Private Video Foreground Extraction through Chaotic Mapping based Encryption in the Cloud

Xin Jin[1,3,*], Kui Guo[1], Chenggen Song[1], Xiaodong Li[1,*], Geng Zhao[1], Jing Luo[1,2], Yuzhen Li[1,2], Yingya Chen[1], Yan Liu[1,2], and Huaichao Wang[3]

[1]Beijing Electronic Science and Technology Institute, Beijing 100070, China,
GOCPCCC Key Laboratory of Information Security, Beijing 100070, China
[2]Xidian University, Xi'an, 710071, China
[3]Information Technology Research Base of Civil Aviation Administration of China,
Civil Aviation University of China, Tianjin 300300, China
{jinxin,lxd}@besti.edu.cn

Abstract. Recently, storage and processing large-scale visual media data are being outsourced to Cloud Data Centres (CDCs). However, the CDCs are always third party entities. Thus the privacy of the users' visual media data may be leaked to the public or unauthorized parties. In this paper we propose a method of privacy preserving foreground extraction of video surveillance through chaotic mapping based encryption in the cloud. The client captures surveillance videos, which are then encrypted by our proposed chaotic mapping based encryption method. The encrypted surveillance videos are transmitted to the cloud server, in which the foreground extraction algorithm is running on the encrypted videos. The results are transmitted back to the client, in which the extraction results are decrypted to get the extraction results in plain videos. The extraction correctness in the encryption videos is similar as that in the plain videos. The proposed method has several advantages: (1) The server only learns the obfuscated extraction results and can not recognize anything from the results. (2) Based on our encryption method, the original extraction method in the plain videos need not be changed. (3) The chaotic mapping ensure high level security and the ability to resistant several attacks.

Keywords: Privacy Preserving, Video Surveillance, Foreground Extraction, Chaotic Mapping, Cloud Computing

1 Introduction

Digital video surveillance has been equipped everywhere in our daily life and public security. Nowadays, cloud computing has changed the way of traditional video surveillance. The big data of surveillance videos are stored and automatically analysed in the cloud server, which supports large scale video surveillance applications such as face tracking, suspect searching.

* Corresponding Authors: {jinxin,lxd}@besti.edu.cn

However, tremendous surveillance cameras have distributed everywhere. The privacy of the contents in the surveillance videos from the public places is being violated. One can suppose that a unauthorized person violates the cloud server and track your trajectory, your home address and the company you work for are learned by the enemy. Such a result can bring violence or crimes for you.

In this work, we focus on the fundamental problem of video surveillance application: the foreground extraction. As shown in Fig. 1, our scenario is set as that the surveillance client captures videos, which are then encrypted and transmitted to the cloud server. The cloud server run a foreground extraction algorithm in the encrypted videos and get the obfuscated results, which are sent back to the client. The client decrypts the obfuscated results and get the final foreground extraction results in plain videos. Using traditional foreground extraction methods, the contents of client videos are completely known to the server.

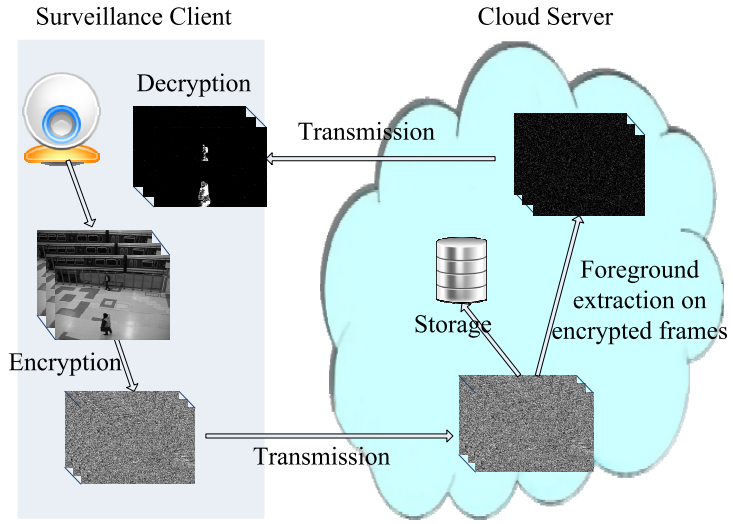

Fig. 1. The application scenario. The client captures surveillance videos, which are then encrypted by our proposed chaotic mapping based encryption method. The encrypted surveillance videos are transmitted to the cloud server, in which the foreground extraction algorithm is running on the encrypted videos. The results are transmitted back to the client, in which the extraction results are decrypted to get the extraction results in plain videos. The server learns nothing and the client knows where the foreground objects are.

To protect the privacy of the public surveillance videos, we propose a method of privacy preserving foreground extraction of video surveillance through chaotic mapping based encryption in the cloud. The server learns nothing and the client knows where the foreground objects are. The foreground extraction is a fun-

damental problem in surveillance video analysis and can be used as the inputs several high level video surveillance tasks such as object tracking, action recognition.

Related Work. Recently, various privacy preserving computer vision algorithms in the cloud have been proposed [1] [2] [3]. Upmanyu et al. [1] propose a system for privacy preserving video surveillance. They split each frame into a set of random images. Each image by itself does not convey any meaningful information about the original frame, while collectively, they retain all the information. Their solution is derived from a secret sharing scheme based on the Chinese Remainder Theorem, which has homomorphic property to some extends. Due to the limitation of the Chinese Remainder Theorem, this system can only support several related simple vision algorithms such as background subtraction for foreground extraction, which limits their application in real world problems.

Most recently, Chu et al. [2] propose a method for real-time privacy preserving moving object detection in the cloud. However, through our experiments, it is found that this method has a main drawbacks: (1) the server can clearly see the contours of the foreground objects, which can release some privacy information of the original surveillance videos, (2) the security of their encryption method is not that well because of the less secure random scheme if the randomness functions used in their work are not based on chaotic mapping.

The particular properties of chaos [4] [5], such as sensitivity to initial conditions and system parameters, pseudo-randomness, ergodicity and so on, have granted chaotic dynamics as a promising alternative for the conventional cryptographic algorithms. The inherent properties connect it directly with cryptographic characteristics of confusion and diffusion, which is presented in Shannon‘s works [6]. Chaotic system has been widely used in chaotic cryptography in recent years for its excellent chaotic dynamic properties, which could maintain longer periodicity in digitalization and gain good performance in cryptography. Moreover, it could be implemented in parallel by hardware, and has larger key space. Thus, it is also suitable for image encryption [7] [8] [9] [10] . Modern image or video encryption needs to follow a rule that cipher image should be sensitive with the changes of the secret key and plain images or videos.

The Proposed Method. Thus, we employ chaotic mapping to increase the security of the video encryption scheme and we de-shuffle the extraction results in the client to avoid the leak of the contours of the foreground objects. First, the client captures surveillance videos, which are then encrypted by our proposed chaotic mapping based encryption method. The encrypted surveillance videos are transmitted to the cloud server, in which the foreground extraction algorithm is running on the encrypted videos. The results are transmitted back to the client, in which the extraction results are decrypted to get the extraction results in plain videos. The extraction correctness in the encryption videos is similar as that in the plain videos. The proposed method has several advantages: (1) The server only learns the obfuscated extraction results and can not recognize anything from the results. (2) Based on our encryption method, the original extraction

method in the plain videos need not be changed. (3) The chaotic mapping ensure high level security and the ability to resistant several attacks.

2 Cryptography Primitive

In this section we briefly introduce the cryptography primitive we used in this paper. The simple but efficient logistic mapping is defined as follows:

$$\begin{aligned} &x_{n+1} = \mu x_n(1 - x_n) \\ &3.569945672... < \mu \leq 4, 0 \leq x_n \leq 1 \\ &n = 0, 1, 2, ... \end{aligned} \tag{1}$$

When the parameter μ and the initial value x_0 follow the Eq. 1, the outputs of this chaotic mapping $x_n \in (0, 1)$ become chaotic state and have good potential to form a random sequence.

3 Private Video Foreground Extraction

An overview of our method is shown in Fig. 2. In this section, we will describe the details of the random inverse, frame confusion, frame diffusion and the final foreground extraction.

3.1 Random Inverse

Firstly, we use the logistic chaotic mapping in Eq. 1 to generate a sequence of random number $x_n \in (0, 1)$, whose length equals to the total number of the frame pixels $L = M \times N$, where M and N are the length and the height of the video frame. The randomness evaluation of the chaotic system shows that when selecting the eighth digit from 16 digit after the decimal point of the output of the logistic chaotic mapping, the randomness of the sequence is the best. Thus, we select the eighth digit x_n^8 of the output of the logistic chaotic mapping.

For example, when $\mu^I = 3.9, x_0^I = 0.62$, one of the subsequence of the output sequence $\{x_n^I\}$ of the logistic mapping is:

$$\begin{aligned} \{x_n^I\} = \{...&,0.804374867512651, \\ &0.613688166103959, \\ &0.924592503462883, \\ &0.271912703412174, \\ &0.772107122027503, \\ &0.686235085153447, ...\}_L \end{aligned} \tag{2}$$

Then we select the $8th$ digit after the decimal point and get the sequence:

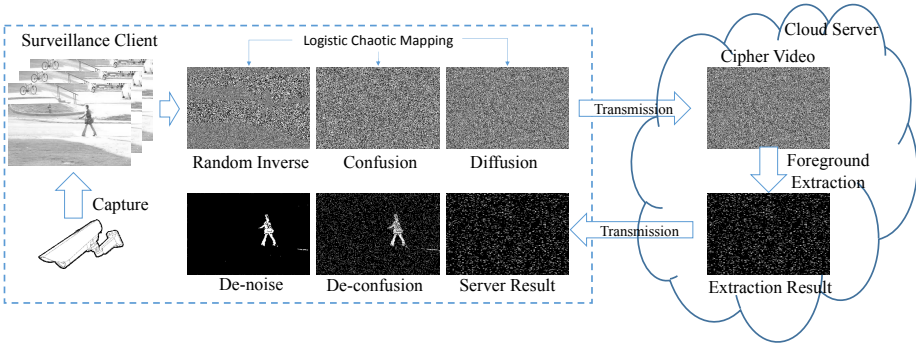

Fig. 2. The overview of our method. In the client, each frame of the captured videos is first been randomly inversed pixel by pixel using a random matrix generated by the logistic mapping. Then we use the logistic mapping to confuse the randomly inversed frame, followed by a diffuse operation to get the final cipher frame. Each are transmitted to the cloud server. The cloud server use the standard mixture Gaussian model [11] to extract foreground directly in each cipher frame. The server can only learn the cipher result and can not recognize anything in the server result. This result is transmitted back to the client. The client de-confuse the server result to get the intermediate extraction result, followed by the median filter to de-noise so as to obtain the final foreground extraction result.

$$\{L_n\} = \{..., 6, 6, 0, 0, 2, 8, ...\}_L \tag{3}$$

This sequence is converted to a $M \times N$ matrix Z. We inverse the original plain video frame F_p (8 bit gray image) as follow:

$$F_I^i = \begin{cases} F_p^i & \text{if } L_i \leq 5 \\ 255 - F_p^i & \text{if otherwise} \end{cases} \tag{4}$$

where the $M \times N$ matrix F_I is the result of the random inverse operation.

3.2 Frame Confusion

In the confusion step, we randomly change the position of each pixel of the matrix F_I. Once again, we use the logistic chaotic mapping in Eq. 1 and another pair of the parameter μ^C and the initial value x_0^C to generate a chaotic sequence $\{x_n^C\} = \{x_1^C, X_2^C, ..., X_L^C\}$ with the length of $L = M \times N$. Then we sort this sequence $\{x_n^C\}$ in ascending order:

$$\begin{aligned} \{x_n^{C'}\} &= \{x_1^{C'}, X_2^{C'}, ..., X_L^{C'}\} \\ Ind &= \{i_1, i_2, ..., i_L\}, x_{i_n}^C = x_n^{C'} \end{aligned} \tag{5}$$

According to the index sequence Ind, the confused matrix F_C of the F_I is defined as:

$$F_C(i,j) = F_I(Ind(i \times N + j)) \tag{6}$$

where N is the height of the video frame.

3.3 Frame Diffusion

In the diffusion step, the value of each pixel in the confused matrix F_C will be changed randomly. By the logistic chaotic mapping in Eq. 1, we use the parameters $\{\mu^{D_1}, \mu^{D_2}, ..., \mu^{D_Q}\}$ and the initial value $\{x_0^{D_1}, x_0^{D_2}, ..., x_0^{D_Q}\}$ to generate Q confused matrix $\{F_{D_1}, F_{D_2}, ..., F_{D_Q}\}$ using the method in Section 3.2. Thus the diffused matrix F_D of the F_C is defined as:

$$F_D = \frac{1}{P+1} \cdot (F_C + \sum_{q=1}^{Q} F_{D_q}) \tag{7}$$

where the coefficient $\frac{1}{P+1}$ ensures that each pixel of the cipher frame is in the range of $[0, 255]$.

3.4 Foreground Extraction

The foreground extraction is done by the standard mixture Gaussian model [11], which can be directly conducted in the cipher frame F_D for three reasons: (1) the inverse operation, the confusion and the diffusion only shift the original mixture Gaussian distribution of each pixel to another mixture Gaussian distribution for 3 facts: (a) the inverse of a pixel only changes the mean value of the Gaussian distribution, (b) the confusion operation do not have any impact because every pixel has its own Gaussian distributions, (c) the sum of Gaussian variables is also a Gaussian) [2], (2) we use the same inverse, confusion and diffusion matrix for each video frame, thus each pixel in the cipher frame F_D still follows a Gaussian distribution, (3) the foreground extraction of the mixture Gaussian model is conducted in each pixel independently.

4 Experimental Results

We test the proposed method in various categories of surveillance videos from a large public dataset [12], which contains nearly 16000 manually annotated surveillance video frames and several subset from other public datasets. The tested surveillance videos contains categories of baseline, shadow, night videos, and intermittent object motion. We test the correctness rate of the foreground extraction, the visual results and the security analysis. In addition we compare our method with that of Chu et al. [2]. Notice that, all the foreground extraction experiments are run in cipher video frames. The experimental results reveal that:

– The correctness rate of our method are slightly higher than that of Chu et al. [2] in cipher frames.
– Our server can learn nothing about the client video while the server in Chu et al. [2] can clearly observe the contours of the foreground objects.
– Our method is secure enough in several attacks such as the brute-force attack and the statistical attack.

4.1 The Correctness Rate

The correctness rate of the foreground extraction is defined as the number of pixels correctly labelled as *foreground* or *background* against the total pixels in the video sequences. As shown in Fig. 3, we use the same standard mixture of Gaussian model as Chu et al. [2], thus the overall correctness rate in the cipher videos of our method is only slightly higher than that of [2] to some extend.

videos/frames	Our Method	Method of [2]
backdoor/1816	0.795028	**0.807418**
busstation/1111	**0.94699**	0.939721
cubicle/2811	**0.974191**	0.973237
highway/1179	**0.936468**	0.929562
office/629	**0.901332**	0.897626
pedestrians/921	**0.988313**	0.984055
sofa/628	0.944334	**0.945863**

Fig. 3. The comparison of the correctness rate with [2] in seven video sequences. The overall correctness rate in the cipher videos of our method is only slightly higher than that of [2] to some extend.

4.2 The Extraction Results

The foreground extraction algorithm of [11] is tested in 7 video sequences from the public change detection benchmark dataset [12]. In the work of Chu et al. [2], the server can completely observes the contours of the extracted foreground objects. While in our work, the server can only observes the shuffled extracted results as random white points, and can not recognize anything of the client videos. We show the foreground extraction results of our method and Chu et al. [2] in Fig. 4. **The video results of two of the test scenes are shown in the supplemental material**.

4.3 Security Analysis

A well designed image/video encryption scheme should be robust against different kinds of attacks, such as brute-force attack and statistical attack [7]. In

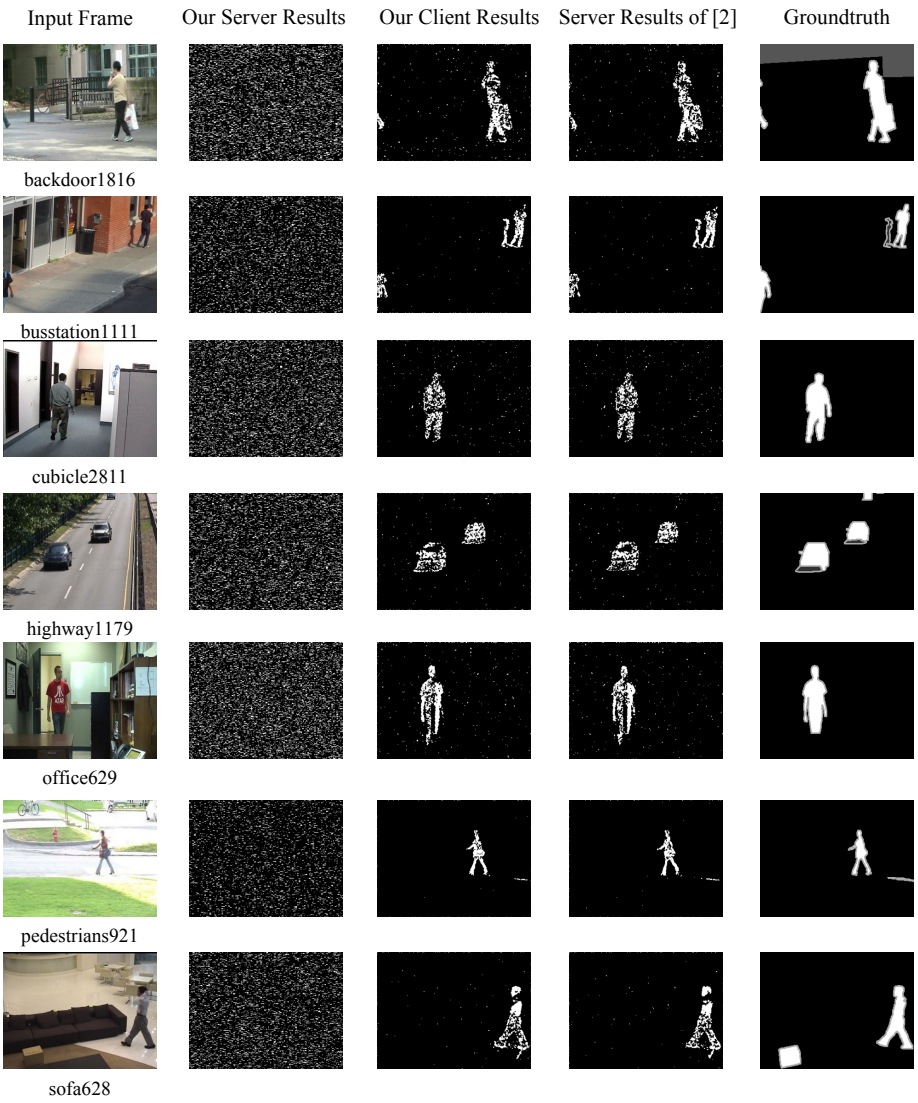

Fig. 4. Parts of the foreground extraction results. The input plain video frames are shown in the first column. The extraction results in the server using our method are shown in the second column. The extraction results after the encryption and median filter in the client are shown in the third column. The extraction results in the server using the method in [2] are shown in the fourth column. The serve can clearly observe the contours of the foreground objects and the privacy of the client video is leaked. The ground truth manually segmented and annotated in [12] are shown in the last column.

this section, we analyse the security of the proposed encryption method in an example frame from the pedestrian video sequence of the [12].

Resistance to the brute-force Attack. The key space of the encryption scheme should be large enough to resist the brute-force attack, otherwise it will be broken by exhaustive search to get the secret key in a limited amount of time. In our encryption method, we have the key space as follow:

$$\begin{cases} \text{Random Inverse} & : \{\mu^I, x_0^I\} \\ \text{Frame Confusion} & : \{\mu^C, x_0^C\} \\ \text{Frame Diffusion} & : \{\{\mu^{D_1}, \mu^{D_2}, ..., \mu^{D_Q}\}, \{x_0^{D_1}, x_0^{D_2}, ..., x_0^{D_Q}\}\} \end{cases} \tag{8}$$

The precision of 64-bit double data is 10^{-15}, thus the key space is about $(10^{15})^{4+2Q}$, in our experiments $Q = 3$, and the key space is $10^{150} \approx 2^{499}$, which is much lager than the max key space (2^{256}) of practical symmetric encryption of the AES. Our key sapce is large enough to resist brute-force attack.

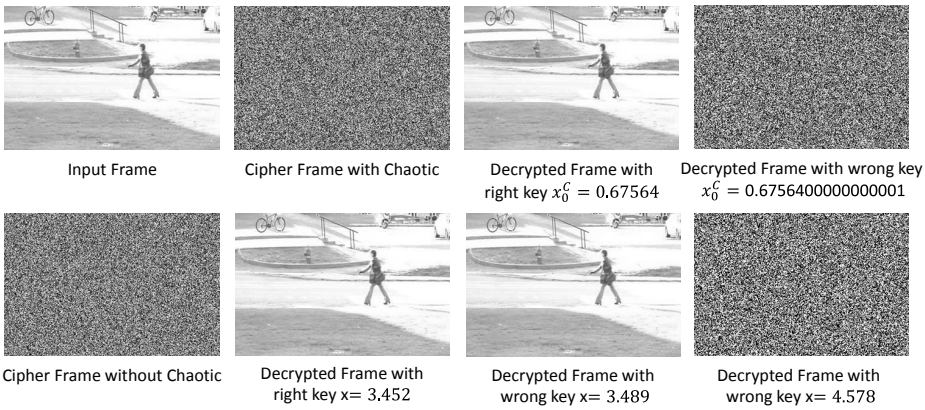

Fig. 5. Decrypted with wrong key. First line: using our method we slightly change the key and get the wrong decrypted result. Second line: using a standard random function, the sensitivity of the secret key x in the confusion step as a example is not well, and is much easier be attacked by the brute-force attack

Sensitivity of Secret Key. The chaotic systems are extremely sensitive to the system parameter and initial value. A light difference can lead to the decryption failure. To test the secret key sensitivity of the image encryption scheme, we change the secret key of the confusion step x_0^C as shown in Fig. 8. We use the changed key to decrypt the input frame, while the other secret keys remain the

same. The decryption result and the comparison with the randomness function without chaotic mapping are shown in Fig. 5. We can see that the decrypted frame is completely different from the input frame. The test results of the other secret key are similar. The experiments show that the encryption scheme is quite sensitive to the secret key, which also indicates the strong ability to resist exhaustive attack compared to whose randomness functions are not based on chaotic mapping.

The Histogram Analysis. The histogram is used to show the distribution of pixel values of a frame. The histogram of cipher frame should be flat enough, otherwise some information can be leaked to cause the statistical attack. This makes cipher-only attack possible through analysing the statistic property of the cipher image. Figure 6. shows the histograms of the input frame and its corresponding cipher image, respectively. Comparing the two histograms we can see that the pixel values of the original frame are concentrated on some values, but the histograms of its cipher image are very uniform, which makes statistical attacks impossible.

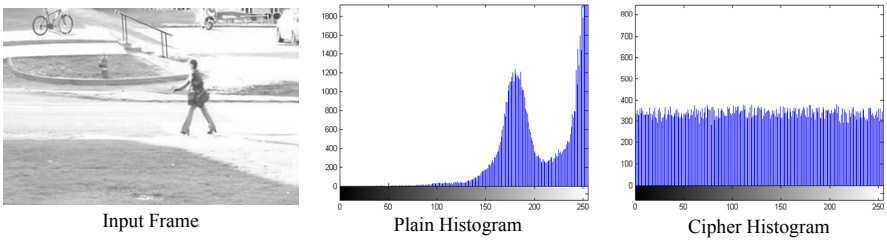

Fig. 6. The histogram of the input frame before and after encryption.

The Information Entropy. The information entropy [7] is used to express randomness and can measure the distribution of gray values in the image. The more uniform the distribution of pixel gray values, the greater the information entropy is. It is defined as follows:

$$\mathrm{H}(m) = -\sum_{l=0}^{L} \mathrm{P}(m_i) \log_2(m_i) \tag{9}$$

where m_i is the i-th gray value for an L level gray image, $L = 255$. $P(m_i)$ is the probability of m_i in the image and $\sum_{i=0}^{L} P(m_i) = 1$. The information entropy of an ideal random image is 8, which shows that the information is completely random. The information entropy of the cipher image should be close to 8 after

encryption. The closer it is to 8, the smaller possibility for the scheme leaks information. The information entropy of the cipher frame using our method in Fig. 5 in our experiments is 7.9978, which is very close to 8.

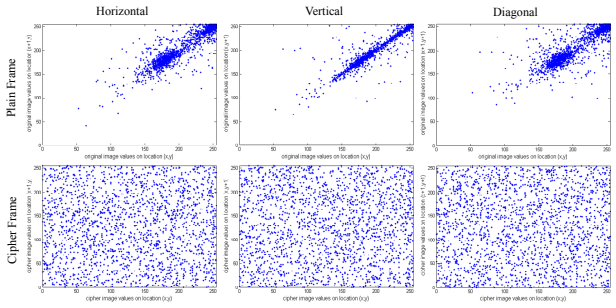

Fig. 7. The correlation of the input frame before and after encryption in 3 directions (horizontal, vertical and diagonal).

The Correlation Analysis. Correlation indicates the linear relationship between two random variables. In image/video processing, it is usually employed to investigate the relationship between two adjacent pixels. Usually, the correlation of between adjacent pixels in the plain image is very high. A good encryption scheme should reduce the correlation between adjacent pixels, i.e., the less correlation of two adjacent pixels have, the safer the cipher image is. In order to test the correlation of two adjacent pixels, we test 3 directions (horizontal, vertical and diagonal) of adjacent pixels from the input frame using our method in Fig. 5 and its corresponding cipher frame.

5 Conclusion and Discussion

In this paper we propose a method of privacy preserving foreground extraction of video surveillance through chaotic mapping based encryption in the cloud. We employ chaotic mapping to increase the security of the video encryption scheme and we de-shuffle the extraction results in the client to avoid the leak of the contours of the foreground objects. The experimental results show that The correctness rate of our method are slightly higher than that of Chu et al. [2] in cipher frames. Our server can learn nothing about the client video while the server in Chu et al. [2] can clearly observe the contours of the foreground objects. Our method is secure enough in several attacks such as the brute-force attack and the statistical attack.

In the future work, we will use high dimensional chaotic mapping to increase the secure level. We plan to embed most recent foreground extraction algorithms into the privacy preserving framework.

6 Acknowledgements

This work is partially supported by the National Natural Science Foundation of China (No.61402021, No.61402023, No.61170037), the Fundamental Research Funds for the Central Universities (No.2014XSYJ01, No.2015XSYJ25), the Science and Technology Project of the State Archives Administrator. (No.2015-B-10), and the Open Project Foundation of Information Technology Research Base of Civil Aviation Administration of China (No.CAAC-ITRB-201403)

References

1. Upmanyu, M., Namboodiri, A.M., Srinathan, K., Jawahar.C.V. Efficient Privacy Preserving Video Surveillance. IEEE 12th International Conference on Computer Vision (ICCV), 1639-1646 (2009)
2. Chu, K.Y., Kuo Y.H., Hsu W.H. Real-Time Privacy-Preserving Moving Object Detection in the Cloud. ACM Multimedia, 597-600 (2013)
3. Jin, X., Liu, Y., Li, X.D., Zhao, G., Chen, Y.Y., Guo, K. Privacy Preserving Face Identification through Sparse Representation. To Appear in the 10th Chinese Conference on Biometric Recognition (CCBR), Tianjin, China, 13-15, November (2015)
4. Huang, C., Nien, H. Multi chaotic systems based pixel shuffle for image encryption. Optical Communication. 282:21232127 (2009)
5. Lian, S., Sun, J., Wang, Z. A block cipher based on a suitable use of the chaotic standard map. Chaos Soliton Fract 26(1):117129 (2005)
6. Claude, S. Communication theory of secrecy systems. Bell System Technical Journal 28(4):656715 (1949)
7. Zhen, P., Zhao, G., Min, LQ., Jin, X. Chaos-Based Image Encryption Scheme Combining DNA Coding and Entropy. Multimedia Tools and Applications (MTA), Published Online: 10 April (2015)
8. Jin, X., Chen, Y.Y., Ge, S.M., Zhang, K.J., Li, X.D., et al. Color Image Encryption in CIE L*a*b* Space. To Appear in the 6th International Conference on Applications and Techniques for Information Security (ATIS), Beijing, China, 4-6 November, (2015)
9. Jin, X., Tian, Y.L., Song C.G., Wei, G.Z., Li, X.D., et al. An Invertible and Anti-Chosen Plaintext Attack Image Encryption Method based on DNA Encoding and Chaotic Mapping. To Appear in the Chinese Automation Congress, 27-29, November, Wuhan, (2015)
10. Li, Y.Z., Li, X.D., Jin, X., Zhao, G., Ge S.M., et al. An Image Encryption Algorithm based on Zigzag Transformation and 3-Dimension Chaotic Logistic Map. To Appear in the 6th International Conference on Applications and Techniques for Information Security (ATIS), Beijing, China, 4-6 November, (2015)
11. Stauffer, C., Grimson, W.E.L. Adaptive Background Mixture Models for Real-Time Tracking[J]. International Conference on Computer Vision and Pattern Recognition,2:2246 (1999)
12. Wang, Y., Jodoin, P.M., Porikli, F., Konrad, J., Benezeth, Y., Ishwar, P. CDnet 2014: An Expanded Change Detection Benchmark Dataset, in Proc. IEEE Workshop on Change Detection (CDW-2014) at CVPR-2014, pp. 387-394. (2014)