Mühendislik Bilimleri ve Tasarım Dergisi
6(1), 87-92, 2018
e-ISSN: 1308-6693

Journal of Engineering Sciences and Design
DOI: 10.21923/jesd.319330

Araştırma Makalesi Research Article

GENERATING A STEGO-AUDIO DATA USING LSB TECHNIQUE AND ROBUSTNESS TEST

Arif KOYUN[1], Hüseyin Bilal MACİT[2]

[1] Suleyman Demirel University, Engineering Faculty, Dept. of Computer Engineering, Isparta, Türkiye
[2] Mehmet Akif Ersoy University, Tefenni MYO, Dept. of Computer Technologies, Burdur, Türkiye

Keywords

Steganography,
LSB,
MSE

Abstract

Steganography is the process of hiding and extracting a data into another data. A secret data can be carried inside another data without any doubt. No one is suspicious of an ordinary music, picture, text or video file. This work explains some steganography algorithms and focuses on hiding image data into an audio file. The LSB algorithm; which is usually used for hiding text data into image files is used to hide picture data in an audio data and the difference between the original audio data and the stego-audio data is shown. Steganography performance test methods have been described, and some artificial attacks applied to stego-audio. Robustness of hidden data is measured by the MSE technique after stego-attacks.

LSB TEKNİĞİ İLE STEGO-SES VERİSİ OLUŞTURMA VE SAĞLAMLIK TESTİ

Anahtar Kelimeler

Steganografi,
LSB,
MSE,

Öz

Steganografi; bir veriyi, başka bir verinin içine gizleyip çıkartma işlemidir. Gizli bir veri, hiç şüphe çekmeden başka bir verinin içerisinde taşınabilir. Hiç kimse sıradan bir müzik, resim, metin veya video dosyasından şüphelenmez. Bu çalışma; bazı steganografi algoritmalarını açıklamakta ve görüntü verilerini bir ses dosyasına gizleme üzerine odaklanmaktadır. Genellikle resim dosyaları içine metin verisi gizleme için kullanılan LSB algoritması; bir ses verisi içerisine resim verisi gizleme amacıyla kullanılmış ve orijinal ses verisi ile stego-ses verisi arasındaki fark gösterilmiştir. Steganografi performans testi yöntemleri açıklanmış, stego-ses verisine çeşitli yapay saldırılar yapılmıştır. Saldırıların sonucunda gizli verinin sağlamlığı MSE tekniği ile ölçülmüştür.

Alıntı / Cite

Koyun A., Macit H. B. (2018). Generating a stego-audio data using LSB technique and robustness test, *Journal of Engineering Sciences and Design*, 6(1), 87-92.

Yazar Kimliği / Author ID (ORCID Number)

A. KOYUN / 0000-0001-6701-363X
H. B. MACIT / 0000-0002-5325-5416

Başvuru Tarihi /Submission Date	06.06.2017
Revizyon Tarihi / Revision Date	22.09.2017
Kabul Tarihi / Accepted Date	19.01.2018
Yayım Tarihi / Published Date	28.03.2018

1. Introduction

Cryptography is a common security method. The transmitting data in the ciphering is coded according to the encryption algorithm in a way that the third person cannot understand in the transmission environment. Developing the encryption algorithms triggers the development of decryption algorithms. The most disadvantage of encryption system is that third persons can understand the data is encrypted because of its complicated and meaningless structure and suspicious of containing some important information (Durdu and Özcerit, 2015). Unlike cryptography, the encryption of information in steganography is not important but it's important how to carry it. The more hidden data is more safe. An ordinary picture or music file may contain highly confidential information in the absence of doubt. No one, including the recipient, doubts anything, without knowing where the data is stored and key information (Bilgin, 2013). Steganography is the art of amalgamating the secret message into another public message which may be text, audio or video file in a way that no one can know or imperceptible the existence of message (Meligy ao, 2015.).

The steganography word comes from the Greek words "steganos: hidden" and "graphy: drawing and writing". Steganography is a very ancient method of hiding data, dating back to the time of Ancient Greek and Heredot. During the Persian Wars, Heredot writes a secret message to a scooped head messenger. After the message is written, the messenger waits for the extension of his hair, then reaches the person waiting for the message, shaves his head again, so that the message appears. This method is the first steganography practice known (Şahin ao, 2006). Another historical example is a message sent by a German spy during World War II; "Apparently neutral's protest is thoroughly discounted and ignored. Isman hard hit. Blockade issue affects pretext for embargo on by-product, ejecting suets and vegetable oils." The secret key is to read second letters of every word. Transferred secret message is: "Pershing sails from NY June 1" (Dereli, 2010).

There are some common used steganography types. Most used type is text steganography method which is practically hiding a secret message into any other text. It has lost its importance because of its ease of decrypt and less carrying capacity. Image steganography is a popular way to hide information. An image file can carry a text message without any doubt. Audio steganography is a masking method that is based on small changes on audio that human ear can not understand clearly. A secret information can be hidden in an audio file. Video steganography is a common used technique because it's not easy to detect a secret information into video. Videos are basicly sequence of pictures with aa audio information. Secret information can be hidden into every frame of a video or its audio. Protocol steganography is used to embed information within the network protocols, such as TCP/IP. Information can be hidden in the header part of a TCP/IP packet and in some fields which are either optional or are never used (Aigal and Vasambekar, 2012).

Since it is difficult to keep data on the Internet, steganography has an important place in data communication. The main task of the steganography is to ensure that the secret message is not noticed. Moreover, even if the existence of the secret message is known, it is not easy to obtain it (Tunçer and Karakuzu, 2016).

Three basic features of steganography is important. The changes made on the cover data should not be in a perceived by the human senses. In case of detection, confidential communication occurs and can be destroyed or changed. The amount of data that can be hidden in the carrier data also has a significant effect, but the increase in the amount of data can bring the change in the cover data to a position where it can be perceived. This creates a dilemma between the first two principles, change and capacity. Used steganography algorithm must be resistible to attacks. For example; steganographic methods can be used together with encryption methods to create a more secure system (Bilgin, 2013). Figure 1 shows the general process which is valid for every steganography algorithm.

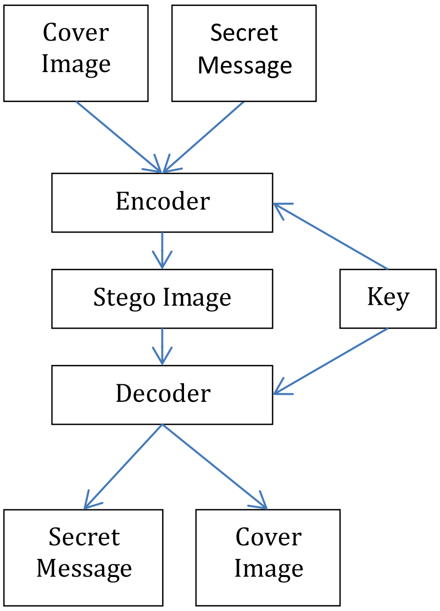


Figure 1. Basic Steganography Process

There are some common algorithms used on steganography. Each algorithm has advantages and disadvantages according to it's usage area as in shown in Table 2.

1.1. Least Significant Bit

This technique is based on adding a non-significant bit to the end of every bit string. When used with audio steganography, the ideal data transmission rate is 1 kbps per 1 kHz. The main disadvantage of LSB coding is its low embedding capacity (Bhattacharyya ao, 2011). In this technique LSB of binary sequence of each sample of digitized audio file is replaced with binary equivalent of secret message. First; audio signal is divided into samples as in shown in figure 2.

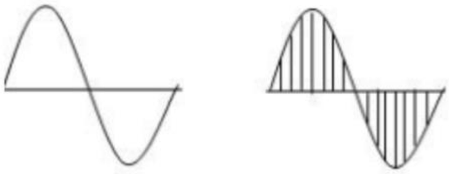


Figure 2: Sampling of the Sine Wave followed by Quantization Process.

Sampling technique followed by Quantization converts analog audio signal to digital binary sequence (Nehru and Dhar, 2012).

There is an example of LSB technique in Table 1. The data "01001001" is hidden in the values of the sample 8x8 bits data. Table 1 shows the before and after state of used data in hiding procedure (Tunçer and Karakuzu, 2016).

Table 1. LSB Example

Column	Original Data	Stego-data
1	01001001	0100100**0**
2	11101001	1110100**1**
3	01001010	0100101**0**
4	00101010	0010101**0**
5	01011011	0101101**1**
6	11000101	1100010**0**
7	01010001	0101000**0**
8	10110010	1011001**0**

Table 1 shows that the bit data just changed on columns 1, 6 and 7. It's clear that LSB technique doesn't cause much distortion in original data when used on text steganography.

There is another LSB algorithm that is called "Last 3 LSB" which is mostly used to hide ASCII codes into 8 bit data strings. Last 3 LSB is mostly used in image files that have 3x8 color definition bits for each pixel.

This algorithms works as:
Step 1. Add "zero bit" in front of the binary ASCII data
Step 2. Divide the 9 bit binary ASCII string into three parts.
Step 3. Read Red, Green and Blue binary values of next pixel
Step 4. Change all last 3 binary values by divided 3 bit ASCII values

Weakness of this technique is too much distortion to carrier image. But it's widely used because of ease of implementation and speed.

1.2. Echo Hiding

Echo hiding method embeds data into audio signals by introducing a short echo to the host signal. The nature of the echo is a resonance added to the host audio. Therefore, the problem of the HAS (Human Auditory System) sensitivity to the additive noise is avoided (Naidu ao, 2016).

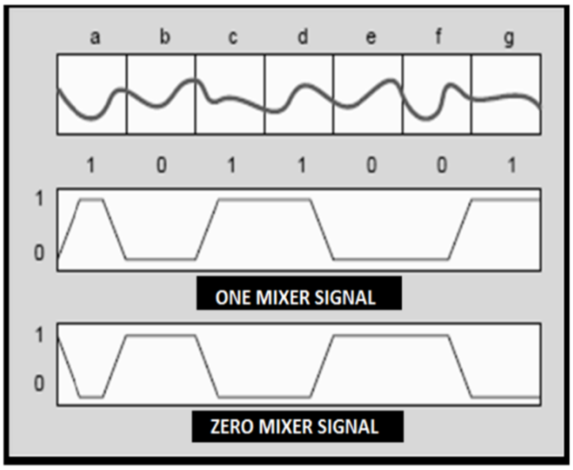


Figure 3. Echo Hiding Process (Bhattacharyya ao, 2011)

Figure 3 shows a signal converted from analog to digital with one mixer and zero mixer show which are negative of each other. In echo hiding; method information is embedded into an audio file by inducing an echo into the discrete signal. To extract the secret message from the final stego-audio signal, the receiver must be able to break up the signal into the same block sequence used during the encoding process (Bhattacharyya ao, 2011).

1.3. Spread Spectrum

Spread spectrum technique spreads hidden data through the frequency spectrum (Naidu ao, 2016). Methodology attempts to spread the secret information across the audio signal's frequency spectrum as much as possible (Bhattacharyya ao, 2011). There are many other rarely used techniques that are not mentioned, like phase coding, tone insertion, cepstral coding in audio steganography.
Audio steganography process is more difficult than text or image steganography. Audio Steganography methods can embed any messages in WAV, and even MP3 sound files (Singh, 2016).]. The Waveform Audio File Format (WAV) is a headerless, non-compressed audio file developed by IBM and Microsoft and takes a lot of disk space.

Table 2. Comparison of Audio Steganography Techniques (Singh, 2016).

Method	Advantages	Disadvantages	Hide Rate
LSB	Easy to hide information with high bit rate	Easy to extract and destroy hidden information	16Kbps
Echo Hiding	Robust against zipping algorithms	Low carriage capacity	40-50bps
Spread Sprectrum	Provides better robustness	Vulnerable to time scale modification	20bps

2. Material and Method

This work focuses on hiding a PNG formatted picture file which contains a letter into a WAV formatted audio file. The audio data used is in wave format, 123KB sized, has 8000 samples per second and each sample is composed of 16 bits of data as shown in time domain in Figure 5. Matlab software is used for coding. The secret data used to carry is a png-formatted, 50x50 pixels and 648 bytes sized image as shown in Figure 4.

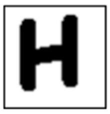


Figure 4. Secret message image

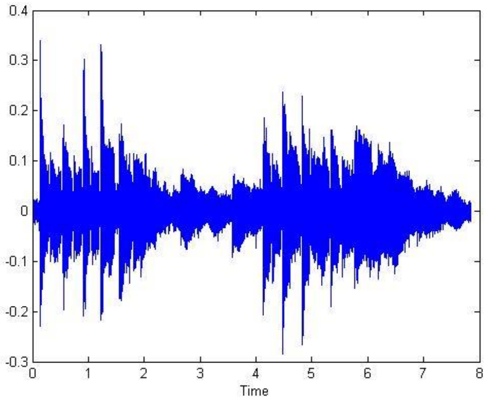


Figure 5. Original Audio Time Domain

The algorithm works on a binary based method LSB. In the embedding process, first the carrier data converts into a binary string. Then the secret image file converts into a binary string. Method divides carrier binary string to segments which are equal to secret image pixel size and a for loop runs to embed the secret message's every bit into every segments last bit. To do this; it must first calculate the length of carrier string. After embedding secret image binary string, algorithm transforms new stego binary string to a wave file which is named as stego-data or stego-audio, shown in time domain in Figure 6. Pseudo code is given below for step by step data embedding process.

```
{
        f_original ← carrier file
        f_original_bin ← binary string(f_original)
        secretmessage ← secret message image
        secretmessage _bin ← binary string(secretmessage)
        sm_length ← length(secretmessage_bin)
        for (i=1 to sm_length)
        {
                sm_str(i) ← f_original_bin(i)
        }
        f_stego ← bin2dec(f_original_bin)
        f_stego.wav ← f_stego
}
```

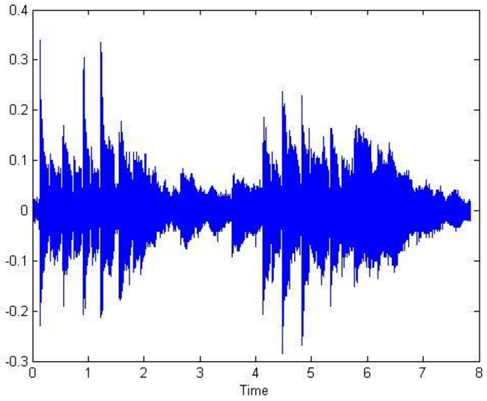


Figure 6. Stego-Audio Time Domain

Extracting process is similar with embedding sequence. First, the carrier data converts into a binary string. Extracter must know the secret data size. This algorithm uses a square image, so the image must be square root of binary string. After identifying image size, carrier binary string divides into parts each size is equal to pixel size of image. Each part is called as a segment. Then, method collects every last bit of a segment which are called as secret data. Last step is reshaping image using this collecting data. Pseudo code is given below for step by step data extracting process.

```
{
        f_stego ← carrier file
        f_original_binary ← binary string(f_stego)
        imagesize ← sqrt(pixelsize)
        for (i=1 to imagesize)
        {
                sm_str(i) ← f_original_binary(i)
        }
        secretmessage ← reshape(sm_str, imagesize * imagesize)
        show(secretmessage)
}
```

2.3. Robustness

There are too many steganography techniques. Efficiency depends on various standarts. These are transparency, capacity, robustness and complexity of the algorithm. Robustness is the most important feature in steganography (Rababah and Abdulgader, 2011).

There are critical values that represent the power of steganography algorithm: MSE (Mean Squared Error) and SNR (Signal to Noise Ratio) (Tuncer ao, 2011).

MSE is a way to represent the difference between stego-audio and original audio. MSE is equal to zero when two audios are the same. Suppose that $x=\{x_i|i=1,2,\cdots,N\}$ and $y=\{y_i|i=1,2,\ldots,N\}$ are two finite-length, discrete signals (visual image, audio image etc.), where N is the number of signal samples (pixels, if the signals are images) and x_i and y_i are the values of the i'th samples in x and y, respectively. The MSE between the signals is (Wang and Bovik, 2009)

$$\mathrm{MSE}(\mathrm{x},\mathrm{y}) = \frac{1}{N}\sum_{i=1}^{N}(x_i - y_i)^2 \tag{1}$$

The MSE result calculated via Matlab for this work is 0,5422.

Another criteria for robustness calculation is SNR (Signal to Noise Ratio), which is a form that gives an idea of the distortion of a given pattern with noise applied and the distortion between the original pattern. For each pixel, the differences are calculated separately to give the sum of the ratios of the squares. The smaller the SNR value means the greater distortion in the pattern. The SNR value approaches 0 when the distortion in the pattern goes to infinity.

$$SNR = 10log_{10}\left(\frac{\sum_{i=1}^{N} f(i)^2}{\sum_{i=1}^{N}(f(i) - g(i))^2}\right) \tag{2}$$

The SNR result calculated via Matlab for this work is 26,7739.

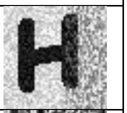
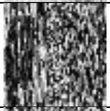
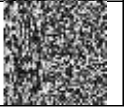
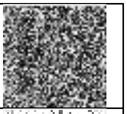
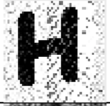
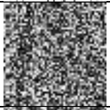
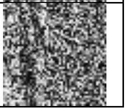
A steganographic algorithm must be resistible to attacks. The first goal of a steganalist is to find out existence of secret information in the image (Olcay and Saran, 2013). Stego-attacks are divided into 5 categories; stego attack only, known cover attack, known message attack, selected stego attack and selected message attack (ZhangT ao, 2010).

Success of a steganographic algorithm is related to many criteria. The ratio of secret data size to carrier size, distortion in original pattern can be calculated with MSE and SNR are mentioned. This paper used MSE method for robustness calculation.

3. Results

This paper presents different audio steganography techniques and their approaches and focuses on LSB algorithm. LSB algorithnm is mostly used on image steganography but this paper runs LSB algorithm over an audio file.

Table 3. Robustness Results of Stego-audio

Attack type	**Similarity (MSE)**	**Extracted image**
50 percent echo added after 0,5. second of stego-audio	%92,49	
20hz high pass filter applied	%42,01	
100hz high pass filter applied	%39,97	
3000hz low pass filter applied	%45,13	
4000hz low pass filter applied	%94,91	
%10 timeline extension	%40,31	
%10 timeline shortening	%43,01	

MSE calculation is represented and runs to measure the difference between stego and original audio signal. Some audio distortion techniques are applied to stego-audio file to see robustness of the method. MSE is also calculates to see the difference between images before and after audio distortion.

The paper represents that, LSB method has an advantage of high carrying capacity when used with an audio image. But it has a disadvantage of low robustness capacity proven with MSE results after distortions as in shown in Table 3. This paper also makes a demonstration of robustness of stego-audio data exposed to some external influences like conversion between analog to digital form and reverse and how it may cause huge loss of carrying data.

Conflict of Interest

No conflict of interest was declared by the authors.

References

Durdu A, Özcerit A. T. 2015. Güvenli İletişim İçin Yeni Bir Veri Gizleme Algoritması , Uluslararası Bilgi Güvenliği Mühendisliği Dergisi, Cilt:1, No:1, 32-36p.

Bilgin M. 2013. Steganografi, XV. Akademik Bilişim Konferansı Bildirileri, Akdeniz Üniversitesi, Antalya. 125-128p.

Meligy A.M. Nasef M.M. Eid F.T. 2015. An Efficient Method to Audio Steganography based on Modification of Least Significant Bit Technique using Random Keys, I. J. Computer Network and Information Security, 24-29p.

Şahin A.Buluş E. Sakallı M.T. 2006. 24-Bit Renkli Resimler Üzerinde En Önemsiz Bite Ekleme Yöntemini Kullanarak Bilgi Gizleme, Trakya Univ J Sci, 7. ISSN 1305-6468. 17-22p.

Dereli, Ç.2010. Dilbilimsel Steganografi Yöntemleri üzerine bir Araştırma, Yüksek Lisans Tezi, Ege Üniversitesi, İzmir.

Tunçer S. Karakuzu C. 2016. Veri Güvenliğini Artırmak Amacıyla Bilgiyi Şifreleme ve Steganografik Yöntemlerle Görüntüye Gizleme, EEB 2016 Elektrik-Elektronik ve Bilgisayar Sempozyumu, Tokat.183-187p.

Aigal P. Vasambekar P.2012. Hiding Data in Wave Files, International Conference in Recent Trends in Information Technology and Computer Science (ICRTITCS), 20-24p.

Bhattacharyya S. Kundu A. Sanyal G.2011. A Novel Audio Steganography Technique by M16MA, International Journal of Computer Applications (0975 – 8887) Volume 30, No.8. 26-34p.

Naidu T.R.K. Kumar P.G. Prasad T.G. 2016. Overview of Digital Audio Steganography Techniques, International Journal of Emerging Technologies and Engineering (IJETE) Volume 3 Issue 7, ISSN 2348 – 8050. 62-66p.

Singh P.2016. A Comparative Study of Audio Steganography Techniques, International Research Journal of Engineering and Technology, Volume: 03 Issue: 04. 580-585p

Tuncer T. Doğan Ş. Avcı E. 2011. Renkli İmgelerde Gizlenen Verilerin Görsel Ataklara Karşı Dayanıklılığının Tespiti İçin Bir Steganografi Uygulaması, 6th International Advanced Technologies Symposium,Elazığ, Turkey. (IATS'11). 32-36p.

Olcay C. Saran N. 2013. İmge İçine Bilgi Gizlemede Kullanılan LSB Yöntemlerinin Karşılaştırması, Çankaya University Journal of Science and Engineering Volume 10, No. 1, 17–32p.

Wang Z. Bovik A. C. 2009. Mean Squared Error: Love It or Leave It?, IEEE Signal Processing Magazine[98].

ZhangT. Li W. Zhang Y. ZhengE. PingX.2010. Steganalysis of LSB matching based on statistical modeling of pixel difference distributions, Information Sciences, Volume 180, Issue 23. 4685-4694p.

Rababah, A. ve Abdulgader, U., New technique for hiding data in audio file. Int. J. Comput. Sci. Network Security, 11(4), 2011.

Nehru, G. ve Dhar, P., A Detailed Look of Audio Steganography Technique susing LSB and Genetic Algorithm Approach, IJCSI Vol.9 1(2), 2012.

ADD-A

Data embedding Matlab code:

```
[f_original, f] = audioread (f_original.wav')
f_original = uint8(255*(f_original + 0.5));
f_original bin  = dec2bin(f_original, 8);
secretmessage = imread(secretmessage.png');
secretmessage_bin =dec2bin(secretmessage(:), 8);
sm_length = length(secretmessage(:))*8
for i = 1:sm_length
f_original_bin(i, 8) = dec2bin(sm_str(i));
end;
f_stego  = bin2dec(f_original _bin);
f_stego  = (double(f_stego)/255 - 0.5);
audiowrite(f_stego.wav', f_stego, f)
```

Data extracting Matlab code:

```
f_stego = audioread (f_stego.wav');
f_stego = uint8(255*( f_stego + 0.5));
f_original_bin  = dec2bin(f_stego, 8);
imagesize = sqrt(pixelsize);
for i = 1:pixelsize
sm_str(i, :) = bin2dec(secretmessage_bin(i, :));
end;
secretmessage = reshape(sm_str, imagesize ,
imagesize);
imshow(secretmessage);
```

Calculation of mean square error Matlab code:

```
[x,freq]=wavread(f_original.wav');
[y,freq]=wavread(f_stego.wav');
MSE_Result=(sum((x-y).^2))
```

Calculation of signal to noise ratio Matlab code:

```
SNR_Result = 10*log10((sum(y.^2)))/MSE_Result)
```