RESİM İÇİNE YAZI GİZLENMESİ AMACIYLA KULLANILAN LSB EKLEME YÖNTEMİNİN SHUFFLE ALGORİTMASIYLA İYİLEŞİTİRİLMESİ

E.Murat Esin
Maltepe Üniversitesi
Elektronik Mühendisliği Bölümü
emesin@maltepe.edu.tr

Erdal GÜVENOĞLU
Maltepe Üniversitesi
Bilgisayar Mühendisliği Bölümü
erdalg@maltepe.edu.tr

ÖZET

Günümüzde metin iletimi için genel kullanıma açık ortamların yaygınlıkla kullanılmaktadır. Bu durum gizliliği olan verilerin güvenliğinin sağlanması için şifrelenmesinin önemini arttırmaktadır. Sayısal ortamda bulunan verilerin güvenliğini sağlamak için çeşitli şifreleme ve steganografi teknikleri geliştirilmiştir. Şifreleme mesajın içeriğinin korunmasını amaçlarken, steganografi mesajın varlığının gizlenmesi ile ilgilenmektedir. Bu çalışmada LSB (En Önemsiz Bite) ekleme yöntemine Shuffle Algoritmasının uygulanması ile gerçekleştirilen yeni bir steganografi ve steganaliz yöntemi tanıtılmaktadır.

Anahtar Kelimeler: Bilgi Gizleme, Steganografi, Steganaliz, LSB Yöntemi.

ABSTRACT

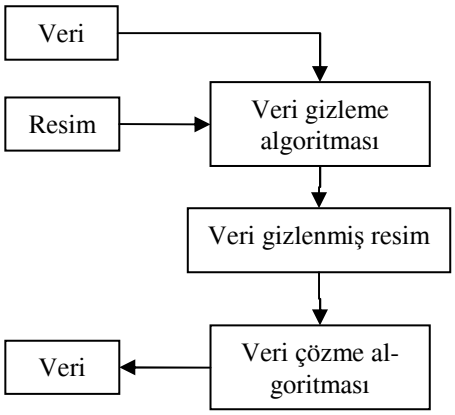
Nowadays, the public data communication mediums are widely used for text transfer. This case incriases the importance of encryption of texts that must be secret to provide their security. There are a lot of methods for providing security of the texts in the digital medium. While encryption aims to quard the coverage of the message, steganography interests with hiding of the message. In this work, a new steganography and steganalysis method that is handeled with application of Shuffle Algorithm to the Least Significant Bit (LSB) Insertion Methods is introduced.

Keywords: *Data hiding, Steganography, Steganalaysis LSB Insertion Method*

1. GİRİŞ

Günümüz teknolojisinin gelişmesiyle dijital ortamda önemli verilerin gizliliği büyük önem kazanmıştır. Iletilecek metnin şifrelenmesi için kullanılan iki temel yaklaşımdan ilki metnin kendi içierisinde karıştırılarak anlaşılmaz hale getirilmesi (kriptografi), ikincisi ise metnin bir başka yapı içerisine gömülerek (steganografi) gizlenmesidir. İkincisi için kullanılan yöntemlerden birisi de steganografidir. Steganografi kelimesi Yunanca "steganos: gizli, saklı" ve "grafi: çizim yada yazım" kelimelerinden gelmektedir. Bu yaklaşım, bir nesnenin içerisine bir verinin gizlenmesi olarak tanımlanmaktadır. Taşınmak istenen mesajın bir başka ortamda saklanarak üçüncü şahısların iletilen mesajın varlığından haberdar olması engellenmektedir. Bu özellik kriptografi ile steganografi arasındaki en temel farkı oluşturmaktadır.

Bu yaklaşımla ses, sayısal resim, video görüntüleri üzerine metin saklanabilir. Görüntü dosyaları içerisine saklanacak veriler metin dosyası olabileceği gibi, herhangi bir görüntü içerisine gizlenmiş bir başka görüntü dosyası da olabilir. Bu yaklaşımda içine bilgi gizlenen



Şekil 1. Steganografi sistem yapısı

ortama örtü verisi (cover-data), oluşan ortamada stego-metin (stego-text) veya stego-nesnesi (stego-object) denmektedir.

Son yıllarda sayısal nesneler üzerinde steganografi uygulamaları sıklıkla kullanılmaya başlanmıştır. Steganografi, "dilbilim steganografi" ve "teknik steganografi" olarak kendi içinde ikiye ayrılmaktadır. Dilbilim steganografi, taşıyıcı verinin metin (text) olduğu steganografi koludur. Teknik steganografi ise birçok konuyu içine almaktadır. Bunlar; görünmez mürekkep, gizli yerler, microdot'lar ve bilgisayar ta-

banlı yöntemler gibi başlıklar altında toplanmaktadır.[4]

Görüntü steganografide bilgiyi resmin içine gizlemek için çeşitli yöntemler vardır. Bunlar şu şekilde sınıflandırılabilir.

- En önemsiz bite ekleme
- Maskeleme ve Filtreleme
- Algoritmalar ve Dönüşümler [4].

Steganografik bir algoritma incelendiğinde üç temel unsur göz önüne alınmaktadır

- Değişimin fark edilememesi
- Saklanabilecek veri miktarı
- Dayanıklılık

Resim üzerinde gerçekleştirilen değişiklikler insan gözü tarafından algılanamamalıdır. Aksi halde en azından bir gizli metin iletilmekte olduğu anlaşılacaktır. Bu durumda üçüncü kişilerin içerisinde gizli veri olan resim üzerinde işlemler yapma olasılığı vardır. Steganografik yöntemlerin bazı sınırlar içerisindede olsa bu tür saldırılara karşı bir dayanıklılık göstermesi gerekir.

Bugüne kadar yapılan steganografi çalışmaları şu şekilde sınıflandırılabilir.

- Yer değiştirmeye dayalı yöntemler
- İşaret işlemeye dayanan yöntemler
- Spektrum yayılmasına dayanan yöntemler
- İstatistiksel yöntemler

Yer değiştirmeye dayalı yöntemlerde, temel olarak resim dosyasında piksel renklerini temsil eden değerler üzerinde çalışılmaktadır. Pikselin rengi bir byte ile ifade edilmektedir. Bu byte'ların en düşük bitlerinin değişmesi resim görüntüsünde gözle farkedilmesi zor bir fark yaratmaktadır. Metne ait karakterleri temsil eden byte'ların her bir bitinin farklı bir pikselin en önemsiz bitine kaydedilmesi "en önemsiz bite ekleme yöntemi" (LSB - Least Significant Bit Insertion Methods) olarak bilinmektedir. Sonuçta ortaya çıkan resim dosyasında renk değerleri ya olduğu gibi kalır ya bir artar yada azalır. Her üç durumda da insan gözü tarafından algılanamamaktadır. Burada veri bit değerleri sırasıyla eklenmektedir[1].

İşaret işlemeye dayanan yöntemlerde, resim sıkıştırma algoritmaları kullanılmaktadır. Tüm sıkıştırma algoritmaları insan gözünün filtreleme özelliğini kullanmaktadırlar. İnsan gözü belli bir frekanstan sonrasındaki renk değişimlerini algılayamamaktadır. Dolayısıyla, asıl resimdeki renk değerleri frekans düzlemine taşınabilir. Frekans düzleminde çeşitli katsayılar oluşacaktır. Ancak bu katsayılar kullanılarak ters işlem sonrası tekrar asıl resmi elde etmek için sonsuz sayıda frekans bileşeninden faydalanmak gerekmektedir[2].

Spektrum yayılmasına dayanan yöntemlerde, gönderilmek istenen mesaj ihtiyaç duyduğu frekans bandından çok daha fazlasına dağıtılmaktadır. Üçüncü bir kişi araya girip bir yada birden fazla frekans bandında bozulmalara neden olsa bile, alıcı geri kalan frekans bantlarındaki bilgiler ile asıl mesajı elde edebilmektedir. [3]

İstatistiksel yöntemlerde ise, bazı istatistiksel bilgilerin değiştirilmesi ile alıcıya gizli bir mesaj iletilebilmektedir. Burada bir hipotez fonksiyonu belirlenir ve bu hipotez fonksiyonuna parametre olarak resim veya resmin bir kısmı gönderilir. Bu fonksiyonun geri döndürdüğü değer ile fonksiyondaki istatistiksel özelliğin değişip değişmediğine bakılır. Burada karşılaşılan sorun ise, uygun bir hipotez fonksiyonunun bulunmamasıdır. Ancak en büyük sorun gönderilecek gizli bilgi miktarıdır. Burada resim dosyası başına sadece bir adet gizli bilgi gönderilebilmekte, alıcının bilinen fonksiyonu kullanarak metnin kalan karakterlerini türetmesi beklenmektedir.

Tüm bu yöntemler göz önüne alındığında en önemsiz bite ekleme (LSB) yönteminin dikkatli uygulanması durumunda herhangi bir veri kaybına neden olmadığı ve gönderilen gizli veri miktarının çok daha fazla olabileceği görülmektedir. Ancak resmin üçüncü bir kişinin eline geçmesi durumunda algoritma yapısı bilindiğinden gizli metin çözülebilmektedir.

Bu çalışmada en önemsiz bite ekleme yapılırken sıralı ve gruplandırılarak değil de belli bir anahtar dizisi üretilerek veri gizleme işlemine bir yenilik getirilmektedir.

2. EN ÖNEMSİZ BİTE (LSB) EKLEME YÖNTEMİ

En önemsiz bite ekleme yöntemi yaygın olarak kullanılan ve uygulanması basit bir yöntemdir.

Bu yöntemde; resmi oluşturan piksellerin her byte'nın en önemsiz bitinin yerine gizlenmesini istediğimiz verinin bitleri sırasıyla verinin başlangıcından itibaren birer birer yerleştirilmektedir.[4]

Örneğin; resmin piksel değerlerinin binary karşılığının

10001001 11101001 11101001 10011011
10011011 10001001 00011111 00011101

şeklinde olduğunu düşünelim. Buna S (01010011) karakterini eklersek resim

1000100**0** 1110100**1** 1110100**0** 1001101**1**
1001101**0** 1000100**0** 0001111**1** 0001110**1**

şeklinde değiştirilmiş olacaktır.

Ancak, ekleme sırayla yapıldığında içine metin gizlenmiş bir resim üçüncü şahıslar tarafından kolaylıkla

çözülebilmektedir. Rasgele ekleme işleminde ise, aynı rastgele sayının birden fazla kere üretilme olasılığı olduğundan, aynı piksel grubuna birden fazla kere değişiklik yapılma olasılığı var demektir. Bu durumda gizlenmiş metin çözüldüğünde karakter kayıplarına rastlanabilmektedir.

3. SHUFFLE ALGORİTMASI

Yukarıda bahsedilen karakter kayıplarının ortadan kaldırılması için aynı pozisyonun bir fazla kere kullanımının önlenmesi gerekir. Böyle karıştırmalar için Shuffle algoritmaları kullanılmaktadır.

Shuffle algoritmaları, bir dizi elemanlarının kendi içerisinde yer değiştirmek suretiyle karıştırılması için tasarlanmıştır. Metni oluşturan karakterlerin

$$X = \{x_1, x_2, x_3, .., x_i, .., x_n\}$$

şeklinde sıralı dizi oluşturduğunu varsayalım. Shuffle algoritması, bu dizi elemanlarının yerlerinin belli bir rastgelelik fonksiyonuna bağlı olarak değiştirilmesi esasına dayanmaktadır. Her dizi elemanı sadece bir defa kullanılmakta ve dizi elemanlarının tekrarı olmamaktadır. Algoritma her çalıştırıldığında karışmış yeni bir dizi elde edilmektedir.

Bu algoritma ile karıştırılacak diziye X, karıştırılma sonunda elde edilen yeni diziye ise X' adını verelim.

$X = \{x_1, x_2, x_3, .., x_i, .., x_n\}$ dizisi için bir Shuffle algoritması aşağıda verilmektedir.

```
for  i = n downto 1 do
begin
        r = random(i) + 1;
        swap(x_i, x_r);
end
```

Örneğin; n = 10 olduğunu düşünelim. *X* dizisi

$$X = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}\}$$

olmaktadır. Elde edilecek olan *X'* dizilerinden mümkün bir tanesi aşağıdaki şekilde olacaktır.

$$X' = \{x_2, x_{10}, x_5, x_1, x_4, x_7, x_6, x_9, x_3, x_8\}$$

Algoritma her çalıştırıldığında yeni bir *X'* dizisi elde edilecektir.

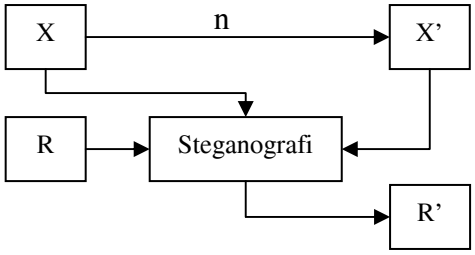
4. LSB EKLEME YÖNTEMİNİN SHUFFLE ALGORİTMASI İLE İYİLEŞTİLMESİ

LSB ekleme yöntemi resim içine bir metnin yerleştirilmesi amacıyla kullanılmaktadır. Literatürde bilinen şekliyle metnin karakterlerinin her bir biti, bir resmin ardışık satır veya sütun piksellerini temsil eden byte'ların en önemsiz bitlerine sıra ile yerleştirilir. Bu yöntem, ek bir rastgelelik içermediğinden gizli mesajın elde edilmesi işlemi oldukça kolaydır. Ancak, resmin içinde bir metin gizlendiği anlaşıldığında orijinal metnin elde edilmesi de çok kolaydır.

Bu bakımdan metni resmin içine bir anahtar dizisine bağlı olarak saklamak daha güvenlidir. Bu çalışmada, LSB yöntemiyle yerleştirilecek karakterlerin yerleştirilme pozisyonlarının belirlenmesi için bir shuffle algoritması kullanılarak karıştırma sağlanması yöntemi gösterilmektedir.

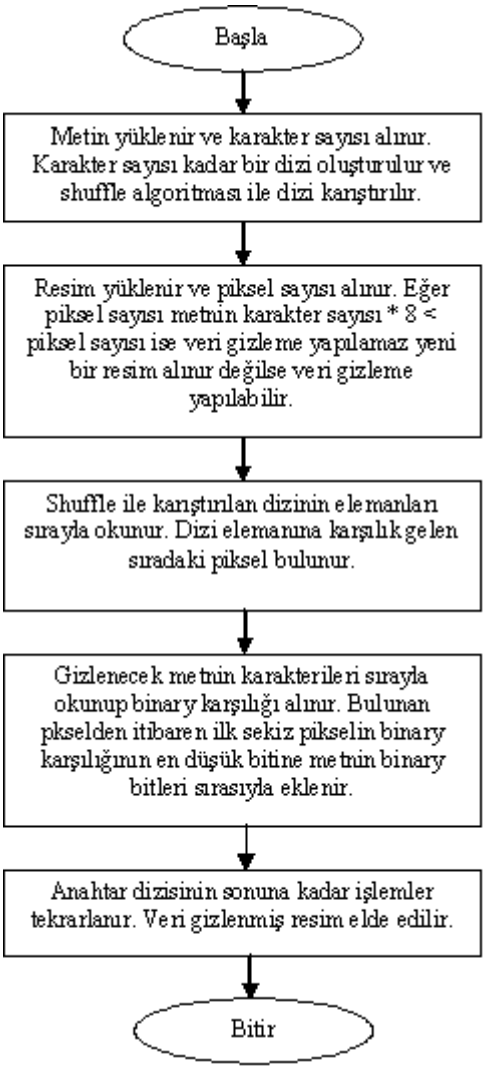
Bu yöntemde, yukarıda verilen Shuffle algoritmasıyla elde edilen *X'* dizisinin elemanları verinin gizleneceği pozisyonları belirlemektedir. O yüzden X' dizisine anahtar adı verilmektedir. Gizlenecek karakterin bitleri, resim dizinin i inci pozisyondan itibaren ardışık 8 pikselinin LSB bitlerinin yerini almaktadır. X' dizi elemanlarının tekrarı olmadığından resim piksellerinin üst üste değiştirilmesi olasılığı ortadan kalkmaktadır. Gizlenecek olan veri sekiz bitten oluştuğuna göre, n karakterden oluşmuş metni gizlemek için n*8 piksele ihtiyaç vardır. Dolayısıyla resmin piksel sayısı bu çarpımdan büyük olmalıdır. Aynı X' anahtarı resmin içine yerleştirilmiş metnin yeniden elde edilmesi amacıyla da kullanılmaktadır.

X dizisi gizlenecek metnin kendisini, X' dizisi X dizisinin eleman sayısına bakılarak elde edilmiş anahtar dizisini, R orijinal resmi temsil eden dizi, R' ise içine metin gizlenmiş resmi temsil eden dizi olsun. Bu durumda yöntemin gizleme blok diyagramı Şekil 2 de görülmektedir.



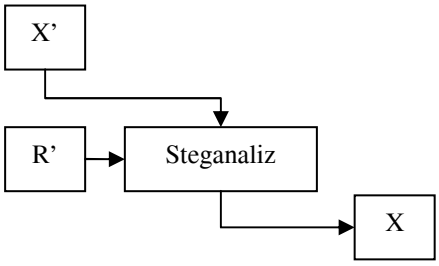
Şekil 2: Veri gizleme blok diyagramı

Bu yöntemin şifreleme akış şeması ise Şekil 3'te gösterildiği gibidir.



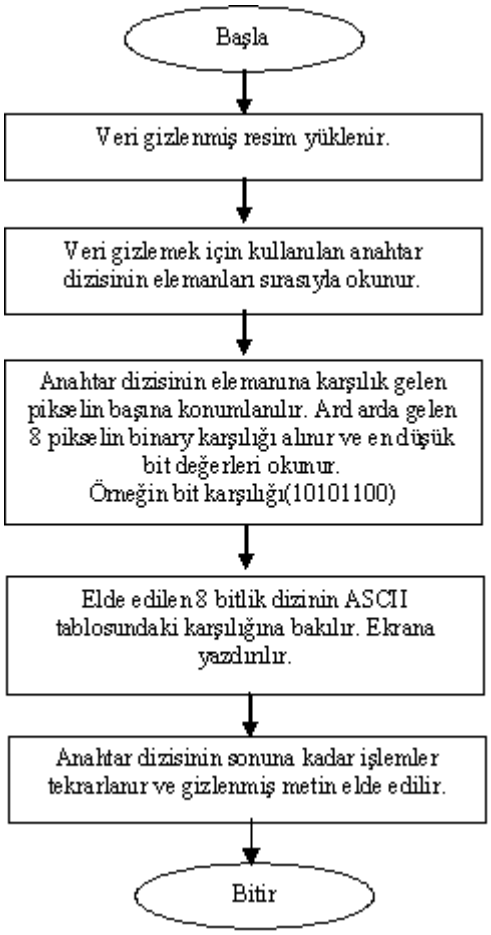
Şekil 3. Shuffle kullanılarak veri gizleme yönteminin akış şeması

Bu algoritmaya göre gizlenmiş metnin yeniden elde edilebilmesi için ayrıştırma blok diyagramı Şekil 4 de verilmiştir.



Şekil 4: Veri çözmenin blok diyagramı

Gizlenmiş olan veriyi çözme işleminin akış şeması da Şekil 5'te gösterilmiştir.



Şekil 5: Veri çözme işleminin akış şeması

5. TARTIŞMA

Yaygın olarak kullanılan LSB yönteminin temel avantajı, her bir karaktere ait bitlerin ardışık 8 pikseli temsil eden byte'ların en önemsiz bitlerinin yerine yazılmasıyla resim üzerinde belirlenmesi zor bir değişiklik yaratması idi.

Shuffle algoritmaları ise, gizlenecek metne ait karakterlerin karıştırılmasını sağlayarak çözülmesini zorlaştırmakta idi.

Bu çalışmada kullanılan yöntem, her iki yaklaşımın olumlu taraflarını birleştirerek hem resmin mümkün olduğunca az tahrifata uğrayarak üzerine metin yerleştirildiğinin anlaşılmasını ve hem de bu anlaşılsa bile anahtar dizi olmaksızın gizlenen metnin çözülmesinin zorlaştırılmasını sağlamaktadır.

Diğer taraftan, Shuffle algoritmasının özelliğinden dolayı metinden karakter kayıplarının olması riski de ortadan kalkmaktadır.
Anahtar değeri tek olduğu için her veri farklı bir pozisyondaki piksel ile onun ardından gelen 7 pikselin en önemsiz bitlerine yerleşmekte ve veri kayıpları oluşmamaktadır. İçine metin gizlenmiş resmin üçüncü şahısların eline geçmesi durumunda anahtar dizisi olma-

dan verinin çözülmesi, tek başına kullanılan LSB yönteminin kullanıldığı duruma göre daha zordur. Diğer yandan anahtar veri uzunluğu bilinerek üretildiğinden gizlenmiş metnin uzunluğunun bilinmediği hallerde deneme yoluyla anahtar üretmek daha da zorlaşmaktadır.

Zira, anahtarın maksimum uzunluğu n_{max} = R/8 olacağından, doğru anahtarın bir denemede bulunma olasılığının;

$$P = 1/\sum_{i=1}^{n_{\max}} (n_{\max} - i)!$$

olacağı açıktır.

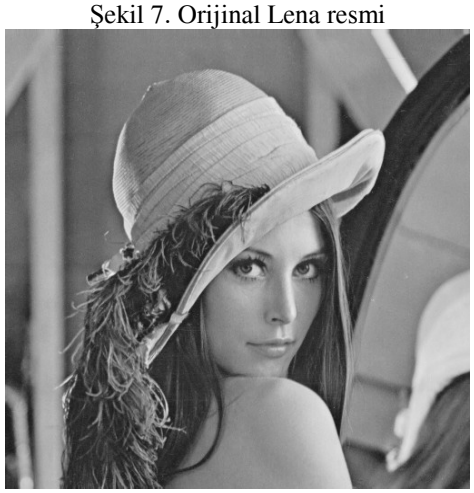
Örneğin 512x512 piksel boyutunda bir resmin içerisine veri gizlendiğini varsayalım. İçerisine gizlenmiş olan verinin uzunluğu bilinemediğinden;
maksimum anahtar uzunluğu (512*512)/8)= 32768 olacaktır. Diğer yandan anahtar uzunluğunun örneğin 32768 olduğunu düşünürsek, 32768! farklı anahtar permutasyonu olacaktır. Fakat doğal olarak daha kısa bir metin yerleştirilme olasılığı da vardır. Bu durumda Doğru anahtar bulununcaya kadar geriye kalan kısımlarda daima her sekiz bit için anlamsız karakterler üretilecektir. Aslında metin içinde yer almayan bu karakterlerin uydurulması çözümü karmaşıklaştıran bir başka etkendir.

Sonuçları göstermek üzere Şekil 6 da verilen metni gizlemek istediğimizi düşünelim. Bu amaçla iki ayrı resim seçilmiştir. Birincisi, şekil 7'de gösterilen ve görüntü işleme alanında en çok tercih edilen 512x512 piksel boyutunda "Lena.bmp" görüntüsüdür. İkincisi ise Şekil 9'da gösterilen 768x512 piksel boyutunda "peppers.bmp" görüntüsüdür.

Kullanılan Shuffle Algoritması sonuçları ile geleneksel steganografi yöntemlerinde biri olan LSB yönteminin sonuçlarını karşılaştırmak için iki tane görüntü ve metin seçilmiştir. Birincisi, şekil 6'da gösterilen, görüntü işleme alanında en çok tercih edilen 512x512 piksel boyutunda "Lena.bmp" görüntüsüdür. İkincisi, şekil 7'de gösterilen 768x512 piksel boyutunda "peppers.bmp" görüntüsüdür.

Şekil 6. Gizlenecek metin

Şekil 7. Orijinal Lena resmi



Şekil 8. Shuffle algoritması ile metin gizlenmiş Lena resmi

Şekil 9. Orijinal peppers resmi

Şekil 10. Shuffle algoritması ile metin gizlenmiş Peppers resmi

Görüldüğü gibi, orijinal ve metin gizlenmiş resimler arasında gözle ayırt edilebilecek bir fark görülememektedir.

6. SONUÇ

Resmin içine yazı gizlenmesi amacıyla kullanılan LSB ekleme yönteminin shuffle algoritması ile iyileştirilmesi yoluyla kullanımı kolay ve etkin bir yöntem elde edilmiştir.

Ayrıca bu yöntemde:

Resim piksel pozisyonlarında çakışma mümkün olmadığından veri kaybı yoktur.

Resim üzerinde veri gizlendiğine dair gözle görülür bir kanıt bulunmamaktadır. Bundan dolayı, herhangi bir resim için anlamsız karakterler üretilecektir.

Resmin içindeki metnin çözülmesi ancak anahtarın bilinmesi ile mümkündür. Anahtarın deneme ile bulunması oldukça güçtür.

Eldeki resme kaç karakterlik bir metin sığdırılacağını hesaplamak mümkündür.

KAYNAKLAR

[1] Chandramouli, R., Memon, N., 2001. Analysis of LSB Based Image Steganography Techniques, *Proceedings of the International Conference on Image Processing*, Thessalonica, Yunanistan, 1019-1022.

[2] Sharp, T.,2001 An Implementation of Key-Based Digital Signal Steganography, *Information Hiding: 4th International Workshop*, l 25-27.

[3] Marvel, L. and Retter, C., 1999. Spread Spectrum Image Steganography, *IEEE Transactions on Image Processing*, 8(8), 1075-1083.

[4] Amin, M.M. and Salleh, M. and Ibrahim, S. and Katmin, 2003. Information hiding using steganography*, 4th National Conference on Telecommunication Technology Proceedings, Shah Alam, Malaysia,* Page(s):21 – 25.

Bu çalışmanın amacı, geleneksel veri gizleme yöntemlerine alternatif olacak daha güvenlikli bir veri gizleme ortamı yaratmaktır. Shuffle algoritması kullanılarak veri gizleme yönteminin daha yüksek bir güvenlik sağlamasından dolayı, bu ve benzer çalışmalarda daha fazla ilgi göreceğine