure2. It is essential to choose a median for scaling because otherwise, during decryption this would cause loss of original pixel values.

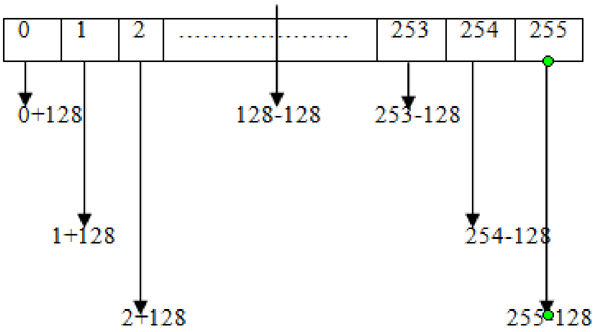

Figure 1

128	129	....	255	0	1	....	126	127

Figure 2

Figure.3. Original frame

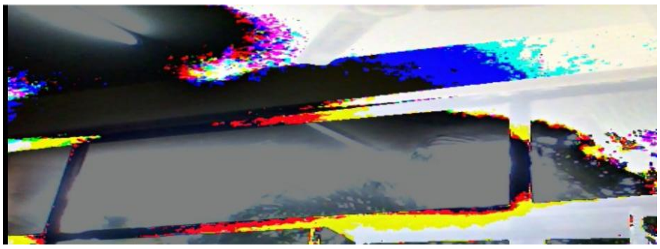

Figure .4. Frame after rgb pixel manipulation

The above phase alone is not enough in terms of security, as it involves shifting by a fixed value and also that value can be guessed easily by experts. Therefore, to increase security in the next phase, we are using the random sequence generated earlier to shuffle the position of the pixel values. Each frame after encryption is added to the video file.

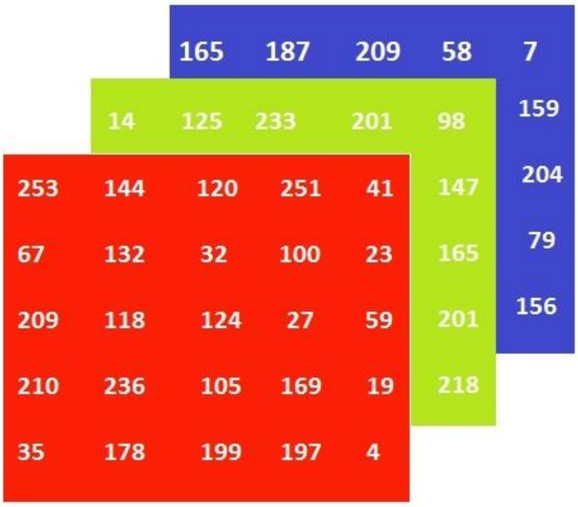

Figure. 5. Original matrix

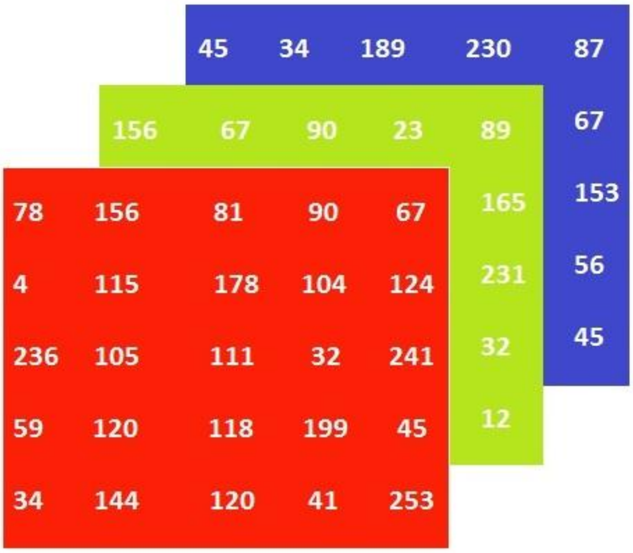

Figure.6. Shuffled matrix

Pseudo Code for Encryption

```
Let N be the number of frames in the video.
while j<N
  img=getsnapshot(vid);
  img_size=size(img);
  img1=zeros(img_size(1),img_size(2),img_size(3));
  if(j==0)
    idx=randperm(numel(img));
  end;
  for a = 1:img_size(1)
    for b=1:img_size(2)
      for c=1:img_size(3)
        img1(a,b,c) = img(a,b,c);
      end;
    end;
  end;
  for a =1:img_size(1)
    for b=1:img_size(2)
      for c=1:img_size(3)
        if(img1(a,b,c)>=128)
          img1(a,b,c)=img1(a,b,c)-128;
        else if(img1(a,b,c)<128)
              img1(a,b,c)=img1(a,b,c) +128;
              end;
        end;
        img(a,b,c)= img1(a,b,c);
       end;
     end;
   end;
  shuffled_im=reshape(img(idx),size(img));
  aviobj=addframe(aviobj,shuffled_im,shuffled_im);
    j=j+1;
end;
```

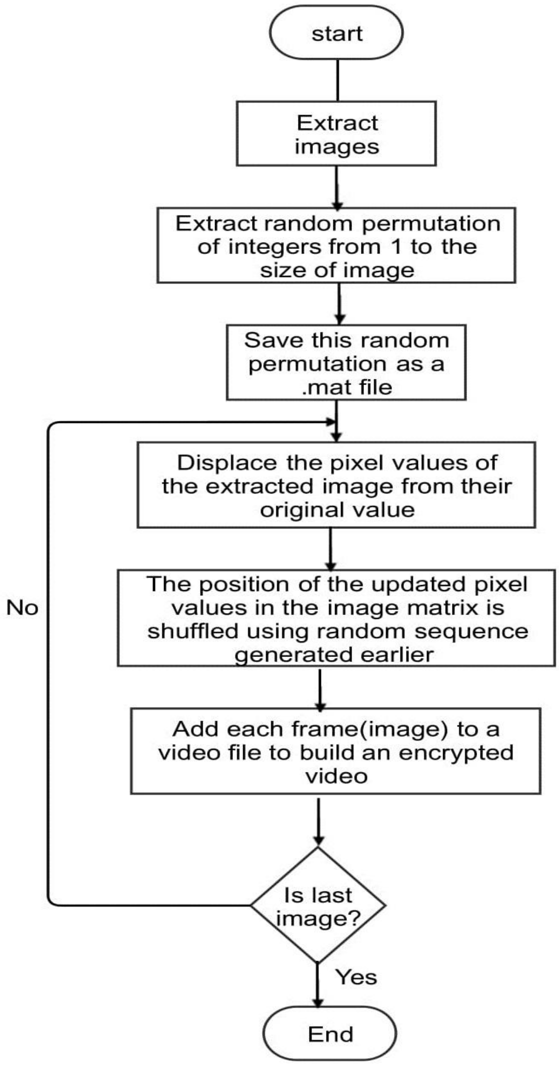
start
Extract images
Extract random permutation of integers from 1 to the size of image
Save this random permutation as a .mat file
Displace the pixel values of the extracted image from their original value
The position of the updated pixel values in the image matrix is shuffled using random sequence generated earlier
Add each frame(image) to a video file to build an encrypted video
Is last image?
No
Yes
End

Figure.7. Flow chart for Encryption

Key Exchange

The shuffled sequence in which the frames are permutated is saved to a file. This file is to be sent to the receiver, which he would use to get the original video. But, if an intruder gets the file then the security of this method would be compromised. To counter this drawback , before sending this file, it also must be encrypted. At the same time the cost of encrypting this file should be as less as possible. Hence , we have a new approach to do the same. The size of the file is equal to the size of image, a finite set of random numbers are generated, we have divided the file in to the number of subset(of fixed range) by using random number as starting indices of each of these subsets. The random numbers used are appended to the file. At last the position of the pixel values are shuffled with in their subsets. In order to perform shuffling there could be many approaches for example swapping of values at even and odd indices, simply reverse the order of elements, swapping adjacent elements. It is to be noted that greater is the size of subset more will be the security.

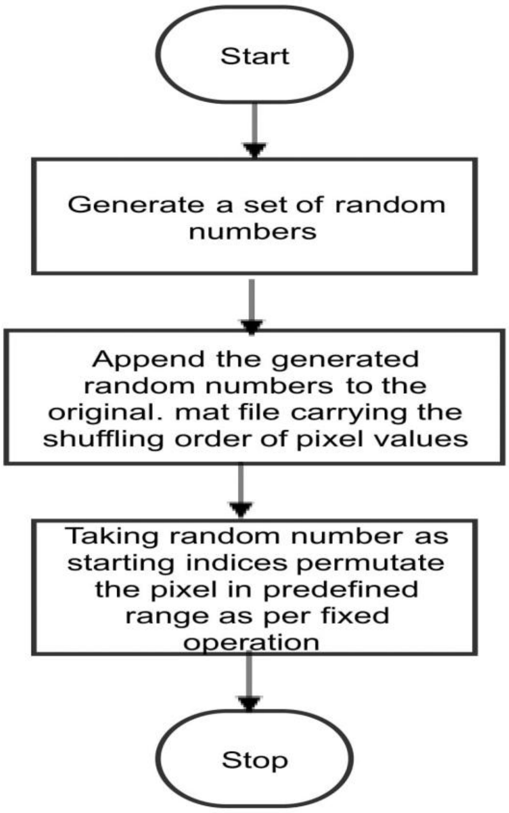
Start
Generate a set of random numbers
Append the generated random numbers to the original. mat file carrying the shuffling order of pixel values
Taking random number as starting indices permutate the pixel in predefined range as per fixed operation
Stop

Figure.8. Flow chart for key exchange

Algorithm for Decryption

The receiver receives the encrypted video file along with the encrypted key. The received key is decrypted by performing reverse operations uses in the phase of key encryption. With the help of this decrypted key, the encrypted video is shuffled again to retrieve the video, whose pixels are still displaced. Now to get the decrypted video reverse displacement of pixel values is performed.

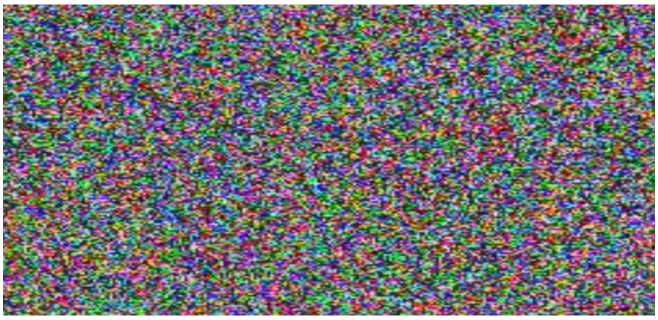

Figure.9. Encrypted frame

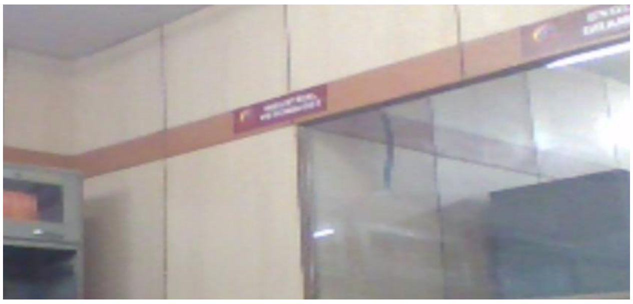

Figure.10. Decrypted frame

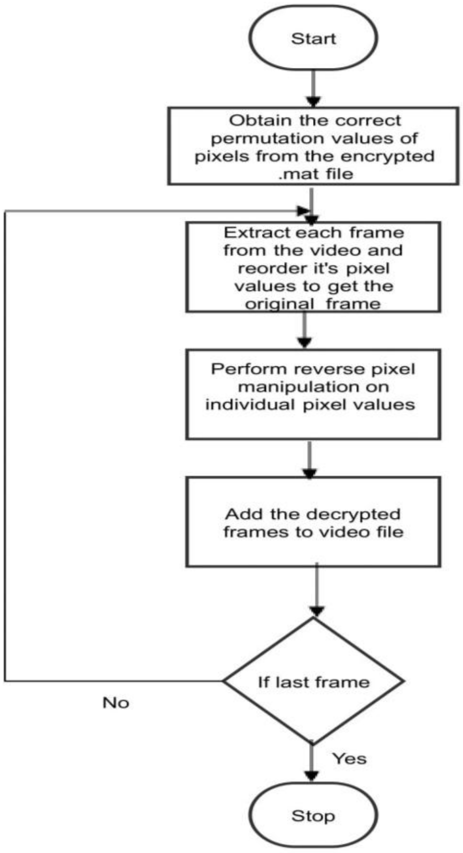
Start
Obtain the correct permutation values of pixels from the encrypted .mat file
Extract each frame from the video and reorder it's pixel values to get the original frame
Perform reverse pixel manipulation on individual pixel values
Add the decrypted frames to video file
If last frame
No
Yes
Stop

Figure.10. Flow chart for decryption

IV. RESULT ANALYSIS

The proposed algorithm has been implemented in MATLAB. A video was recorded using webcam in AVI format simultaneously, in the background the encryption algorithm was run and the video was successfully encrypted. The proposed algorithm is better in terms of performance as compared to the approach in which the whole video file is encrypted using popular algorithms like AES. Our algorithm also provides better security as compared to existing Permutation techniques. The table I gives a comparison between our proposed algorithm and the presently used techniques.

Table .1.

Methodology	**Security Level**	**Speed**	**Suitable for Real time encryption**
Fully Layered	High	Slow	NO
Selective Encryption	Low	Fast	NO
Proposed Algorithm	High	Very Fast	YES

V. CONCLUSION

In this work we have discussed the importance of security in video transmission. Encryption improves transmission confidentiality of the video data. The proposed algorithm is suitable for encrypting real time video, as it provides better security in less computational time. Also, our approach is appropriate for videos of different size, types and different applications.

VI. REFERENCES

[1]. Shrija Somaraj and Mohammed Ali Hussain,"A Novel Image Encryption Technique using RGB" 2016 IEEE 6th International Conference on Advanced Computing.

[2]. Quis-Alphetsi Keste , Laurent Nana and Anca Christine Pascau ,"A Novel Encryption technique for securing digital images in the cloud using AES and RGB pixel displacement"2013 IEEE DOI 10.1109/EMS.2013.51

[3]. Ajay Kulkarni,Saurabh Kulkarni,Ketki Haridas and Aniket More,"Proposed Video Encryption Algorithm v/s Other Existing Algorithms: A Comparative Study", International Journal of Computer Applications (0975 – 8887) Volume 65– No.1, March 2013

[4]. Pradeep Pai T and Raghu ME, Ravishankar K C ,"Video encryption for secure multimedia transmission - a layered approach", 2014 IEEE DOI 10.1109

[5]. Mohammed A. Saleh, Nooritawati Md. Tahir, Ezril Hisham & Habibah Hashim,"An Analysis and Comparison for Popular Video Encryption Algorithm" 2015,IEEE.

[6]. Qiuhua Wang, Xingjuan Wang," A new selective video encryption algorithm for H.264 Standard", 2014, IEEE

[7].Quist-Aphetsi Kester, Koudjo M Koumadi," Cryptographie technique for image encryption based on the RGB pixel displacement ", DOI: 10.1109/ ICASTech .2012. 6381069, 2012, IEEE

[8]. Sesha Pallavi Indrakanti, P.S.Avadhani, " Permutation based ImageEncryption Technique",Volume 28– No.8, August 2011, IJCA

[9]. Goel, Amnesh & Chandra, Nidhi. (2012). A Technique for Image Encryption with Combination of Pixel Rearrangement Scheme Based On Sorting Group-Wise Of RGB Values and Explosive Inter-Pixel Displacement. International Journal of Image, Graphics and Signal Processing. 4. 10.5815/ ijigsp. 2012. 02.03.

[10]. Quist-Aphetsi Kester, "Image Encryption based on the RGB PIXEL Transposition and Shuffling", DOI: 10.5815/ ijcnis.2013.07.05,2013, IJCNIS