Low Complexity Secured Image Compression for Wireless Image Sensor Network with Chaotic Map

1G. Suseela and 2Y. Asanth Victhy Phamila

1School of Computing Science and Engineering,
VIT University,
Chennai, India.

2School of Computing Science and Engineering,
VIT University,
Chennai, India.

Abstract

The problem of secure image transmission over Wireless Image Sensor Network is addressed. The challenge originates from the nature of image data and finite energy budget imposed on wireless sensor network. Visual data in the form of images are self-descriptive than scalar data like pressure, temperature etc., Hence WISN has attained wide application in various surveillance and monitoring systems. Images are captured by camera attached sensor nodes and transmitted over unreliable bandwidth limited wireless link. If tampered the images will reveal more information as they are self-descriptive. Typically sensor nodes are embedded systems built up with microcontrollers and small RAM. The AES, DES algorithms are complex and consume more energy. To deal with the problem, a low complex energy efficient secured image coding using chaotic map and bit shuffling is proposed. The efficiency of the proposed system is analyzed with NPCR (Number of Pixel Changing Rate), UACI (Unified Average Changing Intensity), PSNR and energy consumption.

Key Words: Low complex image encryption, Fibonacci-Lucas transform, chaotic system, secured transmission.
1. Introduction

Wireless Sensor Networks are infrastructure less, self-organized adhoc networks. Almost all the applications require secure transmission without jeopardizing resource scarcity. As the transmitter is the power hunger component of the wireless system, the amount of data to be transmitted is reduced by an energy efficient compression technique. In this method the data are which are to be discarded by the compression system is unnecessarily processed. The proposed method does the encryption along with the compression. There is little work available in secured image transmission over WSN. No work is in the literature with energy consumption analysis of their encryption algorithm. The entire system architecture is as depicted in Fig. 1 and same is simulated for target platform ATmega 128 microcontroller over Atmel’s AVR Studio. The energy consumed by the entire system for processing an 8x8 block found to be as low as 5.63µJ (micro Joules).

2. Proposed System

Transformation and Quantization

The proposed system uses an integer DCT transformation which uses only few additions and bit shift operations based on Chen’s factorization [1]. The work proposed in [2] is adapted for low complex image compression. For an 8 x 8 input block only the most significant components around DC coefficient are computed. This low complexity integer DCT uses 120 additions and 10 binary shift operations for a two dimensional transform of 8 x 8 input block and computes only the top four components (DC component and three coefficients around DC) of the transform matrix. The transformed coefficients are normalized using scalar quantization. The quantization is done with binary shift operation instead of complex floating point division operation.

Fig. 1: Proposed System Design
Encryption

The image encryption algorithm follows the traditional two step method. The first step is pixel confusion – obfuscating the image by changing the pixel position and second step is pixel diffusion modifying the pixel value. In this work pixel confusion is accomplished by Lucas and Fibonacci (LF) Transform whereas pixel diffusion is carried out by bit shuffling of the adjacent pixels in each block.

Confusion is done using chaotic image scrambler. Chaotic systems are dynamic systems which is widely used for image encryption. Arnold’s 2D cat map was the first chaotic system employed for image cryptography and many researchers followed Arnold and developed many encryption algorithms. In this paper a 2D chaotic map for pixel block shuffling is employed by combining Lucas series and Fibonacci series. Lucas series is special type of Fibonacci series found by mathematician François Édouard Anatole Lucas [3].

Fibonacci series is

\[ f_1 = 1, \]
\[ f_2 = 1, \]
\[ f_n = f_{n-1} + f_{n-2}, \quad n > 2 \]

Any 2x2 matrix formed by four successive terms is unimodular. A unimodular matrix is periodic. Hence they can be used as image scrambler.

The Lucas series is

\[ l_1 = 2, \]
\[ l_2 = 1, \]
\[ l_n = l_{n-1} + l_{n-2}, \quad n > 2 \]

As like Fibonacci series the matrix formed by Lucas series are not unimodular. By coalescing the Lucas series with Fibonacci series the transform matrix becomes unimodular. For example by combining the first two terms of the both series, the Lucas and Fibonacci (LF) transform matrix is formed as
The LF transform for block scrambling the pixel block \((a, b)\) is given as

\[
\begin{bmatrix}
    l_1 \\
    f_1 \\
    f_2
\end{bmatrix}
= \begin{bmatrix}
    2 & 1 \\
    1 & 1
\end{bmatrix}
\begin{bmatrix}
    a \\
    b
\end{bmatrix}
\mod M
\]

In (1), \((a', b')\) is the new position of the pixel block,

\[
M = \frac{\text{width of the image}}{\text{size of the pixel block}}
\]

and ‘n’ is the number of rotations or in other words scrambling key and typical value of n for which the image is well scrambled is 8.

### 3. Challenges of Chaotic Systems in Resource Constrained WISN

In general chaotic systems entail more memory, equivalent to the size of plain image. Each block should be projected on its new position without overwriting any block. To avoid this limitation the position \((a', b')\) is considered as the plain image pixel block position. The pixel block \((a', b')\) is projected on the position \((a, b)\) as in equation (2). Also each pixel block is individually processed and coded independent of other block.

\[
\begin{bmatrix}
    a \\
    b
\end{bmatrix}
= \begin{bmatrix}
    f_1 \\
    f_2
\end{bmatrix}
\begin{bmatrix}
    a \\
    b
\end{bmatrix}
\mod M
\]

In encryption system diffusion is an important procedure as it modifies the statistical redundancy exists in the plain image. At diffusion stage, the value of the pixel is modified by bit shuffling of the adjacent pixels of each block without any mathematical operation as shown in Fig. 3 which also reduces the energy consumption.

![Fig. 3: Bit Shuffling](image)

**Entropy Coding**

The Golomb–Rice coding is much suitable for bandwidth limited networks like WSN as it is a dictionary less coder which leads to less data transmission by the transmitter. The coder is designed in a way such that the symbols in each block is coded independent of other blocks. Also the coder uses complementary...
approach for coding the negative values instead of extra bits for representing the sign as implemented in [2]. The bitrate offered by the entire system varies from 0.27bpp to 0.49 bpp and PSNR of 27.6dB to 28 dB with various quantization value Q.

Table 1: Encryption Results of Various Test Images of Size 512X512

<table>
<thead>
<tr>
<th>Image</th>
<th>NPCR (%)</th>
<th>UACI (%)</th>
<th>PSNR(dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lena</td>
<td>99.96</td>
<td>33.38</td>
<td>∞</td>
</tr>
<tr>
<td>Tank</td>
<td>99.98</td>
<td>34.33</td>
<td>∞</td>
</tr>
<tr>
<td>Barbara</td>
<td>99.92</td>
<td>33.53</td>
<td>∞</td>
</tr>
<tr>
<td>Peppers</td>
<td>99.94</td>
<td>33.23</td>
<td>∞</td>
</tr>
<tr>
<td>Lab1</td>
<td>99.95</td>
<td>34.44</td>
<td>∞</td>
</tr>
</tbody>
</table>

4. Experimental Results and Discussion of the Encryption System

Encryption results of the standard test images and randomly taken test images are presented in Table 1. Fig.4 shows the encryption and decryption results of standard Lena image and random test image Lab1 for visual analysis. The encrypted image should not reveal any kind of information to the attacker. In Table 2 the performance of the proposed encryption algorithm is compared with AES image encryption algorithm by the encryption parameters such as Mean Absolute Error (MAE), Entropy, NPCR, UACI, PSNR and the speed of execution [4].

NPCR and UACI

The number of pixels changing rate (NPCR) and unified average changing intensity (UACI) are the parameters used for analyzing the strength of the encryption algorithm against differential attack. NPCR and UACI are computed between two cipher images C1 and C2. The C1 differs from C2 in the corresponding original image by at least one pixel value or change in initial conditions of the encryption system. When one pixel is modified or some change in initial conditions of the system the encryption system should offer a completely different cipher image. The ideal values of NPCR should be higher i.e. ≈ 100 % and the values of UACI should be between 33%-36%. From Table 1 the proposed encryption algorithm is strong against differential attacks. A binary matrix D(x, y) is assigned as 1, If E1(x, y) =E2(x, y) otherwise D(x, y) is assigned as 0. The NPCR and UACI are defined as

\[
NPCR = \frac{\sum_{x=1}^{M} \sum_{y=1}^{N} [D(x, y)]}{MN} \times 100\%
\]

\[
UACI = \left[ \frac{\sum_{x=1}^{M} \sum_{y=1}^{N} [|E1(x, y) - E2(x, y)|]}{255} \right] \times 100\% / MN
\]

Mean Absolute Error

It is the mean absolute difference between the encrypted and original image. Higher the MAE means more distance between the original image and
encrypted image. The proposed encryption offered higher MAE than standard benchmark AES image encryption.

\[ \text{MAE} = \frac{1}{MN} \sum_{x=1}^{M} \sum_{y=1}^{N} [IP(x,y) - IE(x,y)]^2 \]

Where IP and IE are plain image and encrypted image respectively.

Fig. 4: Encryption results of Lena and Random Greyscale Image of Size 512 x 512

<table>
<thead>
<tr>
<th>Method</th>
<th>MAE</th>
<th>Entropy</th>
<th>NPCR (%)</th>
<th>UACI (%)</th>
<th>PSNR (dB)</th>
<th>Time (Sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>AES</td>
<td>34.64</td>
<td>7.99</td>
<td>99.61</td>
<td>33.17</td>
<td>∞</td>
<td>242.4</td>
</tr>
<tr>
<td>Proposed</td>
<td>57.95</td>
<td>7.71</td>
<td>99.97</td>
<td>33.38</td>
<td>∞</td>
<td>21.964</td>
</tr>
</tbody>
</table>

**Entropy**

As per Shannon’s information entropy theory, if a system emits \(2^n\) symbols with equal probability of all symbols then the random source entropy of the system is \(n\).

For an encryption algorithm the entropy must be closer to \(n\). If it is less than \(n\) the system can be cracked by entropy attack. For a gray scale (\(2^8\) symbols) image the random source entropy is 8 and is measured in bits. From Table 2 the entropy of AES and proposed system are closer to 8.

\[ H(s) = -\sum_{n=0}^{2^8} p(n) \times \log_2(p(n)) \]

For a grey scale image with \(2^8\) grey levels, entropy can reach maximum of 8 for ideal encryption.
After applying any image processing technique, the amount of distortion present in the processed image with reference to the original image is measured as peak signal to noise ratio in decibels (dB). The proposed system offered PSNR is found to be infinity between the original image and decrypted image. Which means the encryption technique considered will not lead to any loss of information when incorporated into compression.

The PSNR of an image with spatial resolution M x N is given by

$$PSNR = 10 \log_{10} \frac{\sum_{x=1}^{M} \sum_{y=1}^{N} x^2}{\sum_{x=1}^{M} \sum_{y=1}^{N} (IP(x,y) - ID(x,y))^2}$$

where $x$ is the peak sample value, for an 8-bit grey scale image $x$ is 255. IP and ID denote the plain image and the decrypted image respectively.

**Time Complexity**

The speed of execution depends on the implementation platform. When both AES and proposed system are executed in Matlab 2013 under same system the time taken by proposed system is 21.964 sec which is only 9.1% of the time taken by AES algorithm.

**Energy Consumption Analysis**

The energy consumption analysis is done in Atmel’s AVR Studio 20060421 for the target hardware platform ATmega128 microcontroller. The mica2 sensor mote is built with ATmega128 processor with an operating frequency of 8 MHz, and an active power consumption of 22 mW [5]. The simulation results of proposed encryption algorithm for processing 2x2 block is presented in Table 3 and compared with AES.

The code optimization level is set to “O3” in the AVR GCC compiler. The energy computation is carried out as in [2]. The choice of sample size 2x2 is chosen, because the encryption algorithm presented is to be incorporated with the compression system. The compressions system considers only the 2x2 transform coefficients including DC and coefficients around the DC term. The energy consumed by the encryption system 0.3548 µJ and takes only 0.0161ms.

<table>
<thead>
<tr>
<th>method</th>
<th>Code size (bytes)</th>
<th>Execution cycles</th>
<th>Execution time (ms)</th>
<th>Energy (µJ)</th>
</tr>
</thead>
<tbody>
<tr>
<td>AES</td>
<td>5738</td>
<td>11463</td>
<td>1.4328</td>
<td>31.523</td>
</tr>
<tr>
<td>Proposed</td>
<td>1000</td>
<td>129</td>
<td>0.0161</td>
<td>0.3548</td>
</tr>
</tbody>
</table>
5. Energy Consumption Analysis of the Secured Image Coder

The energy consumption analysis of the entire system presented is carried out as done in the previous section using AVR studio 20060421. The simulation results are compared with the work presented by Lee et al [6] in Table. 4. In Table 4 Column A denotes transformation and quantization cycles and column B denotes encryption cycle. Similarly column C denotes coding cycles and D denotes total number of cycles (D=A+B+C). It is noteworthy that the proposed low complex secured image coder takes only 0.26 ms for processing an 8x8 pixel from transformation to entropy coding. The energy consumption for the same sample is as low as only 5.63 µJ.

<table>
<thead>
<tr>
<th>Work</th>
<th>Execution cycles</th>
<th>Time in ms</th>
<th>Energy in µJ</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A</td>
<td>B</td>
<td>C</td>
</tr>
<tr>
<td>Lee et al [6]</td>
<td>11063</td>
<td>-</td>
<td>21504</td>
</tr>
<tr>
<td>Proposed</td>
<td>1548</td>
<td>129</td>
<td>369</td>
</tr>
</tbody>
</table>

Because of low complexity operations in all the stages of the system the energy consumption significantly/radically reduced.

6. Conclusion

The presented low complex secured image coding has reduced the energy consumption radically. The encryption algorithm implemented in the image coder possess all the qualities of the benchmark AES algorithm with very low energy consumption. Thus the described image coder promises secured image transmission with very little energy consumption under low bitrate platforms like WISN. The future work will be demonstrating the system in real hardware platform for better understanding of the real-time physical constraints.

References


