A resource-efficient multimedia encryption scheme for embedded video sensing system based on unmanned aircraft

Chen Xiao \textsuperscript{a,}\textsuperscript{*}, Lifeng Wang \textsuperscript{b}, Mengjiao Zhu \textsuperscript{a}, Wendong Wang \textsuperscript{c}

\textsuperscript{a} School of Computer Science and Technology, Beijing University of Posts and Telecommunications, 100876, China
\textsuperscript{b} Department of Electronic and Information Engineering, Beijing Electronic Science and Technology Institute, 100070, China
\textsuperscript{c} School of Software Engineering, Beijing University of Posts and Telecommunications, 100876, China

A R T I C L E   I N F O

Article info

Available online 20 July 2015

Keywords:

Video sensing
Multimedia big data
Video encryption
Generalized information security
Embedded system

A B S T R A C T

Battery-powered unmanned aerial vehicle based video sensing system is more cost-saving and energy-saving than traditional aircraft based systems. However, high-volume real-time sensing data is more vulnerable in unmanned system than that in manned system. Meanwhile, the computation and energy resources in such system are very limited, which restricts the use of complex encryption process on video data. Therefore, how to achieve confidentiality of video data under limited resources efficiently needs to be addressed. Firstly, resources constraints with their development trends in video sensing system are studied. Secondly, an information-utility-value-oriented resource-efficient encryption optimization model under resources constraints is given. Thirdly, based on this model, a video-compression-independent speed-adjustable lightweight encryption scheme with its improved version is proposed. Fourthly, a DSP and ARM based embedded secure video sensing system is designed, and the proposed encryption scheme has been implemented in it. In addition, theoretical analyses based on information theory and experimental analyses on throughputs show that the proposed encryption schemes can meet the real-time requirements of system under the tight resources constraints.

© 2015 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

1. Introduction

1.1. Background

In recent years, applications of Internet of Things (IoT) have drawn wide attention from both academia and industry. Multimedia sensing is one of the important IoT applications (Atzori et al., 2010). Unmanned Aerial Vehicle (UAV) based video sensing system is a new class of multimedia sensing. The battery-powered unmanned system can largely reduce the operational risks and be more energy-saving than video sensing system based on traditional aircraft.

Nevertheless, the information technology has always been haunted by information security problem. And this problem is more serious in network times. Consequently, the IoT applications, which are based on network communication technology, inherit the security problem naturally. So the security issue becomes a central concern which may hamper the development of IOT technology and has attracted widespread attentions (Granjal et al., 2015; Almalkawi et al., 2010). As for UAV-based video sensing, since there is no pilot in the aerial vehicle, the aircraft is easily captured by attackers, which makes the airborne data more vulnerable. However, protecting real-time big multimedia sensing data needs complex encryption operation. For battery-powered sensor devices, when compared with general non-battery-powered IoT equipments, the limited computation and energy resources further restrict the protection of data security. Then these limitation make the security issue of multimedia data with resources constraints more troublesome (Ning and Liu, 2012; Chen et al., 2009; Min, 2014).

1.2. Related works on video encryption

To solve the problem of high-volume multimedia data protection, optimized encryption schemes for audio/video data have been researched (Lian et al., 2009; Singh and Manimegalai, 2012). As for our application scenario, that is considering different steps of video compression where encryption process is introduced, those schemes could be generally divided into the following two classes. The first class can be called partial encryption or selective encryption in general. The main idea of this class is taking advantage of the characters of compressed video, and using the...
traditional encryption algorithm to encrypt a selected crucial subset of the video data after or in a certain step of the video compression. Although the left data, except the selected crucial subset, is transmitted in plaintext and could be wiretapped by attackers, the missing selected subset makes attackers hard to rebuild a discernible video.

The first class can then be divided into two subclasses. In the first subclass, some typical algorithms only encrypt the I-frames of video (Maples and Spanos, 1995). In these algorithms about one third of the total data needs to be encrypted (Agi and Gong, 1996). Despite the P-frames and B-frames are transmitted in clear, and could be eavesdropped by attackers, it is very hard to rebuild a clear A/V without the I-frames.

Another subclass is designed based on the first subclass. This subclass is generally called joint compression encryption scheme (Singh and Manimegalai, 2012; Mallika and Sivakumar, 2014; Nidhi et al., 2014). Compared with the first one, the main idea of this subclass is applying the encryption in a certain processing step of the video compression algorithms, while dividing the encrypted subsets into smaller segments. For example, those schemes always encrypt DCT (Discrete Cosine Transform) coefficients, MV (Motion Vector) and other parameters in video encoding.

As to the first subclass, the experimental analysis given in (Agi and Gong, 1996) shows that without the crucial subset, attackers may also get a video with a decreased quality. That is because the unencrypted I-frames in P-frames and B-frames can be decoded fully. Albeit these data lose commercial values after encrypted by such schemes, part of the video can be rebuilt and understood. So those schemes are not secure enough for some confidential applications. An improved scheme is proposed by Meyer and Gadegast (1995), where for levels of security are implemented. One important level is to encrypt all the I-frames and I-frames in both P and B frames. Such scheme increases the security of the encrypted data, but lowers the encryption speed and information leakage still cannot be totally avoid (Agi and Gong, 1996).

Compared to the first subclass of schemes, the second subclass schemes are faster. However, since the encryption rate is relatively small, their security is comparatively low (Lian et al., 2009; L. Qiao and Nahrstedt, 1998, 1997). To be specific, as to the security requirement of application discussed in this paper, just by lowering the qualities of the videos to protect the commercial value of data is not secure enough. Because the information such as contours and movements of the main objects of the videos are all needed to be protected. Under this circumstance, the first class is hard to be qualified. In addition, since those schemes need to combine with video compression processes, as to modularized video capture devices whose video compressions have already been implemented in hardwares, joint compression schemes are hard to integrate with such devices.

Although the joint compression encryption schemes have some security problems, as they have a high encryption speed, they are convenient to be used in commercial or recreational applications to protect the data by reducing the qualities of videos.

The second class is developed from traditional full encryptions, which is commonly called improved full encryption. The scheme presented by Qiao and Nahrstedt (1997) is a representative one. It is a simplified full encryption which is similar to each iterative operation of DES encryption algorithm. In one iterative step of DES algorithm, 64 bit input plaintext is divided into two parts: 32 bit \(L_i\) and 32 bit \(R_{i-1}\), and let output \(L_i = R_{i-1} \oplus F(R_{i-1}, K_i)\), where \(F\) is a permutation function, and \(K_i\) is a key.

Qiao and Nahrstedt (1997) firstly analyzed the statistical character of compressed video data. The analysis results show that the data have high randomness at the byte level because of the use of variable-length Huffman codes and other processes in the video compression. Based on this statistical character of compressed video, Video Encryption Algorithm (VEA) is proposed. The main idea of VEA scheme is as follows: (1) divides the plaintext block into 2 subblocks: \(EVEN\) and \(ODD\) parts, (2) encrypts the \(ODD\) part with a standard encryption algorithm \(E\) to get a half of the ciphertext \(E(ODD)\), (3) exclusive-OR-ed s (\(\oplus\) or \(\lor\)) \(EVEN\) and \(ODD\) parts to get another half \(C\) where \(C = ODD \oplus EVEN\). Therefore, the final ciphertext is the integration of \(E\) and \(ODD\). This mechanism can reduce the encryption complexity to 53% of that of the used standard full encryption algorithm \(E\), and achieve a sufficient security level. Tosun and Feng (2000) further extended this scheme. They bisected the odd parts, and only encrypt one fourth of the plaintext. This scheme reduced the encryption complexity to about one fourth of standard full encryption algorithm \(E\).

Compared with the first class, such schemes can encrypt all kinds of compressed data. However, since its encryption rate is fixed, the improvements of encryption speeds of these schemes are limited.

The two previous classes of optimized encryption schemes always focus on the real-time character rather than the optimization of the use of limited resources (Lian et al., 2009; Liu and Koenig, 2014; L. Qiao and Nahrstedt, 2011; Heer et al., 2011; Hs and Chen, 2012). A resource-efficient, video-compression-independent, throughput-adjustable encryption scheme which can achieve an acceptable video security under tight resources constraints is sorely expected.

1.3. Security analyses of video encryption

Then we analyze the security of multimedia encryption schemes based on Shannon information theory (Shannon and Weaver, 1949; Weaver, 1949; Guiasu, 1977). The variables we used are shown as following: (1) a random multimedia object is denoted as \(M\); (2) traditional Shannon entropy is denoted as \(H(M)\); (3) conditional entropy, which is the uncertainty of \(M\) under the condition of known \(Z\), is denoted as \(H(M|Z)\); (4) mutual information, which is the amount of information about information source \(X\) after \(Y\) is known, is denoted as \(I(X;Y)\). As to generalized information, (5) generalized information entropy of multimedia event \(M\) is denoted as \(H_r^g(M)\); (6) generalized information conditional entropy is denoted as \(H_r^g(M|Z)\); (7) generalized mutual information is denoted as \(I_r^g(X;Y)\). Let \(E_r(M)\) be the encryption output after \(M\) is encrypted by full encryption algorithm \(E_r\). According to classical information theory, we can get \(H_r^g(M|E_r(M))=H_r^g(M)\), and then \(I_r^g(X;Y)=H_r^g(M)−H_r^g(M|E_r(M))=0\).

Very few researches discuss the theoretical security of multimedia encryptions. As far as we know only (Gibson et al., 2004) analyzes the security of its selective encryption of scalable speech coding based on Shannon theory. They use \(X_i\) to indicate core layer information, and use \(X_2^n\) to indicate enhancement layer information. Then the uncertainty of \(X_i\) given \(X_2^n\) can be denoted as \(H(X_i|X_2^n)\). They draw the conclusion that \(H(X_i|X_2^n)\) is equal to \(H(X_2^n)\), or equivalently, mutual information \(I(X_i,X_2^n)=H(X_i|X_2^n)=H(X_2^n)\).
There is no information about $X_{\widehat{0}}^{\text{plaintext}}$ that would be encrypted, namely whether $I(M; X_{\widehat{0}}^{\text{plaintext}}) = 0$, is not important. Nonetheless, the most important issue is whether unencrypted $X_{\widehat{0}}^{\text{plaintext}}$ contains information of the video which needs protection, i.e., whether $I^e(M; X_{\widehat{0}}^{\text{plaintext}}) = H(M) - H(MX_{\widehat{0}}^{\text{plaintext}})$ equals to 0 is very important. For example, as to the typical selective encryption scheme, I-frames in the video are encrypted, but P-frames and B-frames are not encrypted. Even P and B-frames (considered as $X_{\widehat{0}}^{\text{plaintext}}$) do not contain the information of I-frames (considered as $X_{\widehat{0}}^{\text{plaintext}}$), i.e., $I(M; X_{\widehat{0}}^{\text{plaintext}}, X_{\widehat{0}}^{\text{plaintext}}) = 0$, P and B-frames do have a large amount of information of video. (Agi and Gong, 1996) analyzed and attacked the videos encrypted by scheme (Maples and Spanos, 1995), and gave images of some decompressed frames. As shown in the images of their paper, contours of the main objects of videos could be rebuilt and be understood by attackers. As to commercial applications rather than academic ones, contours and movements of the main objects actually have a higher value to attackers.

In selective or partial encryption, the most ideal solution is to find an independent data subset $X_{\widehat{0}}^{\text{plaintext}}$ that contains few information of the video (i.e., $I^e(M; X_{\widehat{0}}^{\text{plaintext}})$ would be close to 0) and can be transmitted in plaintext. But it is almost impossible for most multimedia compression standards to get the ideal solution. However, even such subset $X_{\widehat{0}}^{\text{plaintext}}$ can be found, if the data volume is not big enough, i.e., the encrypted part $M-X_{\widehat{0}}^{\text{plaintext}}$ is not small enough, the improvement of the encryption speed is limited. On the other side, if we wish that $M-X_{\widehat{0}}^{\text{plaintext}}$ is small enough, encryption must be engaged during compression process to encrypt some subsets of data, which will lead to non-universality of encryption schemes. So there will be a tradeoff between high speed and video-compression-independent.

Compared with the first class of schemes, the improved full encryption does not have data which is sent in clear. Our proposed scheme also belongs to this class. Its security will be discussed in Section 4.4.

In this paper, we firstly analyze the application scenario and the variation trends of encryption resources constraints, then put forward a encryption optimization model, based on which we also proposed an encryption scheme aiming at the application mentioned before. Moreover, the efficiency of the proposed scheme is analyzed in the embedded prototype system we constructed.

2. Resources constraints with their variation trends in video sensing system

2.1. Application scenario and its resources constraints

Application scenario considered in this paper is a video sensing system based on a battery-powered multi-rotor Unmanned Aerial Vehicle (UAV) we designed, which is shown in Fig. 1. Since there

Fig. 1. UAV based video sensing system.
constructed; secondly, the impacts of encryption optimization on attacker’s behavior are discussed; thirdly, an information-utility-value-oriented encryption optimization model for multimedia data is proposed.

### 3.2. Security of generalized information for multimedia data

We firstly analyze the security of multimedia data based on generalized information. Due to the characteristic of multimedia compression coding, the length of compressed data is positively correlated to the complexity of original multimedia information. There could be some valuable information beyond all the bits of the original information. For example, in some sensitive applications, the complexity of original multimedia data reflected by the length of compressed data can also expose some multimedia pragmatic information to attackers. However, there is no such problem for original uncompressed video. That is because the original video data with same time lengths have same volumes.

As to generalized information of video mentioned above, their security sometimes could not be protected by traditional encryption schemes. According to the concept of generalized information given in Section 1, there could be generalized mutual information beyond all the bits of the original information, the generalized mutual information of $FE(M)$ and $M$, denoted as $P(M; FE(M))$ is greater than 0. Therefore, using traditional encryption schemes is not a sufficient condition for generalized security of multimedia data. As to the example given above, some tail filling schemes could be used to protect the information of the complexity of the videos, which can be gotten from the length of the compressed video.

Meanwhile, previous studies on multimedia encryption (Liu and Koenig, 2014; Lian et al., 2009) point out that some encryption schemes could achieve an acceptable security level by encrypting some selected crucial subsets of the video data.

Let $MES_i$ denote a selected multimedia encryption scheme, then $MES_i(M)$ is the ciphertext of $M$ encrypted by $MES_i$.

Let us assume that $MES_i$ has a constant encryption ratio $Rate(MES_i)$, where function $Rate(M)$ represents the ratio between the encryption resources cost of $M$, and full encryption. Thus, we can get the following expression:

$$\exists MES_i(T^*(M; MES_i(M)))/H^*(M) < 1 − Rate(MES_i).$$

(1)

Therefore, full encryption is neither a necessary nor a sufficient condition for generalized information security of multimedia data. It is necessary to build a formalized model to discuss the costs and benefits of multimedia encryption processes.

Guiasu (1977) is an earlier literature putting forward the concept of utility value of information. Zhong (1998) suggests that utility value and pragmatic of information have a similar implication. Furthermore, we believe the utility value of data is related with application scenarios. We use $v(M; Scenario)$ to show the utility value of multimedia data $M$ in a specific application scenario which is denoted as $Scenario$.

### 3.3. Information-utility-value-oriented security optimization based on attacker behavior analysis

We use $valueLeak(M, MES_s, Scenario, Cost)$ to denote the maximum utility value of information that could be stolen from $MES_s(M)$ by an attacker with total resources cost $Cost$. Here $MES_s(M)$ is the ciphertext of $M$ encrypted by $MES_s$. The costs spent by attacker include the time, money, computational resources, electric energy of unplugged equipments and other limited resources. According to the definition of generalized mutual information, we have $valueLeak(M, MES_s, Scenario, Cost) < = P^*(M; MES_s(M))$.

As for those legitimate users who have the key of $MES_s$, they need just the normal decryption cost $DecryptionCost$ to get the original multimedia data $M$. At this time, $valueLeak(M, MES_s, "UserHaveKey", DecryptionCost) = v(M, Scenario)$. As for those attackers who have unlimited resources, $valueLeak(M, MES_s, Attacker, \infty) = v(P^*(M; MES_s(M), Scenario))$.

If we focus on the relationship of function value $valueLeak()$ and costs of attackers $Cost$ in a given condition where the type of multimedia data $M$, multimedia encryption scheme $MES_s$, and $Scenario$ are fixed, we can regard the leaked value as a unary function of $Cost$: $valueLeak^*_{MES_s}(Cost)$, where $valueLeak^*_{MES_s}(Cost)$ = $valueLeak(M, MES_s, Scenarios, Cost)$ and the function satisfies following properties:

$$0 \leq valueLeak^*_{MES_s}(Cost) \leq v(M, Scenarios).$$

(2)

And first derivative of function should satisfy the following expression:

$$valueLeak^*_{MES_s}(Cost) \geq 0.$$}

(3)

Since the costs that attackers are willing to spend in different application scenarios are dramatically different, in the following analysis, behaviors of attackers are discussed under two application scenarios: (1) commercial or entertainment applications and (2) confidential applications.

#### 3.3.1. Impacts of encryption scheme optimization on attacker’s utility value benefit under CCS

In commercial application, requirement for data security is always Conditional Computational Security (CCS) (Norman and Landgrave, 1997). CCS requires that the costs of attackers should be greater than the value of information they can steal.

On the contrary, as for attackers, they wish the value of information they stole to be larger than their costs. In general, if a commercial system satisfies CCS, the higher costs may prevent attackers’ attacks. So the constrain expression of the attackers could be expressed as follows:

$$\begin{align*}
V &= valueLeak^*_{MES_s}(Cost) \\
V &\geq Cost.
\end{align*}$$

As is shown in Fig. 2(a), the vertical axis represents the utility value of multimedia information, the horizontal axis represents the resources cost of attackers. The red straight line represents the $V = Cost$, and the area of region shaded with red vertical lines on the upper left indicates the feasible region where the constraint of $V \geq Cost$ is satisfied.

Let us assume that the blue curve represents the function of $valueLeak^*_{MES_s}(Cost)$, and the region shaded with horizontal lines on the lower right part denotes the probable profits could be obtained by attackers. Hence, in Fig. 2(a), only the overlap region colored in some certain applica.

Ordinate of $e$ is the maximum utility value could be obtained by attackers.

According to the application scenarios and resources constraints, multimedia system designers should improve the encryption scheme to minimize the utility value of information gained by attackers.

As is shown in Fig. 2(b), with the improvement of encryption scheme, the blue curve will move closer to horizontal axis. Then the intersection point of the blue curve and the red straight line will move from $e_1$ to $e_2$, $e_3$, and attackers will get less information, which may make the attackers give up attacking. For example,
when intersection point $e_3$ overlaps with the origin point, attackers will not get utility value of multimedia data larger than their costs.

### 3.3.2. Impacts of encryption scheme optimization on attacker's utility value benefit under CS

Compared with it of commercial applications, value of multimedia information in confidential applications might be higher. Consequently, attackers may even put forth their best effort to attack without considering resource costs. In this occasion, we wish a higher security level Computational Security (CS) (Norman and Landgrave, 1997) could be satisfied. Compared with CCS, CS requires that the system could not be broken by attackers in an acceptable time with all best resources available. However, since resources in battery-powered embedded systems are limited, CS is too expensive to be achieved. A compromised scheme is to minimize the utility value of information which could be stolen by attackers.

We can treat CS as CCS with unlimited value of data. In this case, attackers will cost maximum probable resources, denoted as $\text{Cost}^*$, to attack the system. And the new constrain expression of the attackers could be expressed as follows:

\[
\begin{align*}
V &= \text{valueLeak}(M, \text{MES}, \text{Scenarios}, \text{Cost}) \\
\text{Cost} &\leq \text{Cost}^*
\end{align*}
\]  

As is shown in Fig. 2(c), the region shaded with red vertical lines on the left indicates the feasible region where the constraint that $\text{Cost} \leq \text{Cost}^*$ is satisfied. And the region shaded with blue horizontal lines on the lower right part denotes the feasible profits which could be obtained by attackers. The overlap region colored with green satisfies the constraint in expression Eq. (5). Ordinate of intersection point $e$ is the maximum utility value could be obtained by attackers.

Like that in Fig. 2(b), with the improvement of encryption scheme, the blue curve in Fig. 2(c) will also move closer to horizontal axis. Then the intersection point will move from $e_1$ to $e_2$, then $e_3$, and attackers will get less information. In the most optimistic case, the blue curve could overlap with the horizontal axis, and attackers will not get any utility value.

It can be seen that improvements of the encryption schemes can prevent attackers’ attacks. But improvements are always achieved at the cost of computation and energy resources. So a resources-effective encryption optimization model is very necessary.

### 3.4. Information-utility-value-oriented encryption optimization model for multimedia data under resources constraints

In this subsection, an Information-utility-value-oriented Encryption Optimization (IEO) model is proposed. The optimization
principle of this model is to select appropriate subset of data and encryption schemes to minimize the utility value which could be stolen by attackers, under both the real-time constraint and limited resources constraints of application. We use the following optimization expression to construct the optimization model:

\[
\begin{align*}
\text{min } & \text{valueLeak}(M, MES, \text{Scenario, Cast}^*) \\
\text{s.t.} & \text{real-time constraint}, \\
& \text{(2)limited resources constraints.}
\end{align*}
\]

The decision variable of the above optimization model is MES. In this model, M could be a set of target multimedia data, and MES could be a set of corresponding multimedia encryption schemes. When M is a single stream of data, MES means a specific multimedia encryption scheme. Comparatively, when M includes several streams of data, MES means a whole optimization plan including the selection of encryption algorithms and selection of streams to be encrypted.

It should determine the optimum encryption algorithm for each stream in the system, and let the used encryption resources be concentrated on the streams with higher security weightiness. In addition, application scenario Scenario also needs to be considered in the optimization.

4. Resource-efficient speed-adjustable fast multimedia encryption scheme

4.1. Application-oriented encryption optimizing strategy

According to the IEO model given in Section 3, optimization of encryption scheme should firstly take application scenario into consideration. In addition, information utility value got by attackers should be minimized under the resources constraints.

Therefore, under the application environment of our research, the following requirements should be taken into full account in the encryption scheme we study.

1) Although the camera used in the prototype system only has a resolution of 1280×720. With the increasingly extensive usage of 4K (4096×2160) pixels camera and multi-camera, the volume of big video sensing data grows very fast. At the meantime, because the embedded video system and the aerial vehicle share the limited battery, the discrepancy between the massive video data and the limited resources is very serious, and the encryption scheme should be designed as light as possible.

2) When sensing system is applied in some sensitive scenario, there might be some puissant attackers who will make analytical attacks on the data. Simple partial encryption algorithms would probably make part of the video information leaked (Agi and Gong, 1996). Then attackers might get important information from the unencrypted part and rebuild a discernible video with decreased quality. In this time, valueLeak (M, MES, Scene, Cast*) is significant. Then the security requirements of the sensitive system could not be met. Therefore, the encryption scheme should have the ability to defense some analytical attacks.

3) The throughput of existing encryption schemes cannot adjust dynamically (Massoudi et al., 2008). As to video sensing system, the bandwidth fluctuation of real-time video is significant. Hence, the throughputs of the encryption scheme need to be adjustable.

4) The existing encryption schemes are always connected with specific video encoding algorithm, and are lack of universality. As to our embedded system, video encoding module and video encryption module work independently. Thus, universal video encryption scheme will enhance the scalability of system.

4.2. Continuous data stream oriented encryption scheme

Since that in multimedia compression and coding, the variable-length Huffman coding and other coding processes remove the redundant information from the original datagram, statistical character of compressed multimedia data is dramatically different from that of text data. Statistical analysis shows that coded data has high randomness at the byte level (Qiao and Nahrstedt, 1997). Based on this statistical character of media data, we use traditional block cipher to encrypt a block of data, and use its plaintext as stream cipher key to encrypt the following I blocks. By changing the parameter l, the speed of the encryption scheme can be adjusted. The statistical property of encoded multimedia data makes analytical attacks difficult to be effective. Therefore, if the attackers cannot break the full encryption algorithm FE that used in this scheme, they are unable to get any information from the video. This method utilizes small resources and acquires a higher security assurance.

The encryption scheme is shown in Procedure 1. In the scheme, firstly, plaintext is divided into blocks with a same length BlcLength. Secondly, we use a traditional block cipher, full encryption FE, to encrypt one block. Thirdly, as to the next I blocks, we use the plaintext of the previous block as its stream cipher key.

Procedure 2 Package-oriented SAFE scheme

Procedure SAFE

1. Divide plaintext into blocks with a length of BlcLength

Repeat

2. Use FE to encrypt the first block in the buffer. 

3. For \(i = 1\) to \(l\) do

   - Let next \(l\) block’s ciphertext \(CipherBlc_j = Blc_{j-1} \oplus Blc_j\).

   - Until get the last block.

4. For the last block, encrypt it using \(FE\).

This algorithm is designed for file data. To avoid the attackers from guessing the file header, the first \(n\) blocks could be full encrypted, where \(n\) could be calculated from the session key. In addition, although the probability of a block guessed by attackers is very low, permutation could be used before the division in step 1. Parameter \(l\) in step 3 is the speed control parameter. By controlling the parameter \(l\), the encryption throughput could be adjusted. And the parameter \(l\) can be determined dynamically according to the current data volume in the buffer.

For file encryption, parameter \(l\) should be properly saved either by saving in a separate file or saving as an encryption flag \(FE\), which records whether it have been full encrypted, in the header of each block. The decryption process can determine the decryption way according to this flag. If \(FE\) is true, the plaintext equals to \(FE^{-1}(CipherBlc)\), else equals to \(PlainText_{j-1} \oplus Cipher\).

In the prototype system, we can estimate data volume based on the video resolution and the target bit rate, and then choose parameter \(l\). Meanwhile, when the fullness degree of the buffer is higher than a certain threshold value, encrypted parameter \(l\) could be doubled transiently.

4.3. Packet oriented encryption scheme

The former encryption scheme is designed for encrypting continuous data and is suitable for multimedia file encryption. However, as for wireless transmission module, this scheme cannot solve the problems of bit error and packet loss. Moreover, data is packed according to the RTP protocol. Therefore, a packet-based algorithm can achieve higher efficiency. A package-oriented scheme (POSAFE) is shown in Procedure 2.
Procedure 1 Basic SAFE scheme

Procedure SAFE
1. Divide plaintext into blocks with a length of BlcLength
2. Repeat
   3. For \( i = 1 \) to \( l \) do
      a. Let next \( i \) block's ciphertext \( \text{CipherBlc}_i = \text{Blc}_{i-1} \oplus \text{Blc}_i \)
      b. Until getting the last block.
4. For the last block, encrypt it using \( FE \).

Because adjacent packets could have different lengths, the XOR operator in Step 2 is implemented as expression Eq. (7), where \( \text{PlainByte}_i^k \) is byte \( k \) of packet \( i \), \( \text{CipherByte}_i^k \) is its ciphertext, and \( PL_i \) is the length of packet \( i \). In other word, the former packet will be duplicated at the rear of itself if it is shorter than the current packet. In addition, the encryption flag of each RTP packet could be saved in the RTP header.

\[
\text{CipherByte}_i^k = \begin{cases} 
\text{PlainByte}_i^k \oplus \text{PlainByte}_{l-1}^{k \mod PL_{l-1}} & PL_i > PL_{l-1} \\
\text{PlainByte}_i^k \oplus \text{PlainByte}_{l-1}^{k} & PL_i \leq PL_{l-1}
\end{cases}
\]

(7)

4.4. Performance analyses on SAFE

In order to analyze the throughputs of SAFE with different values of parameter \( l \), simulation program is designed in C++. The test platform is Intel Core i7 2.1 GHz quad-cores. The full encryption algorithm \( FE \) in the test is DES algorithm with EBC mode.

Experimental results shown in Table 1, are the throughputs of SAFE when the CPU has difficulty loads. The throughput is measured in MBps. Row 1 (load 1) gives the result of SAFE without any other heavy load competitor processes. Row 2 (load 2) gives the results, where a process with 75 MBps on average disk transmission is added based on load 1. In row 3 (load 3), another process which has heavy load is added based on load 1. Besides, \( l=0 \) SAFE algorithm is equal to full encryption, and \( l=1 \) SAFE is equal to VEA (Qiao and Nahrstedt, 1997).

Figure 3 shows the increasing trends of the throughput of encryption scheme SAFE with different loads. When parameter \( l \) is small, the speed of SAFE almost grows linearly. But when \( l \) is very large, limited by the speed of XOR operation, the throughputs of SAFE converge to 11 Gbps and 5.9 Gbps respectively.

By comparing the three different load series, we have known that other computation intensive process in system will compete computation resources with SAFE and impact the throughput of SAFE a lot. Contrarily, the I/O intensive process would not impact the SAFE a lot. Because the main tasks of control module of our embedded system are data encryption and storing, and the load factor may not be essential.

5. An embedded secure video sensing system

5.1. Structure of embedded system

An embedded video processing system based on DSP and ARM9 is designed, and the encryption scheme proposed in Section 4 is implemented. The hardware platform can be divided into three modules, and the system structure is shown in Fig. 4. It includes video sensing module, data processing module and control module.

5.2. Video sensing module

The video sensing module consists of capture module, A/D, FPGA and FIFO. When control commands are given by CPLD at
runtime, FPGA will parse the command words and process the captured data according to the user request patterns. Then FPGA can capture video signals and store them into FIFO in a given format, and inform data processing module by an interrupt signal. Then the DSP with EDMA transmit the data periodically into the buffer in DRAM.

5.3. Sensing data processing module

The data processing module consists of DSP, DRAM and FLASH memory. TMS320C6455 DSP is used to perform data processing. It has a frequency of 1 GHz, 2 MB RAM and 8 parallel processing units. All the intermediate data of reference frames and reconstructed frames are put into the outer temporary buffer in a 64 MByte DRAM. An 8 MB flash memory is used to save the bootloader program. In this module, H.264/AVC compression standard is adopted. At runtime, data processing module can read video data from the capture module, process them with specified H.264/AVC encoder and then transmit them to the control module. Moreover, the implementation of the encoder is optimized to achieve real-time processing.

5.4. Control module

The control module is designed based on S3C2410 processor with the kernel of ARM920T and embedded Linux. In this module, DRAM and Flash memory are used as the memory, while the CPLD is used to provide external chip select, and the dual RAM is used to engage the data transmission between the DSP and ARM, and JTAG port is used for debugging and testing.

5.5. Implementation and analysis of the encryption schemes

In the control module, H.264 encoded stream is encrypted by the encryption scheme proposed in Section 4 while the parameter \( l \) used in the prototype system is 64. Then the encrypted video stream is stored in flash disk. In consideration of the low resolution of sensing video in prototype system, a fixed parameter \( l \) can meet the requirements of real-time application easily. If multicamera with 4 K (4096×2160) high resolution is adopted in the practical system, a large parameter \( l \) could be selected and be adjusted by the method mentioned before. In the functions of wireless transmission, the control module sends the snapshot video produced by the data processing module to remote control unit through wireless channel. Snapshot data is packaged using RTP protocol. And the RTP payload is encrypted using POSAFE proposed in Section 4.3.

The typical resolution ratio we implemented in the prototype system is 1280×720@25fps while the target bit rate is 5 Mbps. According to the results in Section 4.4, if the parameter \( l \) in SAFE is set to 64, the throughput of the scheme would be greater than 2 Mbps. And the computation resources occupation rate of the encryption process is about 2%, which would not be a significant overhead for the control module.

Thereby, even several 4 K high-definition sensing equipments are introduced, the problem of real-time encryption also can be solved by adjusting the parameter \( l \) of SAFE.

6. Conclusions

Real-time video data in embedded video sensing system has a high bandwidth. However, as to battery- powered system, limited computation and energy resources restrict the usage of full encryption process on video data. Firstly, this research analyses the discrepancy between encryption resources and volume of real-time data, and then draws a conclusion that the discrepancy would not converge because the increase of encryption ability is difficult to catch up with the increase of data volume. Secondly, to get a better optimization for encryption scheme, impacts of encryption optimization on attacker’s behavior are discussed. And then, information-utility-value-oriented encryption optimization model for multimedia data under resources constraints is proposed. Such model can optimize the distribution of limited resources, then minimize the probable information utility value which could be stolen by attacker. Thirdly, based on aforementioned model a video-compression-independent speed-adjustable fast encryption scheme and a packet-oriented improved version are proposed. Fourthly, an embedded video sensing system has been designed, and the proposed encryption schemes have been implemented in data storing and transmission module respectively. Theoretical analyses based on information theory and experimental analyses indicate that the performance of the proposed scheme is effective enough to support the encryption of real-time data in embedded video sensing systems.

Acknowledgments

This work was partially supported by the National Natural Science Foundation of China (Grant no. 61300182 and 61370197), and the Fundamental Research Funds for the Central Universities (Grant no. BUPT2012RC0209).

References


