BİLGİSAYAR BİLİMLERİ VE TEKNOLOJİLERİ DERGİSİ
JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGIES
http://dergipark.org.tr/tr/pub/bibted

Araştırma Makalesi

Kriptografi ve Görüntü Steganografi Tabanlı Bir Veri Gizleme Uygulaması: Sten 0.1

Serhat ÇELİK[1], Nesibe YALÇIN*[1]

[1]Erciyes Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, Kayseri, Türkiye

Anahtar Kelimeler:
Bilgi güvenliği
Kriptografi
Görüntü steganografi
LSB
Veri gizleme

ÖZ

Gelişen iletişim ve bilişim teknolojileri ile dijital iletişim daha hızlı ve kolay olmuş, iletişim ortamlarının kullanımı yaygınlaşmıştır. Diğer taraftan iletilen bilgiye yönelik saldırılar da sayı ve çeşit olarak artış göstermiştir. İletişim kurmak isteyen iki taraf arasında, özel ve güvenli bir iletişim ortamının sağlanmasına ihtiyaç duyulmaktadır. Bilgi güvenliği temel hedeflerinden olan gizlilik, kriptografi veya steganografi yöntemleri kullanılarak güvence altına alınabilir. Ayrıca, daha gelişmiş bir güvenlik için bu yöntemler bileştirilerek kullanılabilir. Çalışmada, veri gizliliğini sağlamaya yönelik kriptografi ve steganografi yöntemlerinin birlikte kullanıldığı bir yaklaşım önerilmiştir. Önerilen yaklaşım, görüntü dosyası içerisine veri gömme/çıkarma işlemlerini bir özet (hash) string ile gerçekleştirerek kullanıcıya ek bir güvenlik katmanı sunmaktadır. Bu işlem az bir gecikmeye neden olsa da maksimum kapasitede veri gizleme yapıldığında ulaşılan sonuçlar memnuniyet vericidir. Her bir görüntü kanalının en az anlamlı 4 bitine rastgele veri gizlenmiş görüntüler, 30 dB ve üzeri kabul edilebilir bir görüntü kalitesine sahiptir. Ayrıca, en yüksek benzerlik indeksi 0,991 olarak elde edilmiştir. Çalışma kapsamında ayrıca bir uygulama geliştirilmiştir. Geliştirilen uygulamada kriptografi yöntemleri ve en az anlamlı bit tabanlı görüntü steganografi ayrı ayrı veya birlikte çeşitli amaçlar için kullanılabilmektedir.

A Data Hiding Application based Cryptography and Image Steganography Methods: Sten 0.1

Keywords:
Information security
Cryptography
Image steganography
LSB
Data hiding

ABSTRACT

Digital communication has become easier and faster with the developing communication and information technologies, the use of communication environments has become widespread. On the other hand, attacks on transmitted information have increased in number and variety. There is a need to provide a private and secure communication environment between the two people who want to communicate securely. Confidentiality, one of information security targets, can be secured by using cryptography or steganography methods. In addition, these methods can be combined for enhanced security. This study proposes an approach that combines both methods to ensure data privacy. It provides enhanced security to the user by embedding/extracting the data in the image file with a hash. Although this process has resulted in a slight delay, the satisfaction results are obtained when random data is hiding at maximum capacity. The images with data hidden in the 4-least significant bit (LSB) of each color channel have an acceptable image quality of almost 30 dB and above. In addition, the highest similarity index is obtained as 0.991. The study also includes the development of an application. In the developed application, cryptography methods and the LSB-based image steganography can be used separately or together for various purposes.

*Sorumlu Yazar
*(nesibeyalcin@erciyes.edu.tr) ORCID ID 0000-0003-0324-9111
(serhatcelik@erciyes.edu.tr) ORCID ID 0000-0002-4717-1507
e-ISSN: 2717-8579

Geliş Tarihi: 06/06/2023l; Kabul Tarihi: 22/09/2023

1. GİRİŞ

Günümüzde bilişim teknolojilerinin hızlı gelişmesi ve yaygınlaşması ile internet kullanımı artmakta, bununla birlikte internet üzerinden gönderilen ve alınan verilerin güvenliği ciddi bir sorun olmaya devam etmektedir. Bilgi güvenliği; bilgilerin, verilerin, kaynakların, varlıkların ve değeri olan her şeyin yetkisiz erişim ve işlemlere (değiştirilme, silinme, yok edilme, izinsiz kullanma, ifşa edilme gibi) karşı korunması olarak tanımlanabilir ve üç temel başlık altında incelenebilir:

- Gizlilik; sadece yetkili kişilerin verilere erişimine izin verilmesi
- Bütünlük; verilerin yetkisiz değiştirilme veya bozulmalara karşı korunması
- Erişilebilirlik; yetkili kişilerin verilere ihtiyaç duyduklarında sürekli erişiminin sağlanması

Gizlilik, bütünlük ve erişilebilirlik ihtiyaçları, çeşitli uygulamalarda farklı şekilde vurgulanabilir. Örneğin, bir bankacılık uygulamasında kullanıcı parolasının gizli tutulması önemli iken, bireysel işlemlerin bütünlüğü korunmalı ve işlemlerin gerçekleştirilebilmesi için bankacılık sistemi erişilebilir olmalıdır. Bilgi güvenliği ihtiyaçlarını karşılamak için erişim denetimi, kimlik doğrulama, veri sınıflandırma, şifreleme, risk yönetimi, değişim kontrolü, yedekleme ve olay müdahale gibi araçlardan yararlanılmaktadır. Kriptografi ve steganografi, verileri yetkisiz kişilerden korumada en yaygın kullanılan yöntemlerdendir. Kriptografi, bir metnin içeriğini diğer bir kişinin anlamayacağı hale getirerek (şifreleyerek), steganografi ise bir veriyi başka bir verinin (görüntü, metin, ses gibi kapak/örtü dosya) içerisine saklayarak veriyi korur. Gizliliği sağlamanın en etkili yollarından biri, bu iki yöntemin birleştirilerek kullanılmasıdır.

Bilgi güvenliğini sağlamak, verileri güvence altına almak için kriptografi ve steganografiyi birleştirerek çift katmanlı bir güvenlik sunan birçok çalışma yapılmıştır. Hammad vd. (2022), steganografi ve kriptografi birleştiren bir yaklaşım benimsemiştir. Çalışmada metin önce Vigenère daha sonra Sezar şifreleme ile şifrelenmiştir. Daha sonra periyodik tabloya dönüştürülerek (her bir karakterin ASCII kodu, periyodik tablodaki bir atom numarası olacak şekilde kimyasal elementin sembolü ile değiştirilerek) şifreli metin elde edilmiştir. Elde edilen şifreli metin, örtü görüntü içerisine gömülerek gerçek bilginin varlığı gizlenmiştir. Awadh vd. (2022) tarafından veri güvenliğini sağlamak ve internet üzerinden veri alışverişi kapasitesi sorununu çözmek için hibrit bir güvenlik sistemi önerilmiştir. Önerilen sistem, bir görüntüyü başka bir görüntü içerisinde gizleme sürecini içermektedir. İlk olarak ayrık dalgacık dönüşümü kullanılarak gizlenecek görüntü sıkıştırılmış, daha sonra sıkıştırılmış görüntü Gelişmiş Şifreleme Standardı (Advanced Encryption Standard, AES) ile şifrelenmiştir. Elde edilen şifreli bitler, en az anlamlı bit (Least Significant Bit, LSB) ekleme algoritması kullanılarak başka bir görüntü içerisine yerleştirilmiş ve internet üzerinden aktarıma hazır hale getirilmiştir. AES algoritmasının kullanıldığı bir başka çalışmada (Singh ve Atria, 2015), ekstra güvenlik için gizli mesajın saklandığı görüntü dosyası şifrelenmiştir. Eliptik Eğri Kriptografisi ile görüntü steganografi birleştirilerek yapılan çalışmada (Hureib ve Gutub, 2020), gizli ve özel verilerin daha güvenli bir şekilde korunması sağlanmıştır. Adee ve Mouratidis (2022), veri güvenliğini artırmak ve bulut bilişim ortamlarında gizliliği korumak amacıyla dört aşamalı bir güvenlik modeli önermişlerdir. AES ve Rivest Shamir Adleman algoritmaları ile şifreleme ve LSB steganografi yöntemini uygulama modelin ilk iki aşamasını oluşturmaktadır. Bu aşamalardan sonra, şifreleme ve şifre çözme işlemlerinin sonuçlarına ilişkin veri yedekleme/kurtarma ve güvenli veri paylaşımı gelmektedir. Koçak tarafından yapılan çalışmada (2015), simetrik şifreleme sonucu elde edilen metin toplamda en az anlamlı 4 bit kullanılarak görüntü içerisine gizlenmiştir. Orijinal görüntü ile metnin gizlendiği görüntü benzerlik ve bozulma açısından incelenmiş, yapılan işlem sonucu görüntü kalitesinde fazla bir bozulma olmadığı belirtilmiştir. Özbilgin vd. (2018) tarafından, Vigenère şifreleme ile farklı bir şifreleme gerçekleştirilmiş ve daha sonra görüntüdeki satır, sütun veya köşegenlerde yer alan piksellerin mavi (blue, B) renk kanallarına LSB yöntemi ile gizlenmiştir. Osman vd. (2022), tek kullanımlık şerit akış şifreleme ile görüntü steganografi yöntemine dayalı hibrit bir yaklaşım önermişlerdir. Çalışmada, çeşitli görüntü formatlarında örtü görüntüleri ve farklı metin boyutları test edilmiş ve gizlenecek metin boyutunun örtü görüntüden %15 daha küçük olması gerektiği ifade edilmiştir. Ansari vd. (2020), PNG görüntülerde kullanılabilen bir şifreleme ve kümelemeye dayalı bir algoritma önermiştir. Önerilen algoritma ile PNG görüntü kullanılarak 40 bin üzeri bitin internette güvenli bir şekilde aktarımının sağlanabildiği belirtilmiştir.

Steganografi konulu çalışmalarda, LSB ekleme algoritmasının yaygın olarak ele alındığı, farklı renk kanalları ile iki veya daha fazla LSB kullanıldığı da görülmüştür. Şahin vd. (2006), geliştirdikleri kullanıcı arayüzü aracılığı ile farklı boyutlardaki 24-bit renkli bitmap görüntü dosyaları üzerinde LSB yöntemini kullanarak metin gizleme ve çıkarma yapmışlardır. Koçak çalışmasında (2015), kırmızı (red, R) ve yeşil (green, G) renk kanalları ile her bir kanaldaki en az anlamlı 2 biti gizli metni gömmek için kullanmıştır. Doğan vd. (2016) ise görüntülerdeki 0. ve 2. bitler veri gömme için tercih etmişlerdir. Hureib ve Gutub tarafından yapılan çalışmada (2020), olumlu ve olumsuz yönlerini ortaya koyabilmek amacıyla bir ve iki LSB ekleme test edilmiştir. Mesajın bir pikselin R, G ve B

kanallarının rastgele bit konumlarına gömülmesi yaklaşımının önerildiği bir çalışmada (Ali vd., 2019) standart LSB ekleme yöntemi ile aynı PSNR değerleri elde edilmiştir. Bununla birlikte mesaj pikselin bitlerine ardışık yerleştirilmediği için ek bir güvenlik sunmaktadır. LSB ekleme yöntemine bit kaydırma işleminin eklendiği çalışmada (Solak ve Altınışık, 2019), anahtar ile metin önce şifrelenmiş daha sonra şifrelenmiş metinde kaydırma yapılarak gizlenecek metin elde edilmiştir. 8 bitlik bir veri için RGBBGRRG kanalları kullanılmış ve gizlenecek metin sırasıyla bu kanalların en az anlamlı bitlerine yerleştirilmiştir. Yakut (2022a) tarafından LSB yöntemine dayalı önerilen yaklaşımda, gizlenecek mesaj taşıyıcı verinin en az anlamlı bitlerinde herhangi bir değişiklik olmaksızın iletilmektedir. İletilen bitler ve taşıyıcının bitleri XOR işlemine tabi tutulur ve sonuç verisi üretilir. Bu şekilde gizlenecek mesajın tespiti mümkün olmamaktadır, ancak sonuç verisinin gönderimi alıcıya ek bir yük getirmektedir.

Balkesen ve Koçer tarafından yapılan çalışmada (2020), AES ile şifrelenmiş metni görüntülere gizlemek için rastgele piksel seçim yaklaşımı benimsenmiştir. Söz konusu yaklaşımda seçilen piksel bilgisi, görüntüde gizlenmekte ve böylece gizlenen metnin doğru çıkarılması sağlanmaktadır. Rastgele piksellere bit yerleştirmenin yapıldığı bir diğer çalışmada (Emam vd., 2016), bir karakterin yerleştirilmesi 1. piksel BG kanalları ve 2-LSB, 2. piksel B kanalı ve 1-LSB, 3. piksel BG kanalları ve 2-LSB ekleme şeklinde gerçekleştirilmiştir. Bhardwaj ve Sharma (2016) tarafından rastgele seçilmiş piksellere ters bit LSB yöntemi kullanılarak 8 bitlik gri tonlamalı bir görüntü seti üzerinde veri gizleme deneyleri yürütülmüştür. Önerilen yöntem ile veri gizlenmiş görüntülerin, standart LSB yöntemi ile elde edilenlerden daha yüksek görsel kaliteye sahip olduğu belirtilmiştir. Bu çalışmaların en önemli kısımlarından biri rastgele sayı üretimidir. Yakut (2021, 2022b) çalışmalarında bu konu üzerine odaklanmış ve ayrık kosinüs dönüşümünü kullanan bir rastgele sayı üreteci önermiştir. Gri tonlamalı görüntüler içerisine gizlenmiş mesajın tespitini zorlaştırmak için önerilen bir yöntemde (Macit ve Koyun, 2020) ise örtü görüntü ilk olarak bloklara ayrılmış ve her bir blok için çeşitli hesaplamalar yapılara görüntüye ilişkin daha az ayrıntı içeren bloklar tespit edilmiş ve gizli mesaj yerleştirilmiştir. Böylece görüntünün tamamı yerine bazı bloklarına veri gizlendiğinden standart LSB çıkarma yöntemi, gömülü metnin eldesi için kullanılamayacaktır. Çalışma sonucunda blok sayısını artırmanın, veri kapasitesini artırdığı diğer taraftan karmaşıklığı ve işlem süresini de artırdığı ifade edilmiştir. Baysan ve Özekes (2023) tarafından veri gizleme için uzaklaştırılmış LSB ekleme yöntemi önerilmiştir. Önerilen yöntem ile gizlenecek mesaj, görüntü içerisine orantılı olarak dağıtılarak belirli bir alanda yığılma engellenmiştir. Konyar vd. (2018) tarafından ise kullanıcıların veri gizleme için en uygun görüntüyü seçebilmelerine olanak tanıyan bir arayüz tasarlanmıştır. Gizleme kapasitesi için RGB kanallarının hepsi kullanılmış ve gizleme sonrası en az değişikliğe uğrayan görüntünün kullanıcıya önerilmesi sağlanmıştır.

Bu çalışmada, bazı klasik kriptografi yöntemleri ve LSB görüntü steganografi algoritması kullanılarak ve özet string aracılığı ile verileri gizleyerek üç katmanlı bir güvenlik uygulaması "Sten 0.1" geliştirilmiştir. Uygulama ile çeşitli formattaki görüntü dosyaları üzerinde farklı renk kanalları ve LSB sayısı seçimi ile veri (orijinal ya da şifrelenmiş metin) gizleme işlemi mümkündür. Çalışmada kullanılan kriptografi ve steganografi yöntemleri Bölüm 2'de açıklanmıştır. Geliştirilen uygulamaya ilişkin detaylar Bölüm 3'te, uygulama sonuçları ise Bölüm 4'te verilmiştir. Son bölümde, sonuçlar değerlendirilmiş ve öneriler sunulmuştur.

2. BİLGİ GÜVENLİĞİ SİSTEMLERİ

Bilişim teknolojisi alanındaki gelişmelerle birlikte, verilerin güvenliğini sağlamak için birçok yöntem uygulanmaktadır. Kriptografi ve steganografi, veri gizliliğini sağlamada kullanılabilecek yöntemlerdir. Bu bölümde, klasik kriptografi yöntemleri ve görüntü steganografi yöntemi olan LSB ekleme sunulmuştur.

2.1. Kriptografi

Kriptografi, bir mesajı yetkili olmayan kişiler için anlaşılması zor bir forma dönüştürerek koruma altına almayı amaçlar. Mesajın yalnızca alıcı ve gönderici tarafından okunmasına, görüntülenmesine izin verir ve üçüncü şahıslara, kötü niyetli taraflara karşı haberleşmenin güvenliğini sağlar. Geliştirilmiş birçok kriptografi algoritması vardır ve anahtar kullanımı, anahtar yapısı/türü, karmaşıklık ve performans açısından farklılık göstermektedir. Çalışma kapsamında yerine koyma, yer değiştirme, tek alfabeli, çok alfabeli, blok şifreleme gibi farklı özellikler sunan klasik kriptografi yöntemleri tercih edilmiştir. Bu yöntemlerin ortak özelliği ise şifreleme ve şifre çözme için sadece bir gizli anahtar kullanmasıdır.

- Sezar şifreleme; tek alfabeli bir yerine koyma şifreleme yöntemidir. Romalı lider Julius Caesar tarafından ordu haberleşmesinde mesajların güvenliğini sağlamak için kullanılmıştır (Abraham ve Shefiu, 2012). Şifrelenecek metindeki her bir harf yerine, o harften anahtar olarak belirlenen miktar kadar ileri kaydırma (öteleme) yapıldığında karşılık gelen harf kullanılarak şifreleme yapılır. Şifre çözme (deşifre etme) işlemi için ise yapılan işlemin tersi uygulanır.
- Scytale (sarmal) şifreleme; en bilinen yer değiştirme şifrelemesidir. Scytale olarak adlandırılan bir tahta çubuğun etrafına sarılmış parşömen veya papirüs şeridine gizlemek istenilen mesaj yazılır. Şerit açıldığında orijinal mesajdaki tüm harfler farklı bir konuma aktarılmış ve böylece şifrelenmiş metin elde edilmiş olur (Diepenbroek,

2021). Mesajın deşifre edilebilmesi için aynı yarıçapa sahip Scytale cihazının kullanılması gerekir.

- Vigenère şifreleme; çoklu alfabe kullanan bir yerine koyma şifreleme türüdür. Alfabedeki harfler satır ve sütun isimlerini ifade etmek üzere bir tablo oluşturulur (Aliyu ve Olaniyan, 2016). Tablonun ilk satırına orijinal alfabe yazılır, daha sonra satırdaki harfler birer kaydırılarak bir sonraki satıra yazılacak alfabe elde edilir ve bu işlem tablonun bütün satırları için tekrarlanır. Şifreleme işleminde oluşturulan bu tabloda satır, şifrelenecek metnin harflerini; sütun ise anahtar metnin/kelimenin harflerini temsil eder. Seçilen anahtar metnin boyutu, şifrelenecek metinden daha kısa uzunlukta ise anahtar metin, şifrelenecek metni tamamen örtene kadar yinelenir. Sırasıyla şifrelenecek metnin her bir harfi ile anahtar metnin aynı sıradaki (indeksteki) harfine tabloda karşılık gelen harf yazılarak şifreli metin oluşturulur.
- Hill şifreleme; bir blok şifreleme türüdür ve lineer cebire dayanır. Şifrelenecek metin belirli bir büyüklükteki bloklara ayrılarak bloklar halinde şifrelenir. Blok boyutuna (k) göre şifrelemede kullanılacak anahtar kare matrisin boyutu (k × k) belirlenir. Şifreleme işleminde kullanılan anahtar matrisin elemanlarını (ASCII kod için 0-255 aralığında) belirlemek önemli bir sorundur. Şifre çözme işlemi de düşünüldüğünde matrisin bir tersi olmalıdır ve anahtarın determinantı 1 olacak şekilde elemanlar seçilmelidir (Putera vd., 2016). Şifrelenecek metindeki harflerin alfabedeki sırasından ya da ASCII değerlerinden oluşan her bir blok, anahtar matris ile çarpılır ve daha sonra kullanılan alfabedeki harf sayısına ya da 256'ya göre modu alınır ve elde edilen sayısal değerlere karşılık gelen harfler ile şifreli metin elde edilir.

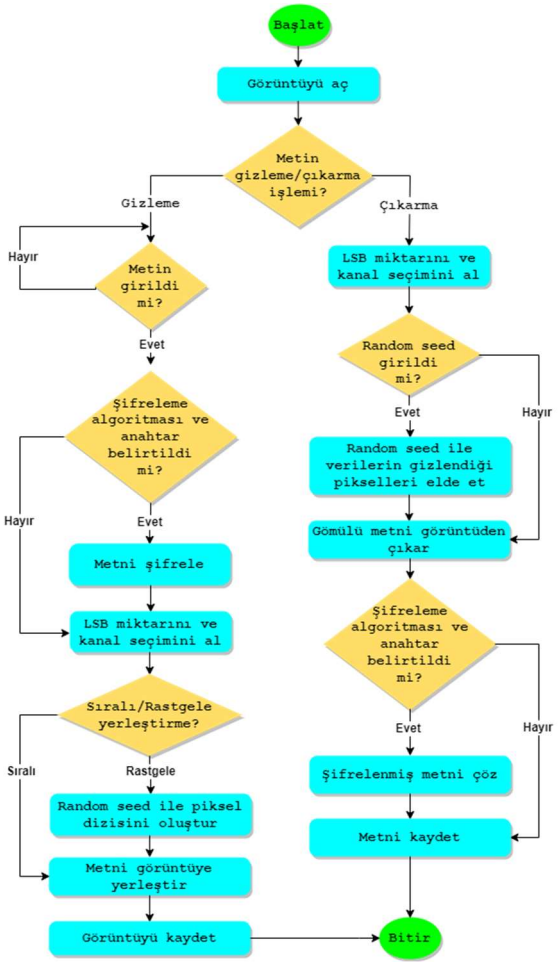
2.2. Steganografi

Steganografi, gizlilik, mahremiyet, veri bütünlüğü gibi bilgi güvenliği kavramlarını sağlamak için yaygın olarak kullanılır. "Gizlenmiş yazı" anlamına gelir ve veriyi yetkisiz kişilerden gizleme amacını taşır. Veriyi gizlemek için görüntü, metin, video gibi çeşitli ortamlar kullanılır, bu ortama örtü (cover, kapak) ve içerisinde gizlenmiş veri bulunan ortama ise stego denir. Elde edilen tüm stego ortamlar için kalite, çok önemli bir parametredir. Objektif kalite değerlendirmesi yapmak temel doğruluk için gereklidir.

Bu çalışmada, LSB tabanlı görüntü steganografi yöntemi kullanılmıştır. LSB ekleme, sıklıkla uygulanan bir steganografi yöntemidir ve daha az karmaşıktır. Gizlenecek metin, ikili sayı (binary) sisteminde ifade edilir ve daha sonra görüntünün her bir pikselinin en az anlamlı bitlerine sırasıyla yerleştirilir. Yerleştirmede LSB kullanımı, orijinal görüntü üzerinde gözle görülür bir değişiklik oluşturmamaktadır. RGB görüntülerde her bir renk kanalının en az anlamlı bitleri kullanılarak da veri gizlenebilir. Ayrıca farklı kanal(lar) ve/veya LSB miktarı tercih edilerek güvenlik artırılabilir. En uygun gizleme yöntemini belirlemek ve örtü görüntü ile stego görüntü arasındaki farkı değerlendirmek için Tepe Sinyal Gürültü Oranı (Peak Signal-to-Noise Ratio, PSNR) ve Ortalama Karesel Hata (Mean Squared Error, MSE) kalite ölçümleri yaygın olarak kullanılmaktadır (Sara vd., 2019).

3. UYGULAMA

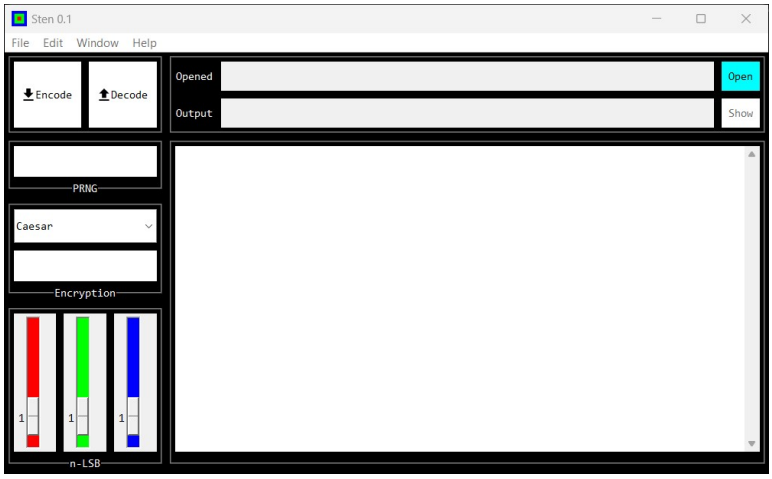
Çalışmada, çeşitli klasik kriptografi yöntemlerini ve LSB görüntü steganografiyi birlikte kullanan Sten 0.1 uygulaması geliştirilmiştir. Uygulama aracılığıyla metin şifreleme ve şifre çözme, bir metnin orijinal veya şifrelenmiş halini bir örtü görüntüye gizleme ve çıkarma yapılabilmektedir. Süreci özetleyen akış diyagramı Şekil 1'de verilmiştir.



Şekil 1. Sten 0.1 uygulaması işlem akışı

3.1. Sten 0.1

Sten 0.1 uygulaması, Python dilinde Tkinter Grafiksel Kullanıcı Arayüzü (Graphical User Interface, GUI) kütüphanesi yardımıyla geliştirilmiştir. Uygulama ile yapılabilecek şifreleme, şifre çözme, veri yerleştirme/gömme ve çıkarma gibi işlemleri son kullanıcıya en etkili biçimde sunabilmek amacıyla Şekil 2'de verilen menüye dayalı GUI tasarlanmıştır.



Şekil 2. Sten 0.1 kullanıcı arayüzü

3.2. Şifreleme ve şifre çözme

Arayüz ile kriptografi için tercih edilen şifreleme yöntemi (Sezar, Scytale, Vigenère ve Hill şifreleme) seçilebilmekte ve seçilen yönteme ait bir gizli anahtar girişi yapılabilmektedir (bkz. Şekil 3). Uygun anahtar seçiminin yapılabilmesi için çeşitli kontroller gerçekleştirilmektedir. Örneğin, Sezar şifrelemede sadece sayı girişine izin verilmektedir. Bu durumun dışında kalan her giriş reddedilmektedir. Aynı şekilde, "Decode" işleminin yapılabilmesi için de bu bilgilerin girilmesi gerekmektedir. Şekil 3'te Sezar şifrelemeye ait girilen şifreleme anahtarı *** olarak gözükmektedir. Bu, ekstra bir güvenlik gerekçesiyle yapılmıştır. Kullanıcı isterse uygulamaya ait konfigürasyon (yapılandırma) dosyasında değişiklik yaparak bu davranışı değiştirebilir. Bu şekilde girilen anahtar değeri varsayılan olarak görünür olacaktır.

Şekil 3. Şifreleme algoritması seçimi ve anahtar girişi

Şifreleme işlemlerinin gerçekleştirilmesinde ele alınan hususlar aşağıda listelenmiştir.

- Sezar şifreleme için girilen anahtar değerinin (öteleme miktarı) sayı olup olmadığı kontrol edilir ve alfabe uzunluğunun katlarına eşit olmadığından emin olunur.
- Scytale şifreleme ve şifre çözme için aynı özellikte Scytale cihazı kullanılmalıdır. Bu nedenle cihazın yarıçap değeri, uygulamada anahtar olarak kullanılmıştır. Şifrelemede metin matris (satır × sütun) formatında ele alındığında anahtar değeri, sütun sayısına (sonraki satıra geçme adımına) karşılık gelir. Şifrelenecek metnin harfleri, satır ve sütunlara sırasıyla yerleştirilir ve sonra tek boyutlu diziye dönüştürülerek şifrelenmiş metin elde edilir. Anahtar değerinin sıfırdan farklı bir sayı olarak girilmesi sağlanır.
- Vigenère ve Hill şifrelemede, girilen anahtar metnin karakterlerinin alfabede olduğundan emin olunur.
- Hill şifreleme işleminde seçilen bir metin, önceden belirlenmiş bir kare matrise metnin her harfinin ASCII tablosundaki sayısal karşılığı yerleştirilerek anahtar oluşturulur. Matriste eksik kalan kısımlar rastgele sayılarla tamamlanır. Ancak her metin, anahtar olarak kullanılamaz. Metnin anahtar olarak kullanılabilmesi için anahtar ile oluşturulacak matrisin tersinin alınabiliyor olması ve aynı zamanda bu matrisin determinantının, kullanılacak alfabedeki karakter sayısıyla aralarında asal olması gerekmektedir. Bunun sebebi, şifre çözme işleminde matrisin ters çevrilecek olmasıdır. Eğer matrisin determinantı alfabe uzunluğu ile aralarında asal olmazsa ortak bölenleri 1'den farklı gelir. Bu, anahtar olarak kullanılamaz. Bu şartları sağlayan bir anahtar seçildikten sonra, şifrelenecek metin anahtar matrisin bir boyutunun uzunluğunda bloklara ayrılır. Bloklar ile anahtar matris çarpılır, elde edilen sayıların ASCII karşılıkları bulunur ve şifrelenmiş metnin blokları elde edilir. Şifre çözme işlemi ise anahtar matrisin tersi ile şifrelenmiş metnin her bir bloğunun matris çarpımına sokulması ile gerçekleştirilir.

3.3. Veri gizleme ve çıkarma

3.3.1. Örtü görüntüsü seçimi

Örtü görüntü, kullanıcıdan alınan ve içerisine gizli mesajın yerleştirilmesi amacıyla kullanılan görüntü dosyasıdır. LSB yöntemi, yapısı gereği sıkıştırmasız görüntü formatlarıyla uğraşmayı gerektirir. Bu yüzden çalışmada JPEG tarzı sıkıştırılmış görüntü formatları yerine RGB veya RGBA renk modunda BMP ve PNG görüntülerle çalışılmıştır. Uygulamada Şekil 4'te gösterilen 24 bit derinliğine sahip dört farklı test örtü görüntüsü üzerinde deneyler gerçekleştirilmiştir. "Lena" ve "Pepper" standart görüntü dosyaları 512×512, "Cat" görüntüsü (URL-1) 900×900 ve "Scenery" (URL-2) ise 1280×853 piksel boyutundadır.

Seçilen örtü görüntüye ait faydalı bilgiler "File→Image Properties" seçeneği ile (boyut, renk modu, veri kapasitesi gibi) görüntülenebilmektedir. Veri gizleme öncesi görüntünün veri yükleme kapasitesini görmesi açısından kullanıcıya avantaj sağlamaktadır.

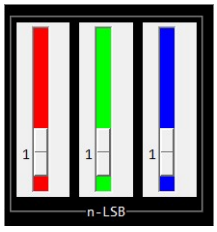
a)  b) 

c)  d) 

Şekil 4. Test görüntüleri, a) Lena, b) Pepper, c) Cat ve d) Scenery

3.3.2. Verinin örtü görüntüye sıralı yerleştirilmesi

Gizlenecek metnin ve örtü görüntünün boyutuna bağlı olarak kullanıcı, arayüz ile kanal(lar) ve LSB miktarı tercihi yapılabilmektedir. Bu tercih, gizli verinin örtü verisine yerleştirilmesi ve stego verisinden gizli verinin çıkarılması aşamasında bir anahtar görevi görecektir. Verilerin gizlenmesinde önceki çalışmalarda kullanılan, bir, iki veya ikiden fazla LSB kullanımına alternatif olarak kullanıcıya, renkli bir görüntünün her bir renk kanalı için ayrı ayrı LSB miktarını belirleme seçeneği sunulmuştur. RGB renk modunda bir görüntü için minimum 1 ve maksimum 24 bit kullanılabilir. Kullanıcı arayüzü üzerinde her bir kanal için LSB miktarının belirlenebilmesini sağlayan kısım Şekil 5'te gösterilmiştir.



Şekil 5. Test görüntüleri, a) Lena, b) Pepper, c) Cat ve d) Scenery

Her bir piksel için n-LSB kullanımı durumunda, görüntü içerisine yerleştirilebilecek maksimum bit sayısı (kapasite), Denklem (1) yardımıyla hesaplanmaktadır. 1-LSB kullanımı durumunda elde edilen değerler, Lena görüntüsü için 786.432, Cat görüntüsü için 2.430.000 ve Scenery görüntüsü için 3.275.520 bittir.

kapasite = n × piksel genişlik
× piksel yükseklik
× renk kanalı sayısı (1)

Gizlenmek istenen veri, gizleme seçenekleri ile birlikte "gizleme fonksiyonuna" verilir. Bu fonksiyon gizli veriyi örtü görüntüye orijinal veya şifrelenmiş hali ile yerleştirerek stego görüntünün elde edilmesini sağlar. Ardından stego görüntünün kaydedileceği konum bilgisi alınır ve işlemin başarılı olması durumunda bir bilgilendirme mesajı kullanıcıya gösterilir.

3.3.3. Verinin örtü görüntüye rastgele yerleştirilmesi

Ek opsiyonel bir güvenlik adımı olarak gizlenecek veri bitleri doğrudan örtü görüntüye gömülmek yerine rastgele seçilmiş piksellerine yerleştirilebilmektedir. Bunun için bir Sözde Rastgele Sayı Üreteci (Pseudo-Random Number Generator, PRNG) seed (tohum) değeri, anahtar görevi üstlenecek şekilde kullanıcıya sunulmuştur (bkz. Şekil 2). PRNG, bir seed değeri alan ve bir dizi benzersiz sayı üreten bir kara kutu görevi görmektedir (Emam vd., 2016). Uygulama kapsamında, kullanıcıdan bir string (dizge) değer alınır. Girilen değer için PRNG yardımı ile sabit uzunlukta bir özet string elde edilir. Anahtar değer girilmeden de renk kanalı ve LSB miktarı seçimi girdi olarak da alınabilmektedir. Böylece ek bir anahtar değere ihtiyaç ortadan kalkmaktadır. Son olarak özet kullanılarak görüntüdeki piksellerin indislerinin rastgele dağılmış bir versiyonu elde edilir. Böylece kullanılan veri bitleri de saklanarak gömülü metnin standart LSB çıkarma yöntemi ile tespit edilme şansı azaltılmaktadır. Eğer bu işlem tercih edilmek istenmez ise veriler görüntü içerisine sıralı olarak yerleştirilecektir. İlgili işleme ait sözde kod aşağıda verilmiştir.

Girişler : Gizlenecek mesaj, *M*
Şifreleme algoritması, *C*
Şifreleme anahtarı, *K*
PRNG, *R*
Resim dosyası pikselleri, *P*
R, G ve B için LSB, $L = \{LSB_R, LSB_G, LSB_B\}$
Çıkış : Stego görüntü, *S*

```
if M is None then
    return None
else
    if C is not None then
        if K is None then
            return None
        else
            sifrelemeAlgoritmasiKullan = True
        end
    else
        sifrelemeAlgoritmasiKullan = False
    end
end
if R is not None then
    P piksellerini listeye dönüştür
    P piksellerini R değerine göre rastgele karıştır
else
    P piksellerini listeye dönüştür
end
if sifrelemeAlgoritmasiKullan is True then
    M mesajını C şifreleme algoritması ve K şifreleme anahtarı ile şifrele
    L değeri ile M mesajını P piksellerine yerleştirerek S stego nesnesini oluştur
else
    L değeri ile M mesajını P piksellerine yerleştirerek S stego nesnesini oluştur
end
return S
```

3.3.4. Gizli verinin stego görüntüden çıkarılması

Stego görüntü, arayüz aracılığıyla seçildikten sonra şifreleme ve veri gizleme işlemlerinde kullanılan şifreleme yöntemi ve anahtarı, renk kanalı ve LSB miktarı seçimi ile PRNG seed uygulanma durumu için aynı bilgiler girilir. Bu bilgiler, "uygulama tarafından gizlenmiş veriyi tekrar elde etme fonksiyonuna" gönderilir. Bu şekilde fonksiyon önceden yapılan işlemleri tersten yürüterek orijinal metnin elde edilmesini sağlar.

4. SONUÇLAR VE TARTIŞMA

Uygulama kapsamında Şekil 4'te verilen örtü görüntülere çeşitli alternatiflerle aynı boyutta veri gizlenmiş işlem süreleri incelenmiştir. Sıralı ve rastgele seçilmiş görüntü piksellerine farklı LSB miktarında yerleştirme yapılarak elde edilmiş stego görüntüler, örtü görüntülerle MSE, PSNR ve Yapısal Benzerlik İndeksi Ölçümü (Structural Similarity Index Measurement, SSIM) bakımından karşılaştırılmış ve görüntülerdeki bozulma oranları analiz edilmiştir.

MSE, orijinal görüntü (O) ile stego görüntü (O') arasındaki hatanın karesel ortalamasını verir ve Denklem (2) yardımı ile hesaplanır (Balkesen ve Koçer, 2020). Görüntülerdeki pikseller, h×w boyutunda bir matris olarak ele alınır.

$$MSE = \frac{\sum_{i=1}^{h}\sum_{j=1}^{w}\left(O_{(i,j)} - O'_{(i,j)}\right)^2}{h \times w} \quad (2)$$

MSE, Denklem (3) ile PSNR'ye dönüştürülür. Orijinal görüntü ile stego görüntü arasındaki farkı değerlendirmek için bir kalite ölçümü olarak kullanılır. Desibel (dB) cinsinden ölçülür (Ali ve ark., 2019; Ansari ve ark., 2020). Daha yüksek bir PSNR, PSNR'nin daha iyi olduğunu gösterir (Awadh ve ark., 2022). X, orijinal görüntünün tepe sinyal değeridir ve Denklem (4) kullanılarak hesaplanır (Bhardwaj ve Sharma, 2016).

$$PSNR = 10 \times \log_{10}\frac{X^2}{MSE} \quad (3)$$

$$X = max\left(O_{(i,j)} - O'_{(i,j)}\right) \quad (4)$$

SSIM, iki görüntünün ne kadar benzer olduğunu belirlemek için kullanılır ve değeri ne kadar yüksekse o kadar iyidir. Denklem (5) yardımı ile hesaplanır. c_1 ve c_2, her terimi stabilize eden küçük pozitif sabitlerdir. μ_O ve σ_O, orijinal görüntüye ilişkin piksel ortalamasını ve standart sapmasını temsil etmektedir. $\mu_{O'}$ ve $\sigma_{O'}$ ise sırasıyla stego görüntüye ilişkin piksel ortalaması ve standart sapmasıdır (Awadh ve ark., 2022).

$$SSIM(O, O') = \frac{2 \times \mu_O \times \mu_{O'} + c_1}{\mu_O^2 + \mu_{O'}^2 + c_1} \times \frac{2 \times \sigma_{OO'} + c_2}{\sigma_O^2 + \sigma_{O'}^2 + c_2} \quad (5)$$

Piksellerin R, G ve B kanalları için ayrı ayrı 1-LSB, 3-LSB ve 4-LSB ekleme sıralı ve rastgele şekilde uygulanmış ve görüntü içerisine maksimum bit yerleştirilmesi yapılmıştır. 1-LSB, 3-LSB ve 4-LSB yöntemleri sonucu hesaplanan PSNR, MSE ve SSIM değerleri ve stego görüntünün oluşturulması için harcanan süreler, Tablo 1, Tablo 2 ve Tablo 3'te sırasıyla karşılaştırmalı olarak verilmiştir. Deneyler sonucu elde edilen görüntüler ise Tablo 4'te sunulmuştur.

Tablo 1. RGB kanalları için 1-LSB kullanıldığında elde edilen deneysel sonuçlar

Görüntü	Maksimum Kapasite (bit)	Gizlenen Karakter Boyutu (bayt)	PSNR (dB)		MSE		SSIM		Gizleme Süresi (sn)	
			Sıralı	Rastgele	Sıralı	Rastgele	Sıralı	Rastgele	Sıralı	Rastgele
Lena	786.432	98.304	51,141	51,143	0,319	0,303	0,997	0,997	1,321	1,341
Pepper	786.432	98.304	51,138	51,133	0,313	0,302	0,996	0,997	1,320	1,411
Cat	2.430.000	303.750	51,071	51,069	0,484	0,486	0,998	0,998	2,994	3,360
Scenery	3.275.520	409.440	51,143	51,146	0,317	0,306	0,997	0,997	4,767	5,356

Tablo 2. RGB kanalları için 3-LSB kullanıldığında elde edilen deneysel sonuçlar

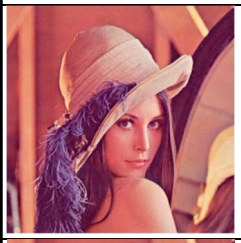
Görüntü	Maksimum Kapasite (bit)	Gizlenen Karakter Boyutu (bayt)	PSNR (dB)		MSE		SSIM		Gizleme Süresi (sn)	
			Sıralı	Rastgele	Sıralı	Rastgele	Sıralı	Rastgele	Sıralı	Rastgele
Lena	2.359.296	294.912	38,395	38,394	2,083	2,102	0,994	0,994	1,323	1,412
Pepper	2.359.296	294.912	38,285	38,285	2,395	2,426	0,990	0,990	1,320	1,411
Cat	7.290.000	911.250	38,833	38,838	9,133	9,031	0,995	0,995	3,001	3,655
Scenery	9.826.560	1.228.320	38,327	38,313	2,210	2,214	0,990	0,990	5,096	5,723

Tablo 3. RGB kanalları için 4-LSB kullanıldığında elde edilen deneysel sonuçlar

Görüntü	Maksimum Kapasite (bit)	Gizlenen Karakter Boyutu (bayt)	PSNR (dB)		MSE		SSIM		Gizleme Süresi (sn)	
			Sıralı	Rastgele	Sıralı	Rastgele	Sıralı	Rastgele	Sıralı	Rastgele
Lena	3.145.728	393.216	32,691	32,678	13,373	13,358	0,991	0,991	1,326	1,419
Pepper	3.145.728	393.216	32,861	32,851	8,955	9,088	0,981	0,981	1,320	1,411

Cat	9.720.000	1.215.000	29,217	29,216	67,967	67,973	0,992	0,992	3,317	3,742
Scenery	13.102.080	1.637.760	32,803	32,807	13,008	13,003	0,979	0,979	5,182	5,955

Tablo 4. Sıralı ve rastgele 1-LSB, 3-LSB ve 4-LSB yerleştirme sonucu elde edilen görüntüler

Yerleştirme	n-LSB	Stego Görüntü			
		Lena	Pepper	Cat	Scenery
Sıralı	1-LSB				
	3-LSB				
	4-LSB				
Rastgele	1-LSB				
	3-LSB				
	4-LSB				

Sıralı piksellere ve rastgele seçilmiş piksellere 1-LSB (bir pikselin her bir kanalına 1 bit olmak üzere toplamda 3 bit) ekleme yönteminin uygulanması elde edilen stego görüntülerin benzerlik indeksleri, %99,6 ve üzeridir. Bununla birlikte her iki durumda da birbirine yakın PSNR değerleri elde edilmiş ve 0,002 - 0,005 dB arasında bir değişim gözlenmiştir (bkz. Tablo 1). MSE değerleri açısından ise rastgele 1-LSB yerleştirme sonucu Cat görüntüsü hariç diğer görüntülerde azalma görülmüştür. Diğer taraftan 0,002 değerinde hata artışına sebep olarak, Cat görüntüsünün baskın bir arka planının olması gösterilebilir.

En yüksek boyuta sahip Scenery görüntüsüne, maksimum kapasitede, rastgele piksellere 4-LSB ekleme işlemi uygulandığında elde edilen stego-görüntü kalitesi (PSNR değeri 32,803 dB), sıralı ekleme sonucu elde edilen stego-görüntü kalitesiyle

(PSNR değeri 32,807 dB) hemen hemen aynıdır. Bu sonuçlar, önerilen rastgele bit yerleştirme yönteminin etkinliğini ortaya koymaktadır.

1-LSB kullanılması durumu ile karşılaştırıldığında, 4-LSB eklenmiş görüntülerdeki değişim insan gözünün algılayabileceği boyutlara ulaşabilir. Tablo 4'te sunulan görüntüler incelendiğinde, özellikle beyaz bir arka plana sahip olan Cat görüntüsü için sıralı yerleştirme sonucu elde edilen görüntüdeki farklılık çıplak gözle tespit mümkün olabilmektedir. Belirgin özellikleri olan görüntüler için çalışmada önerilen rastgele yerleştirme yöntemi tercih edilebilir.

İşlem sırasında ekstra hesaplamalar nedeniyle veri gizleme için harcanan süre, LSB miktarı ve gizlenen veri boyutu ile genel olarak artmıştır. Sıralı 4-LSB yerleştirme ile karşılaştırıldığında rastgele 4-LSB yerleştirme ile veri gizleme sürelerinde Lena görüntüsü için 0,07; Pepper için 0,069; Cat için 0,128 ve Scenery için 0,149 oranında artış görülmüştür. Diğer taraftan, verilerin görüntü içerisinde rastgele piksellerin en az anlamlı bitlerine atanması daha yüksek güvenlik seviyesi sunmaktadır.

Sıralı ve rastgele yerleştirme işlemleri sonucunda yaklaşık %98 ve üzeri SSIM değerine sahip stego görüntüler elde edilmiştir. Kullanılan LSB miktarının ve gizlenen veri boyutunun artması ile elde edilen PSNR değerlerinde bir düşüş gözlense de stego görüntüler, kabul edilebilir bir görsel kaliteye (30 dB ve üzeri) sahiptir. Ayrıca rastgele yerleştirme ile ihmal edilebilir derecede bir gürültü sağlanmıştır. Elde edilen deneysel sonuçlar, literatürdeki çalışmalar ile de karşılaştırılmış ve Tablo 5'te sunulmuştur.

Tablo 5. Diğer araştırmacılar tarafından elde edilen sonuçlar ile çalışmanın karşılaştırılması

Referans	Yöntem	Görüntü	Gizlenen Veri (bit)	PSNR (dB)	SSIM
(Koçak, 2019)	RG ve sıralı 2-LSB	Lena	1.046.519	29,885	0,986
		Pepper	1.046.519	27,963	0,984
(Ansari vd., 2020)	1-LSB	Lena	15.160	68,45	-
		Pepper	15.160	67,09	-
	3-LSB	Lena	15.160	51,89	-
		Pepper	15.160	51,89	-
	4-LSB	Lena	15.160	46,28	-
		Pepper	15.160	46,28	-
(Doğan vd., 2016)	Sıralı 2-LSB	Lena	12.282	34,34	0,997
(Solak ve Altınışık, 2019)	RGBBGRRG ve 1-LSB	Lena	698.984	51,656	0,99982
		Pepper	698.984	51,621	0,99981
(Balkesen ve Koçer, 2020)	RGB ve rastgele 1-LSB	Lena	693.600	51,29	0,98
(Emam vd., 2016)	3 piksel, BG ve rastgele 2-1-2 LSB	Lena	349.520	51,83	-
		Pepper	349.520	51,85	-
Bu çalışma	RGB ve rastgele 1-LSB	Lena	786.432	51,143	0,997
		Pepper	786.432	51,133	0,997
	RGB ve rastgele 3-LSB	Lena	2.359.296	38,394	0,994
		Pepper	2.359.296	38,285	0,990
	RGB ve rastgele 4-LSB	Lena	3.145.728	32,678	0,991
		Pepper	3.145.728	32,851	0,981

Koçak (2015) tarafından Lena ve Pepper görüntüleri üzerinde yapılan testlerde, R ve G kanallarına 2-LSB (bir piksel için toplam 4 bit) ekleme ile 1.046.519 bit yerleştirme yapılmıştır. Belirtilen görüntüler için sırasıyla PSNR değerleri 29,885 dB - 27,963 dB ve SSIM değerleri 0,986 ve 0,984 olarak elde edilmiştir. Çalışma kapsamında R, G ve B kanalları için 3-LSB (bir piksele toplamda 9 bit) ekleme yöntemi ile 2.359.296 bit yerleştirilmiş ve PSNR değeri en düşük 51,133 dB ve SSIM değeri 0,997 olarak hesaplanmıştır. Önerilen yöntem ile daha yüksek PSNR ve SSIM değerlerinin bulunması, görüntü üzerinde daha az bozulmaya neden olunduğunu gösterir.

Renk kanallarının her birine 1-LSB yöntemini uygulayan ve toplamda 3 piksele 8 bit yerleştirme yapan Solak ve Altınışık (2019), 698.984 bit veri gizlenmiş stego Lena ve Pepper görüntüleri için PSNR değerlerini sırasıyla 51,656 dB ve 51,621 dB olarak bulmuşlardır. Bu çalışmada, her bir renk kanalına rastgele 1-LSB (toplamda 3 piksele 9 bit) yerleştirme deneyleri sonucunda elde edilen PSNR değerleri ise sırasıyla 51,143 dB ve 51,133 dB'dir. Çalışma kapsamında, (Solak ve Altınışık, 2019) ile oldukça yakın PSNR ve SSIM değerleri elde edilmekle birlikte, daha fazla veri gizlenmiş (786.432 bit) ve bitler rastgele yerleştirildiği için daha fazla güvenlik sunulmuştur. Lena görüntüsünün RGB kanallarına rastgele 1-LSB ekleme işleminin yapıldığı çalışmada

(Balkesen ve Koçer, 2020), 693.600 bit veri gizlenmiş ve 51,29 dB PSNR değeri elde edilmiştir. Rastgele bit yerleştirmenin yapıldığı diğer çalışmada (Emam vd., 2016) ise 349.520 bit veri gizlenmiş ve PSNR değeri 51,83 dB hesaplanmıştır. Önerilen yöntemde, aynı görüntünün RGB kanallarına 1-LSB ekleme ile daha fazla veri gizlenmiş ve buna rağmen PSNR değeri 51,143 dB elde edilmiştir. SSIM değerleri açısından karşılaştırıldığında 0,997 ile görüntü için daha yüksek yapısal benzerlik sağlanmıştır.

5. DEĞERLENDİRME

Çalışmada, hassas verileri gizlemek ve saldırılardan korumak için kriptografi (gizli bilgileri şifreleme) ve steganografi (gizli bilgileri saklama) yöntemlerini kullanan bir güvenlik sistemi önerilmiş ve Sten 0.1 isimli bir uygulama geliştirilmiştir. Uygulama ile kriptografi ve steganografi işlemleri kullanıcı seçimine bağlı olarak ayrı ayrı gerçekleştirilebilmektedir. Ayrıca, LSB ekleme işleminin bir rastgele dizi dikkate alınarak yapılması ile veri gizleme daha güvenli hale getirilmiştir. Çalışmada önerilen yöntemde, rastgele karıştırma sürecinden dolayı çıktının üretilmesi biraz daha yavaş olabilmektedir. Ancak gizlenen metnin elde edilmesi zorlaştırılmış ve ek bir anahtar kullanımı ile ekstra güvenlik sunulmuştur.

Kullanılan LSB miktarı arttıkça normal veya rastgele yerleştirme fark etmeksizin stego görüntüde gözlenen farklılık gözle tespit edilebilir boyutta olacaktır. Özellikle Şekil 4b'deki görüntü gibi baskın renklerin bulunduğu durumlarda bu vaziyet daha da ciddidir. Rastgele yerleştirme yöntemi, özellikle belirgin arka plana sahip görüntüler ve/veya daha fazla LSB kullanılması gereken durumlar için artan güvenlik sunduğundan önerilmektedir. Bununla birlikte normal yerleştirme sonucu üretilen çıktı üzerinde benzer bir etki oluşturduğundan düşük veri gizleme süre artışı göz ardı edilerek artan güvenlik sebebiyle tercih edilebilir.

Çalışma kapsamında geliştirilen Sten 0.1 uygulaması, tıbbi kayıtlar, ihale verileri gibi önemli verilerin saklanması, korunması ve güvenli iletimi amaçlı kullanılabilir. Uygulamada kullanılan klasik şifreleme yöntemlerine modern yöntemler de eklenerek kullanıcılara daha gelişmiş bir güvenlik sunulabilir.

KAYNAKÇA

Abraham, O., ve Shefiu, G. O. (2012). An Improved Caesar Cipher (ICC) Algorithm. *IJESAT International Journal of Engineering Science & Advanced Technology*, 2(5), 1198-1202.

Adee, R., ve Mouratidis, H. (2022). A Dynamic Four-Step Data Security Model for Data in Cloud Computing Based on Cryptography and Steganography. *Sensors*, 22(3), 1109.

Ali, U. A. M. E., Sohrawordi, M., & Uddin, M. P. (2019). A Robust and Secured Image Steganography Using LSB and Random Bit Substitution. *American Journal of Engineering Research (AJER)*, 8(2), 39-44.

Aliyu, A. A. M., ve Olaniyan, A. (2016). Vigenère Cipher: Trends, Review and Possible Modifications. *International Journal of Computer Applications*, 135(11), 46-50.

Ansari, A., Mohammadi, M. S., & Ahmed, S. S. (2020). Digital colour image steganography for PNG format and secured based on encoding and clustering. *International Journal of Engineering Research and Technology*, 13(2), 345-354.

Awadh, W. A., Alasady, A. S., & Hamoud, A. K. (2022). Hybrid Information Security System via Combination of Compression, Cryptography, and Image Steganography. *International Journal of Electrical and Computer Engineering*, 12(6), 6574-6584.

Balkesen, C., ve Koçer,H. E. (2020). Embedding Encrypted Data into an Image with a Random Pixel Layout Approach. *European Journal of Science and Technology*, (Special Issue), 123-130.

Baysan, B., ve Özekes, S. (2023). DLSB - Uzaklaştırılmış En Önemsiz Bit Steganografi. *Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi*, 38 (3), 1725-1736.

Bhardwaj, R., ve Sharma, V. (2016). Image Steganography Based on Complemented Message and Inverted bit LSB Substitution. *Procedia Computer Science,* 93, 832-838.

Diepenbroek, M. (2021). Secret Communication in Antiquity the Spartan Scytale. *Ancient Warfare* XIV-3, 44-47.

Doğan, F., Dağ, R., & Türkoğlu, İ. (2016). İmgeler İçin Farklı Bir Veri Gizleme Yaklaşımı. *Dicle Üniversitesi Mühendislik Fakültesi Mühendislik Dergisi*, 7(3), 501-514.

Emam, M. M., Aly, A. A., & Omara, F. A. (2016). An Improved Image Steganography Method based on LSB Technique with Random Pixel Selection. *International Journal of Advanced Computer Science and Applications*, 7(3), 361-366.

Hammad, R., Latif, K. A., & Amrullah, A. Z. vd. (2022) Implementation of Combined Steganography and Cryptography Vigenère Cipher, Caesar Cipher and Converting Periodic Tables for Securing Secret Message. *Journal of Physics*, 2279(1), 012006(1-6).

Hureib, E. S., ve Gutub, A. A. (2020). Enhancing Medical Data Security via Combining Elliptic Curve Cryptography with 1-LSB and 2-LSB Image Steganography. *IJCSNS International Journal of Computer Science and Network Security*, 20(12), 232-241.

Koçak, C. (2015). Kriptografi ve Stenografi Yöntemlerini Birlikte Kullanarak Yüksek Güvenlikli Veri Gizleme. *Erciyes Üniversitesi Fen Bilimleri Enstitüsü Fen Bilimleri Dergisi*, 31(2), 115-123.

Konyar, M. Z., İlkin, S., Çelik, N., & Sondaş, A. (2018). Steganografi için En Uygun Resmi Belirleyen Uygulama Arayüz Tasarımı. *İleri Teknoloji Bilimleri Dergisi*, 7(1), 83-89.

Macit, H. B., ve Koyun, A. (2020). A New Imperceptible Steganography Method for Grayscale Images. *Mühendislik Bilimleri ve Tasarım Dergisi*, 8 (2), 357-365.

Osman, O. M., Kanona, M. E. A., Hassan, M. K., Elkhair, A. A. E., & Mohamed, K. S. (2022). Hybrid Multistage Framework for Data Manipulation by Combining Cryptography and Steganography. *Bulletin of Electrical Engineering and Informatics*, 11(1), 327-335.

Özbilgin, F., Durmuş, F., & Karagöl, S. (2018). Yazılı Metni Şifreleyip LSB Yöntemi ile Gizleme. *Düzce Üniversitesi Bilim ve Teknoloji Dergisi*, 6(3) (Özel Sayı: UMAS 2017), 676-685.

Putera, A., Siahaan, U., & Rahim, R. (2016). Dynamic key matrix of hill cipher using genetic algorithm. *International Journal of Security and Its Applications*, 10(8), 173-180.

Sara, U., Akter, M., & Uddin, M. (2019). Image Quality Assessment through FSIM, SSIM, MSE and PSNR—A Comparative Study. *Journal of Computer and Communications*, 7, 8-18.

Singh, S., ve Atria, V. K. (2015). Dual Layer Security of Data Using LSB Image Steganography Method and AES Encryption Algorithm. *International Journal of Signal Processing, Image Processing and Pattern Recognition*, 8(5), 259-266.

Solak, S., ve Altınışık, U. (2019). A New Approach for Steganography: Bit Shifting Operation of Encrypted Data in LSB (SED-LSB). *Bilişim Teknolojileri Dergisi*, 12(1), 75-81.

Şahin, A., Buluş, E., & Sakallı, M. T. (2006). 24-Bit Renkli Resimler Üzerinde En Önemsiz Bite Ekleme Yöntemini Kullanarak Bilgi Gizleme. *Trakya Üniversitesi Fen Bilimleri Dergisi*, 7(1), 17-22.

Yakut, S. (2021). Random Number Generator Based on Discrete Cosine Transform Based Lossy Picture Compression. *Naturengs*, 2(2), 76-85.

Yakut, S. (2022a). Steganography Approach Based on the Least Significant Bit Technique. 6th International Artificial Intelligence & Data Processing Symposium, 8-9 Sep. 2022, Malatya, Türkiye, 165-169.

Yakut, S. (2022b). Kayıplı Resim Sıkıştırma Algoritmalarını Temel Alan Rastgele Sayı Üreteci. *Adıyaman Üniversitesi Mühendislik Bilimleri Dergisi*, 9(18), 571-580.

URL-1: https://i.pinimg.com/originals/f9/25/e1/f925e13343ffc8726316f519b3619424.png [Erişim Tarihi: 11.05.2023]

URL-2: https://demo.joomlalabs.com/images/slideshow/1280x853/stars-345902_1280.jpg [Erişim Tarihi: 11.05.2023]