UDC [004.056+003.26](075.8)

Qiu Shifeng, Sadov V. S.

AN IMAGE STEGANOGRAPHY ALGORITHM BASED ON CHAOTIC SEQUENCE

Belarusian State University, Minsk, Belarus

Представлены результаты исследования стеганографического алгоритма на основе хаотической последовательности и матричного кодирования с минимальным изменением стеганографического контейнера.

1. Introduction in problem

Information hiding is one of the important means of information security. In the process of network communication and data storage, it can not only ensure the security of the confidential information itself, but also hide the fact that the secret information is being transmitted and stored. As the main technology in the field of information hiding, digital steganography usually uses image, video, audio, text and other carriers to hide information.

In the process of digital image steganography, at least three subjects are involved: secret message, cover image and stego image. By modifying the cover image, the secret message is embedded into the cover image in a way that is difficult for humans and programs to detect, and finally the stego image is obtained. The ways to modify the cover mainly include three research directions: spatial domain, frequency domain and mixed domain. Among them, the spatial domain steganography supports a large capacity to embed the secret message, so this paper focuses on the algorithm related to the spatial domain and one of the main problem: imperceptible.

Traditional algorithms include: LSB replacement, LSB matching and STC (Syndrome-Trellis Codes) encoding.The LSB (least significant bits) is a typical spatial domain algorithm developed earlier, which hides secret messages by directly changing the value of cover information. That is, the secret is embedded in the least significant bit of a pixel in the cover image. The advantages of LSB are good transparency, fast embedding speed and large capacity. The disadvantage is poor robustness and weak anti-attack ability. The LSBM algorithm is an improvement of LSB. When the embedded secret message bit is the same as the lowest bit of the pixel value, the pixel value remains unchanged. If it is different, one is randomly selected to be added or subtracted. This algorithm removes the statistical asymmetry introduced by LSB replacement.

In the field of communication, convolutional codes and Viterbi decoding algorithms can use fewer bits to represent more information bits. The STC encoding method is similar to such convolutional codec algorithm. The STC algorithm consists of an encoding method, a decoding method, and a parity check matrix. The check matrix of STC is a striped matrix arranged by sub-matrices, avoiding the construction of high-dimensional linear extraction equations, optimizing based on local properties, eliminating the impossible construction route in advance, and finally selecting all the local extraction equations, and choose distorting of the construction route (Number of embeddings or distortion functions define metrics relevant to steganographic security) with the lowest sum [1, 2].

2. Methodology of the Research

This paper proposes an algorithm combining chaotic sequence and STC coding. Firstly, we use logistic function to shuffle secret message, and then use the STC method to embed it into the cover image to obtain the stego image. The algorithm proposed has distinctive characteristics by using logical functions to generate chaotic sequence. Chaotic sequence has excellent pseudo-randomness and uniform distribution. Since the generation of chaotic sequence only depends on the initial value and generation parameters, it is reversible, and the extraction algorithm is easy to solve. The detailed steps are shown below in Figures 1 and 2.

Algorithm 1	Embedding algorithm

1: Input: Cover image C of size M × N and secret message SE of size m × n.
2: Output: Stego-image S.
3: Begin
4: Perform logistic function to generate chaotic number sequence J of size m×n×8.
5: Sort J by ascending order to generate a new sequence K.
6: Calculate Xi by the formula (Ki-1)/N +1.
7: Calculate Yi by the formula Ki mod N +1.
8: Shuffle SE by coordinates of Xi and Yi.
9: Perform STC encoding method to the suffled SE and C to get S.
10: End

Figure 1 – Embedding Algorithm

Algorithm 2	Extracting algorithm

1: Input: Stego-image S of size M × N.
2: Output: Secret message SE.
3: Begin
4: Perform STC decoding method to S to get suffled message SM.
5: Perform logistic function to generate chaotic number sequence J of size m×n×8.
6: Sort J by ascending order to generate a new sequence K.
7: Calculate Xi by the formula (Ki-1)/N +1.
8: Calculate Yi by the formula Ki mod N +1.
9: Shuffle SM by coordinates of Xi and Yi to get SE.
10: End

Figure 2 – Extracting Algorithm

3. Experiment of the Research

This experiment firstly embeds secret messages of different sizes into different *cover* images to obtain the corresponding *stego* images; and then generates "*diff image*" – the difference between *cover* and *stego*, visualized as the positions of modified pixels. Data set includes multiple groups of 24-bit BMP images. Image types are divided into three categories by style: real photos, cartoon patterns and map textures. The size of *cover* is 256×256 pixels. Four sizes of secret messages are used, which are 3.1%, 6.2% and 12.5% of the *cover*, respectively named *sm1*, *sm2*, and *sm4*. The results of the experiment are shown in Figure 3.

4. Conclusion

By comparing the visual effects of different covers and stegos in the experiment, the proposed algorithm has already well satisfied the imperceptibility in the case of large embedding capacity. By analyzing the embedding positions of secret messages of different sizes, it can be concluded that the secret is evenly distributed to most positions of the cover images, and at the same time it keeps unpredictable. Statistical results of LSB and proposed algorithm are shown in Table 1 below.

Table 1

Statistical Results of LSB and Proposed Algorithm

	id	size of message	bits modified by lsb	modified rate of lsb	bits modified by stc	modified rate of stc
0	1	196608	92502	0.470500	19648	0.099900
1	2	196608	92518	0.470600	21061	0.107100
2	3	196608	92321	0.469600	21110	0.107400
3	4	196608	91819	0.467000	20837	0.106000
4	5	196608	92342	0.469700	21117	0.107400
5	6	196608	92714	0.471600	20903	0.106300
6	7	196608	91997	0.467900	20899	0.106300
7	8	196608	92457	0.470300	21130	0.107500
8	9	196608	92105	0.468500	21149	0.107600

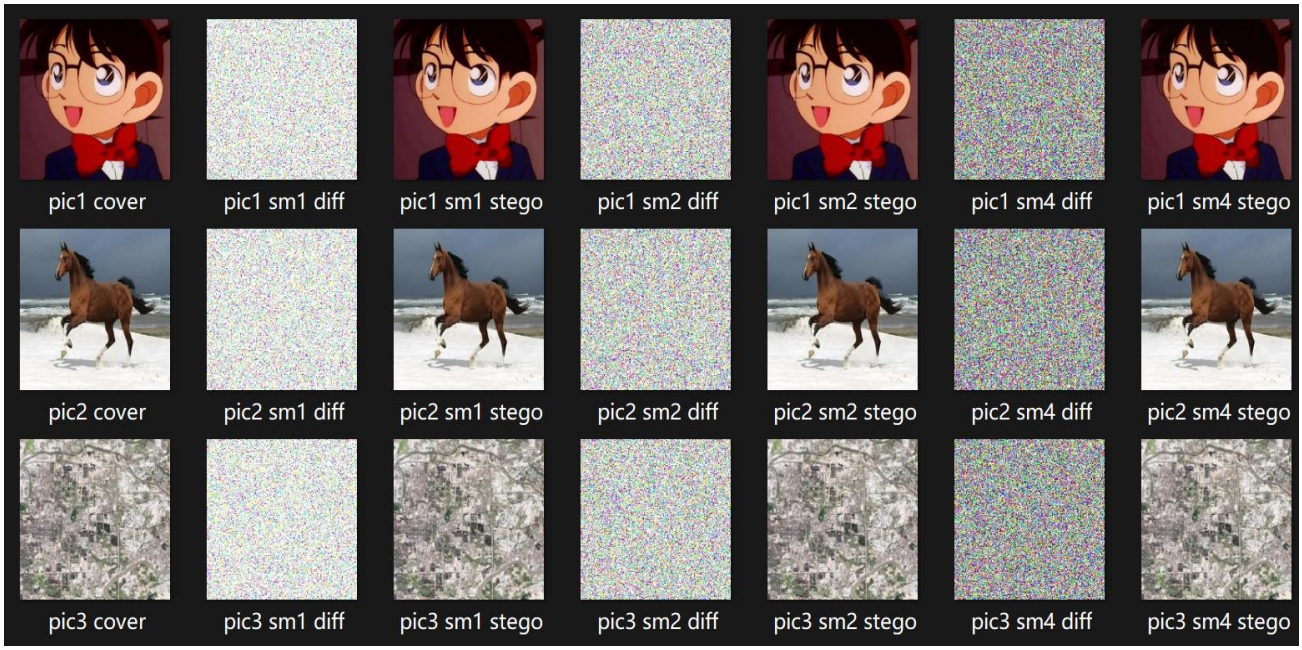

Figure 3 – Experimental Results of the Proposed Algorithm

Reference

1. Tomas Filler, Jan Judas, Jessica Fridrich "Minimizing Embedding Impact in Steganography using Trellis-Coded Quantization", Proc. SPIE, Electronic Imaging, Media Forensics and Security XII, San Jose, CA, January 18-20, 2010.

2. Westfeld A.F5-A steganographic algorithm: High capacity despite better steganalysis [C]. New York, Berlin, Heidellberg: Springer-Verlag, 2001.289-302.