International Journal of Security and Its Applications
Vol. 10, No. 10 (2016) pp.67-76
http://dx.doi.org/10.14257/ijsia.2016.10.10.08

A General Encryption Algorithm for Different Format Videos

Hao Li, Cheng Yang, Jiayin Tian and Jianbo Liu

School of Information Engineering, Communication University of China, Beijing 100024, China;
muyue_8710@163.com, cafeeyang@163.com, 34tianjiayin@163.com

Abstract

With the rapid development of social network, more and more information is being presented in the form of multimedia, so the security of multimedia files, especially video files, has been capturing the attention of the researchers. Video encryption is widely applied to the DRM system, video conference, medical and military fields and so on. A general video encryption algorithm is proposed to solve the problem that the existing multiple formats (compression formats and container formats) of video requires a unified encryption scheme. Meanwhile, the proposed scheme can not only provide variable levels of security, but also have good performance of compression efficiency and computational complexity, which can ensure that the encrypted video data can be operation.

Keywords*: video, selective encryption, general, security, compression efficiency, computational complexity*

1. Introduction

The selective encryption has been used to encrypt video for many years. Researchers usually focus on how to reduce the amount of encrypted data. It seems to be a good way to analyze the video structure to find encrypted data, such as I frame, slice and DCT coefficients. However, the detail analysis will bring the poor compatibility, which is a big problem for the video operators to protect a number of videos. A unified encryption schemes which can handle videos with different compressed formats and package formats, and keep the encrypted videos can be played in order to urge the users to buy the decrypted videos is necessary.

There are six performance parameters to evaluate and compare video encryption algorithms: encryption ratio, compression efficiency, degradation, security, format compliance, and speed. [1] We mainly deal with the existing video, so we need to discuss its four aspects: security, compression efficiency, computational complexity and operability.

The Aegis mechanism proposed by George was the early selective encryption schemes in 1995. The schemes chose I frames and the MPEG video sequence header to encrypt. [2] Qiao divides every 128-byte stream segment into two groups in order to reduce the encryption computational work. [3] A selective encryption based on video monitoring system was proposed by Deepti C. Gavankar. They developed a secure real time video monitoring system, which compresses and selectively encrypts streaming video to enhance the security. [4] However, because of the changes of the video internal structure, the encrypted videos cannot be played by the common video player.

Tang tries to achieve compression and encryption in one step with minimum overhead to the encoding procedure.[5] The selective region encryption proposed by Richard also needs to segment an image to its constituent regions using some edge detection algorithms.[6] But the two above methods are not suit for the existing compressed videos.

The modified RVEA video encryption algorithm only can handle the video based on MPEG-2.[7] The SSS method solves the problem that the video streaming system can be

ISSN: 1738-9976 IJSIA

able to stream video to heterogeneous clients over time-varying communication links. [8][9] But using a unified approach for different formats videos to segment each image into tiles is not mentioned in the papers.

Singh showed a study report on various video encryption algorithms and presented a survey of over 15 research papers dealing with video encryption and decryption techniques in 2014.[10] However, many existing videos which need to be protected may have kinds of compression formats and container formats, so it is difficult to use one of the above methods to process all types of videos. In order to solve this problem, a general video encryption scheme is proposed to seek a unified method to encrypt the videos in this paper. The detail of the proposed algorithm is described in Section 2. Section 3, and we will discuss the performance of the proposed algorithm from four aspects. Which includes security, compression efficiency, computational complexity and the operability. After the discussion, we can get the conclusion in Section 4.

2. The Proposed General Video Encryption Algorithm

In this section, we will introduce the proposed algorithm in detail. Many other algorithms discussed in the past, especially the selective encryption, use the function bytes, which are based on the video format, to find the key information and then to encrypt them. It is limited for the algorithms to process the only one special video format. However, there are many existing videos which may have kinds of video formats, so it is necessary to find a general encryption algorithm in order to protect the videos with different formats.

As we can see in the Figure 1, the proposed algorithm is mainly divided into three parts: unpacking, encryption, packing. The details will be declared in the following text.

We unpack the video with a certain format in Part 1. In this part, we use FFMPEG [11] to solve the problem that it is difficult for a encryption algorithm to deal with the different data structures in the different videos. There are 5 steps in this part:

Step 1: analyze the input video format at first;
Step 2: create a format container according the analysis results in the step 1 called input containers;
Step 3: use the function of FFMPEG to read one video frame and write the data into the container;
Step 4: determine whether the read frame needs to be encrypted according the input selected encryption parameter;
Step 5: select the frame payload if the frame is need to be processed, put the selected payload into Part 2. Otherwise bring the read frame to Part 3 directly.

In Part 2, we prepare the exact data which need to be encrypted, and display the encryption.

Step 6: divide the frame payload data into group by the standard of each group containing 16 bytes;
Step 7: get the content key from the Key Management Center, and encrypt the data use AES.

We gather the processed frames and generate the encrypted video file in the Part 3. The main steps are as follow:

Step 8: create the out format container according to the input container called output container;
Step 9: fill the encrypted data into the output container;
Step 10: write the video file with the content of the output container or the input container.

Back to the step 3, the algorithm dose not loop the following execution until the last input video frame is processed. We regard the frame as the unit to encrypt the video.

Because the frame is the outermost data structure in the video, it is easy to control the encryption in depth and save the time that is used to parse the further video structure, such as slices, DCT coefficients and motion vectors.

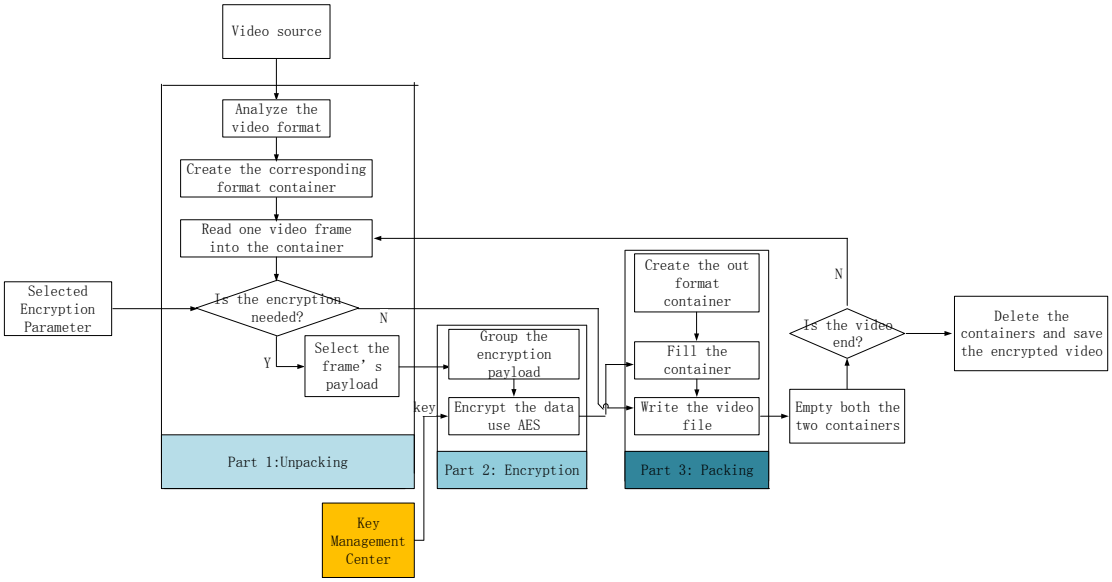


Figure 1. The Encrypting Processes in the Proposed Algorithm

3. Discussion

We can discuss the encryption algorithm from the following four aspects: security, compression efficiency, computational complexity and operability. And we also implement the detail experiment and analysis to support that our proposed algorithm has the capability of encrypting the videos with different formats, costs low power and has a good real-time performance.

3.1. Security

Security is the basic demand for an encryption algorithm. In our algorithm, we provide different levels of security. When the video is transmitted in the military communication, we can choose a higher security level. When the video is used for commercial purposes, we choose a lower security level in order to arouse the users to buy the video copyright.

The paper uses AES to encrypt the videos, as the AES has been considered to be a secure encryption algorithm. Worked with the security key transport and management protocol, it is difficult to decrypt the processed video without the correct key. The following experiment shows the encrypted video images by the different encryption depth.

We choose 0% (unencrypted), 25%, 50% and 100% encryption depth to encrypt the same video respectively. As we can see in the Table 1, the encryption depth of 25% or more can disrupt the part of the video information and affect the users viewing experience in order to induce the users to buy the video. All of the video information will be encrypted when the depth is up to 100%. This will provide the highest security services, so the eavesdropper cannot get any useful information from the encrypted video.

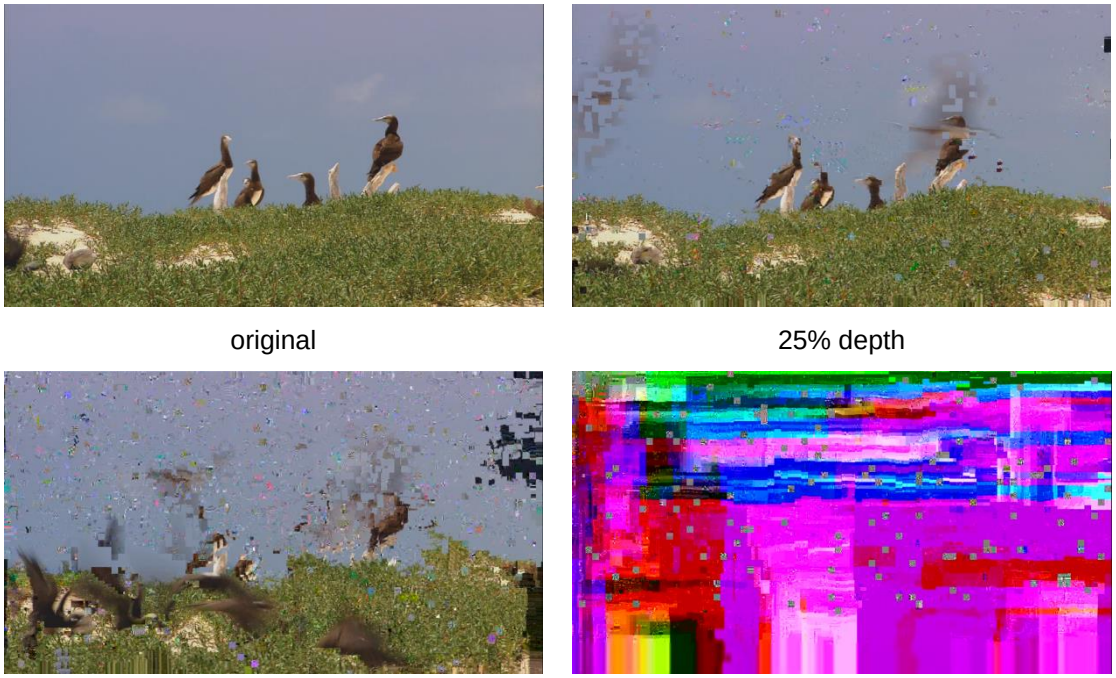


Figure 2. The Different Performance in the Different Encryption Depth

3.2. Compression Efficiency

The main processed object is the existing videos which have already been compressed and packed. So the encryption in the compressed-domain will not influence the entropy coding efficiency.

As the AES belongs to the block encryption, the bytes of output data will increase if the input bytes number is not the integer times the content key length. The Step 6 in the Part 3 will solve this problem to maintain the volume.

As we can see in the Table 1, the different videos with the different formats keep their volume before and after the encryption.

Table 1. The Size of Video before and after the Encryption

Video Source	Compression format	Size before Encryption	Size after Encryption
video3.ts	mpeg2	19.638Mb	19.638Mb
Widdlife.wmv[1]	WVC1	26.246Mb	26.570Mb
InToTree_10bit_18M.mp4	h.264	45.012Mb	45.012Mb

From the Table 1, we also can see that our algorithm can handle the different container formats, such as, Transport Stream, WMV and mp4.

3.3. Computational Complexity

The lower computational complexity means the lower energy expense or the less time consumption. The electronic equipment depending on the battery can keep working for longer time. The less time consumption makes it possible to encrypt the videos in real time.

[1] The original video does not obey the WVC1 standard strictly. The size of video will not change if the video follows the WVC1 standard fully.

There are three parts which needs to be discussed about the encryption computational complexity. The first part is the cost which is used to analyze the input video format in order to find the encryption data. The encryption consumption itself is the second part. Because the input and output are both files, the time of reading and writing file can be ignored in the real-time mode, but the time should be included in the mode of off-line. We compare the scheme of encrypting DCT coefficients [6] in the first two parts and discuss the read-write time during the encrypt process. In the third part, we will talk about the read-write consumption.

Because the scheme of encrypting DCT coefficients can only support to handle the video with the transport stream format, we choose the transport stream format videos as the input video. In order to keep the environment consistent, we move the core code of the DCT coefficients encryption algorithm to the same project with our own method.

The computer configuration used to perform the experiment is showed as follows:

Table 2. Machine Configuration

CPU	Intel(R) Core(TM) i5-4570 CPU @ 3.20GHz
RAM	4GB
OS	Windows 7 64bit
TOOL	Visual studio 2010

There are four videos with the transport stream format as the input. The detail about the four videos is displayed in Table 3.

Table 3. The Detail of the Four Videos

name	time of duration	compression format	container format	size
video1.ts	9 seconds	MPEG 2	transport stream	983Kb
video2.ts	24 seconds	MPEG 2	transport stream	2434kb
video3.ts	3 minutes 16 seconds	MPEG 2	transport stream	19222kb
video4.ts	3 minutes 46 seconds	MPEG 2	transport stream	543973kb

We encrypted each of the four videos 5 times separately, and then got the average values per encryption time. We can see the detail data in the Table 4 and the Figure 3.

Table 4. The Cost Time using Different Methods

video \ method	DCT (us)	100% depth(us)	50% depth(us)	25% depth(us)
video1.ts	47	47	30	16
video2.ts	64	112	109	105
video3.ts	297	304	260	221
video4.ts	8238	6609	4766	3526

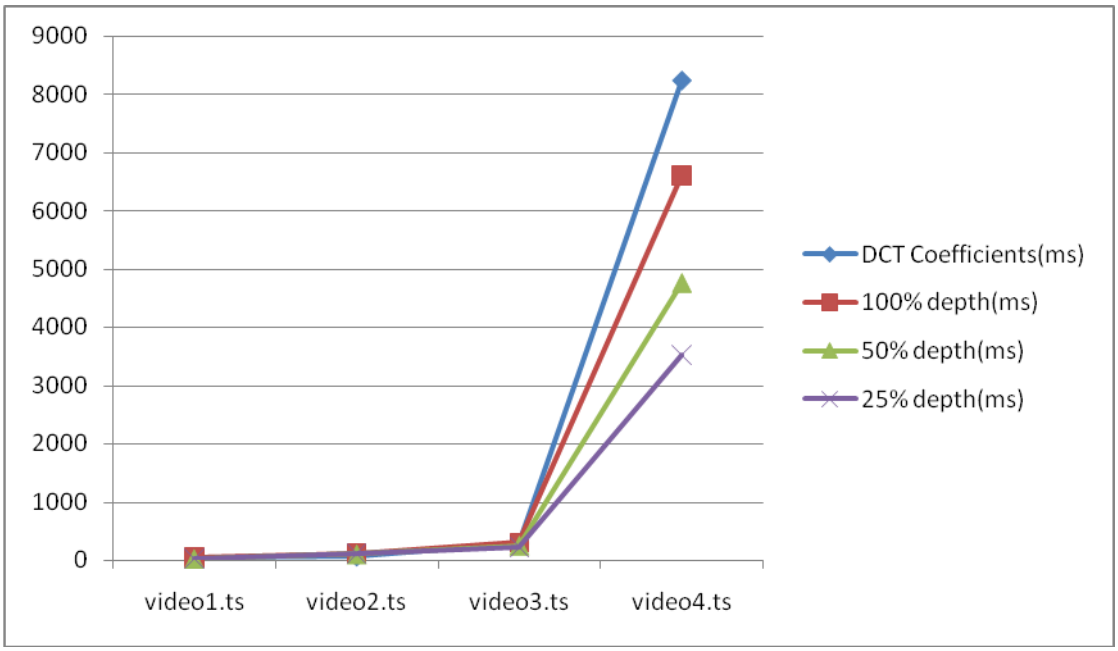


Figure 3. The Time Comparison about the Different Encryption Modes

As we can see in the Figure 3, the proposed algorithm spends a little less time than the algorithm of using DCT coefficients when we choose the 100% depth parameter. If the depth is less than 100%, our algorithm will save more time.

As is mentioned in the above context, the encryption time is composed of the time of parsing the video, encrypting the data and read-write file. Now, we will talk about the first two respectively.

We have counted the number of the encrypted data and the time of the encryption used in the both two methods. The detail data is in the following table. Our proposed method encrypts more data while using less time than the DCT coefficients encryption method. As we see from the Figure 3, the general video encryption algorithm with the 100% depth saves more time than the DCT one. So we draw a conclusion that that the DCT coefficients encryption method costs more time in the parsing process or the encryption time.

Table 5. The Encryption Data and Times

video name	encryption times		encryption data(byte)	
	DCT	100% depth	DCT	100% depth
video1.ts	9020	250	144320	124073
video2.ts	16460	1536	263360	94592
video3.ts	57130	11087	914080	2615967
video4.ts	356420	14192	5702720	503548766

In order to prove how the number of encryption times influences the whole encryption time, we carry out the following experiment. We use the same AES to encrypt the three different binary files. A whole file is divided into many blocks with the same size, so the encryption time would be different.

Table 6. The Influence of the Encryption Times

Encryption times \ File size	18.8M	226M	500M
1000	190.4	2100.2	4290
10000	227.8	2155.8	4305
30000	305.6	2249.6	4339
300000	1288.8	3351	5663

As we can see from the Table 6, the encryption times have little effect on the encryption delay. So we can see the video parsing in the encryption process may cost many time. That is the main reason between the two methods.

We will talk about the effect of the read-write time during the encrypt process in the third part of the discussion. It is very useful to discuss this content. First, the consumption of read and write should not be included into the encryption process in the real-time systems. The read and write rate should be one of system specifications. Second, choosing the proper size of the file will be encrypted according to the system read and write rate is significant in the distributed encryption structure.

The following figure shows the percentage that the read-write time consumption occupied in the whole process.

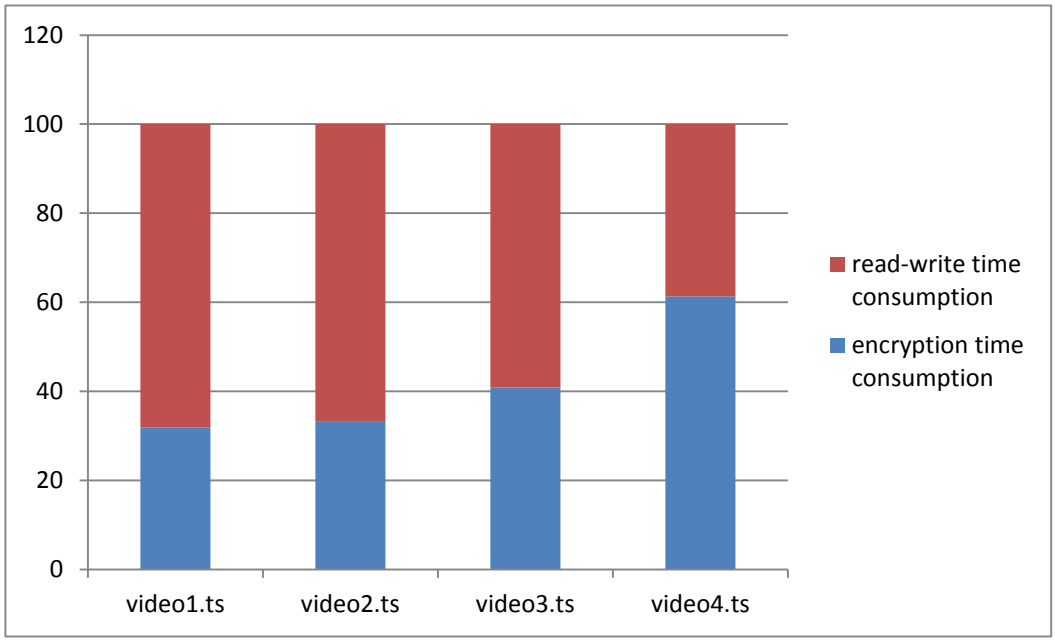


Figure 4. The Read-write Time Consumption and the Encryption Time Consumption

3.4. Operability

The video operability refers to that the encrypted videos are still able to be decoded and displayed in the player. This function is emphasized in the commercial field, because the viewer must see the part of the video contents firstly and then decide whether to buy. The consequence used our algorithm is showed in the Figure 5.

When we use the encryption depth less than 50%, the encrypted video can be displayed smoothly. When we use the 100% encryption depth, the videos data are fully disturbed. The player parse the video to get the general information about the video, but can get nothing about the detail of the video images.

From the aspect of the algorithm theory, we encrypt the video at the layer of the frame. So the information above the frame layer is maintained and the under layer such as slice, macro block and DCT coefficients, will be encrypted.

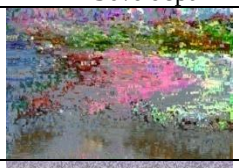
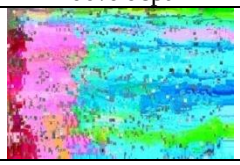
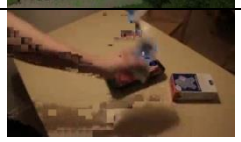
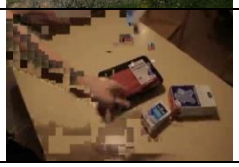
	25% depth	50% depth	100% depth
WMV			
mp4			The player can decode and display the video, but show the black screen[2].
Transport stream			The player can decode and display the video, but show the black screen.

Figure 5. The Encrypted Videos' Operability

4. Conclusions

In this paper, we propose a general video encryption algorithm in order to solve the problem that the different encryption schemes are need to encrypt the existing multiple formats videos. The experiment proves that our algorithm can handle the formats of WMV, mp4, Transport Stream and so on. From the experiments, our scheme provides several security levels and does not change the video compression ratio. It is also be proved that the proposed scheme has the same (in computational complexity) as or somewhat better than the DCT coefficients encryption scheme. The proper encryption depth used can maintain the full video operability.

Acknowledgement

The work on this paper was supported by the National Science and Technology Support Plan (2014BAH10F00) and the New Century Talent Scheme from Education Ministry of China (NCET-12-0944).

References

[1] A. Kulkarni, S. Kulkarni, K. Haridas and A. More, "Proposed Video Encryption Algorithm V/S Other Existing Algorithms: A Comparative Study", International Journal of Computer Applications, vol. 65, **(2013)**, pp. 1-5.
[2] G. A. Spanos and T. B. Maples, "Performance Study of a Selective Encryption Scheme for the Security of Networked, Real-Time Video", in Proceedings of Fourth International Conference on Computer Communications and Networks - IC3N'95 (1995), pp. 2-10.
[3] L. Qiao and K. Nahrstedt, "A New Algorithm for Mpeg Video Encryption", in Proc. of First International Conference on Imaging Science System and Technology (Citeseer, 1997), pp. 21-29.
[4] D. C. Gavankar, M. Chatterjee and S. J. Lawand, "Secure Real Time Remote Video Monitoring Using Selective Encryption", in Information and Communication Technologies (WICT), 2012 World Congress on (IEEE, 2012), pp. 453-57.
[5] L. Tang, "Methods for Encrypting and Decrypting Mpeg Video Data Efficiently", in Proceedings of the fourth ACM international conference on Multimedia (ACM, 1997), pp. 219-29.

[2] The player can parse the encrypted video and get the video information, such as the video duration, the video format and the current playing position.

[6] R. EL Metzler and S. S. Agaian, "Selective Region Encryption Using a Fast Shape Adaptive Transform", in Systems Man and Cybernetics (SMC), 2010 IEEE International Conference on (IEEE, 2010), pp. 1763-70.
[7] M. Roy and C. Pradhan, "Secured Selective Encryption Algorithm for Mpeg-2 Video", in Electronics Computer Technology (ICECT), 2011 3rd International Conference on (IEEE, 2011), pp. 420-23.
[8] S. J. Wee and J. G. Apostolopoulos, "Secure Scalable Video Streaming for Wireless Networks", in Acoustics, Speech, and Signal Processing, IEEE International Conference on **(2001)**, pp. 2049-52.
[9] Y. Zhao and L. Zhuo, "A Content-Based Encryption Scheme for Wireless H. 264 Compressed Videos", in Wireless Communications & Signal Processing (WCSP), 2012 International Conference on (IEEE, 2012), pp. 1-6.
[10] S. Singh, N. Verma and V. Kumar, "A Survey Report on Video Encryption and Decryption Techniques", **(2014)**.
[11] F. Bellard and M. Niedermayer, "Ffmpeg", Availabel from: http://ffmpeg. org **(2012)**.

Authors

Hao Li, he is a PhD student studies in the Communication University of China. His current research interests include digital right management, key management and users' privacy protection.