Design and Realization of Video Signals Encryption and Decryption via Chaotic Sequence

Chaowen Shen

Security and Protection department, Zhejiang Police Vocational Academy, 310018 Hangzhou, China

Abstract. As is known to all, data transmission in wireless networks is easy to intercept because of the openness of wireless signals. In order to deal with this security problem, this paper presents a secure communication method based on driving-response synchronization and TCP protocol. The video signal is encrypted and decrypted by a chaotic sequence and transmitted through WIFI network in real time. Furthermore, the system configuration and design procedures are described in detail. Experimental results show that our approach is feasible for video secure communication.

1 Introduction

With the rapid development of mobile Internet, cloud computing, social networking, big data and other information technologies, the security of multimedia data has attracted more and more attention in recently years. However, the large amount of multimedia data and the strong correlation between adjacent elements make the traditional text encryption algorithm encounter severe challenges. To solve these problems, scientists have introduced chaotic secure communication technology into the transmission process of multimedia signals [1-4].

Chaotic secure communication is mainly generated by chaotic sequences with good random performance. It can be applied in multimedia encryption and decryption. The complex chaotic structure and behaviour can greatly improve the security of the system. Therefore, chaotic cryptography has a good application prospect in secure communication [5-6].

The rest of this paper is organized as follows. In Sect. 2, the structure of the secure communication system is introduced. In Sect. 3, the principle of chaotic encryption and decryption of signals is investigated. In Sect. 4, the system design and experiment results are given. Finally, Sect. 5 concludes the paper.

2 Video secure communication system

2.1 System structure

The system structure for realizing video signals encryption and decryption is shown in Figure 1, which can be divided into 3 parts. The first part is the server, including video acquisition module, signal coding module and chaotic encryption module. The server is mainly responsible for video capture, coding, encryption, and pre-processing for the wireless transmission. The second part is wireless network transmission, including WiFi module and wireless router, which function is for data transmission and reception through TCP transmission protocol. The last part is the client, including the chaotic decryption module, the signal decoding module and LED module, which is mainly in charge of the decryption, decoding and restored signals displaying.

Figure 1. The main structure of the system.

2.2 Hardware platform selection

This system is developed under ARM embedded platform and Linux environment. In the case of cost-effective, ARM11 chip S3C6410 is chosen as the hardware platform. The development board integrates the video stream coding and decoding functions, which can also run Android systems and support NAND-flash. Moreover, the development board is also equipped with a number of expansion modules, such as CMOS camera module, WiFi module, LED screen.

The hardware consists of two parts, the server and the client. The server captures video data and performs chaotic encryption. The client is responsible for signals
decryption and displaying. The corresponding hardware platforms are shown in figures 2 and 3 respectively.

2.3 The features of video signal

The video signal has the characteristics of large data volume, fast transmission rate, and high network bandwidth requirements. In this system, the CMOS camera can collect video signals with pixels of 340×240×2, which can make the video signals clarity.

Figure 2. Hardware platform of the server.

Figure 3. Hardware platform of the client.

3 The principle of chaotic encryption and decryption

3.1 Driving-Response Synchronization

The theory of driving-response synchronization has been widely applied to the field of chaotic secure communication. The main block diagram is shown in Figure 4 [7].

Consider n-D discrete time chaotic system of the form

$$x(k+1) = Ax(k) + u(k)$$ (1)

where $x(k+1) = [x_1(k+1), x_2(k+1), \ldots, x_n(k+1)]^T$, $x(k) = [x_1(k), x_2(k), \ldots, x_n(k)]^T$, $A = (a_{ij})_{n \times n}$, $u(k)$ is the state variable controller, presented by

$$u(k) = u_m = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$ (2)

where $m, j = 1, 2, \ldots, n; m \neq j$, $\sigma_j$ is the gain coefficient, and $e_j$ is the finite region. The matrix $A$ can be described as follows:

$$A = \begin{bmatrix} a_{11} & \cdots & a_{1,j-1} & a_{1,j} & a_{1,j+1} & \cdots & a_{1n} \\ \vdots & & \vdots & & \vdots & & \vdots \\ a_{m,1} & \cdots & a_{m,j-1} & a_{m,j} & a_{m,j+1} & \cdots & a_{mn} \\ \vdots & & \vdots & & \vdots & & \vdots \\ a_{j,1} & \cdots & a_{j,j-1} & a_{j,j} & a_{j,j+1} & \cdots & a_{jn} \\ \vdots & & \vdots & & \vdots & & \vdots \\ a_{n,1} & \cdots & a_{n,j-1} & a_{n,j} & a_{n,j+1} & \cdots & a_{nn} \end{bmatrix}$$ (3)

In order to make the nominal system asymptotically stable, it must be ensured that the $n$ eigenvalues of the matrix $A$ are located in the unit circle [8].

Theorem 1 Considering the controlled system (1), the drive-response system can be synchronized if the following conditions are satisfied.

Let $|\sigma_j| < l (j = 1, 2, \ldots, n)$ in (3). The corresponding cofactor $M_j$ of $A$ satisfies that $n - 1$ eigenvalues are located in the unit circle, which is presented by

$$M_j = \begin{bmatrix} a_{11} & \cdots & a_{1,j-1} & a_{1,j} & a_{1,j+1} & \cdots & a_{1n} \\ \vdots & & \vdots & & \vdots & & \vdots \\ a_{m,1} & \cdots & a_{m,j-1} & a_{m,j} & a_{m,j+1} & \cdots & a_{mn} \\ \vdots & & \vdots & & \vdots & & \vdots \\ a_{j,1} & \cdots & a_{j,j-1} & a_{j,j} & a_{j,j+1} & \cdots & a_{jn} \\ \vdots & & \vdots & & \vdots & & \vdots \\ a_{n,1} & \cdots & a_{n,j-1} & a_{n,j} & a_{n,j+1} & \cdots & a_{nn} \end{bmatrix}$$ (4)

3.2 Typical example

Considering the memory capacity of the ARM and the real-time requirements of the system, a three-dimensional chaotic system is selected, as shown in (5).
\[
\begin{pmatrix}
    x_1(k+1) \\
    x_2(k+1) \\
    x_3(k+1)
\end{pmatrix} =
\begin{pmatrix}
    x_1(k) & & \\
    x_2(k) & & \\
    x_3(k) & & \\
\end{pmatrix}
+ \begin{pmatrix}
    u_1 \\
    u_2 \\
    0
\end{pmatrix} \quad (5)
\]

Let \( A = \begin{pmatrix} 0.1 & -0.1 & 0.1 \\ -0.1 & 0.15 & 0.1 \\ 0.13 & -0.12 & 0.1 \end{pmatrix} \), \( u(k) = [u_1, u_2, 0]^T \).

\( u_i = e_2 \sin(\sigma_2 X_i(k)) \), \( \sigma_2 = 15 \), \( e_2 = 1.8 \times 10^3 \).

According to Theorem 1, the system (5) can synchronize.

Therefore, the driver system can be described as
\[
\begin{align*}
    x_1^{(d)}(k+1) &= a_1 x_1^{(d)}(k) + a_{12} p(k) + a_{13} x_3^{(d)}(k) + u_1^{(d)} \\
    x_2^{(d)}(k+1) &= a_{21} x_1^{(d)}(k) + a_{22} x_2^{(d)}(k) + a_{23} x_3^{(d)}(k) \\
    x_3^{(d)}(k+1) &= a_{31} x_1^{(d)}(k) + a_{32} p(k) + a_{33} x_3^{(d)}(k)
\end{align*}
\]

where \( u_1^{(d)} = e_2 \sin(\sigma_2 p(k)) \), \( p(k) = s(k) + x_2^{(d)}(k) \) is the encrypted signal, \( s(k) \) is a byte pixel video signal, and \( x_2^{(d)}(k) \) is a 4-byte chaotic sequence.

Similarly, the response system can be expressed as
\[
\begin{align*}
    x_1^{(r)}(k+1) &= a_{11} x_1^{(r)}(k) + a_{12} p(k) + a_{13} x_3^{(r)}(k) + u_1^{(r)} \\
    x_2^{(r)}(k+1) &= a_{21} x_1^{(r)}(k) + a_{22} x_2^{(r)}(k) + a_{23} x_3^{(r)}(k) \\
    x_3^{(r)}(k+1) &= a_{31} x_1^{(r)}(k) + a_{32} p(k) + a_{33} x_3^{(r)}(k)
\end{align*}
\]

where \( u_1^{(r)} = e_2 \sin(\sigma_2 p(k)) \).

Due to (6)-(7), one can get the video encryption and decryption, whose block diagram is shown in Figure 5.

The decrypted signal is \( s(k) = x_2^{(r)}(k) \oplus p(k) \).

When the drive side and the response side are precisely synchronized, one can obtain
\[
\begin{align*}
    x_2^{(r)}(k) &= x_2^{(d)}(k) \\
    s(k) &= x_2^{(r)}(k) \oplus x_2^{(d)}(k) \oplus s(n) = s(n)
\end{align*}
\]

Therefore, the initial video signal can be restored.

System design includes the design of underlying file, calling video capture function, display function, encoding and decoding function, etc. The flow chart of the soft design is shown in Figure 6.

In our system, the video signal is transmitted by WIFI network. In order to ensure the high-quality of the network, the TCP protocol is used for transmission. TCP programming includes two aspects: server programming and client programming. The general steps on the server side are as follows: (1) Create a socket with the function socket(); (2) Set the socket property by using the function setsockopt(); (3) Bind the IP address, port, and other information to the socket with the function bind(); (4) Turn on listening with the function listen(); (5) Accept connections from clients with function accept(); (6) Send and receive signal by using functions send() and recv(); (7) Close network connection with closesocket(); (8) Close listening. The client side takes the same steps as the server side, which requires three handshakes operation to establish a reliable connection. The flow chart of the TCP protocol is shown in Figure 7.

4 The realization of video encryption and decryption

4.1 Software design

The software design mainly consists of three parts, namely Linux embedded system design, TCP network programming and algorithm design. The embedded

Figure 5. The principle diagram for signals encryption and decryption.

Figure 6. The flow chart of the software design.

Figure 7. The flow chart of the TCP protocol.
4.2 Hardware design

The hardware system consists of two ARM development boards, which are server and client respectively, as shown in Figures 8 and 9. The server side connects the CMOS camera and wireless transceiver, which is responsible for collecting and sending video signals. The client connects to the wireless transceiver and the LED, which is in charge of the receiving and displaying video signals in real-time.

4.3 Experimental results

After a series of processes such as theoretical analysis and parameter adjustment, the experimental results are shown in Figure 10-12, which are the original signals, the encrypted signals, and the recovered signals.

It can be seen that the encryption effect is satisfactory. The video signals after decryption and restoration are also highly consistent with the original signals when all parameters are exactly match in the client side. It also shows that the use of chaotic sequences to encrypt video signals can be successfully implemented.

5 Conclusion

In this paper, according to a 3-D discrete-time chaotic system, the video signal is encrypted and decrypted in real time by using a driving-response synchronization method. In the experiment, two development boards are used as server and client respectively. Based on the ARM embedded platform and TCP transmission protocol, wireless chaotic video security communication is realized via Wi-Fi transmission. Furthermore, the hardware implementation results are given, which proves the effectiveness of this approach. In the subsequent experimental research, the H.264 encoding and decoding of video will be considered, and the problem of transmission rate will be further solved.

Acknowledgement

This research was financially supported by the Scientific Research Fund of Zhejiang Provincial Education Department under Grant Y201533283 and the Fundamental Research Funds for Classroom teaching reform project of Zhejiang Provincial Education Department under Grant kg20160874.

References

1. Q. Wang, S. Yu, C. Li, J. Lu. IEEE Trans. Circuits Syst. I, 63, 3 (2016).
2. Z. Lin, S. Yu, C. Li, J. Lu. Int. J. Bifurcation Chaos Appl, 2016, 26, 9 (2016).
3. K. Li, M. Zhao, X. Fu, IEEE Trans. Circuits Syst. I 56, 10 (2009).
4. C. Luo, X. Wang, Int. J. Mod. Phys. C, 24, 4 (2013).
5. H. Khanzadi, M. Eshghi, S. Borujeni, IETE J. Res. 59, 1 (2013).
6. K. Rao, C. Gangadhar, IETE J. Res. 58, 2 (2012).
7. L. Pecora, T. Carroll, Phys. Rev. Lett. 6, 8 (1996).
8. H. Zheng, S. Yu, Int. J. Bifurcation Chaos Appl. 24, 4 (2014).