



**SAYISAL İMGELERDE GÜVENLİ VERİ GİZLEMEYE
YÖNELİK UYGUN MASKE BELİRLEME
Yıldıray YİĞİT**

**Yüksek Lisans Tezi
Yazılım Mühendisliği Anabilim Dalı
Danışman: Doç. Dr. Murat KARABATAK
MAYIS-2018**

T.C
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

SAYISAL İMGELERDE GÜVENLİ VERİ GİZLEMeye YÖNELİK UYGUN MASKE
BELİRLEME

YÜKSEK LİSANS TEZİ

Yıldıray YİĞİT

(141137108)

Tezin Enstitüye Verildiği Tarih : 25 Nisan 2018

Tezin Savunulduğu Tarih : 14 Mayıs 2018

Tez Danışmanı : Doç. Dr. Murat KARABATAK (F.Ü)

Diğer Jüri Üyeleri : Dr. Öğr. Üyesi Muhammet BAYKARA (F.Ü)

Dr. Öğr. Üyesi Özal YILDIRIM (M.Ü)

MAYIS-2018

ÖNSÖZ

Tez çalışmamda başta zorlukları aşma noktasında benden yardımlarını esirgemeyen danışman hocam Doç. Dr. Murat KARABATAK'a, bugüne kadar olduğu gibi bugün de çalışmamda maddi, manevi desteğini esirgemeyen aileme ve destekleriyle bana güç veren iş arkadaşlarıma, son olarak her zaman bana destek olan çok değerli eşim Gülin TİTİRİNLİ YİĞİT'e sonsuz teşekkürlerimi sunuyorum.

Yıldıray YİĞİT

ELAZIĞ-2018



İÇİNDEKİLER

	<u>Sayfa No</u>
ÖNSÖZ	I
İÇİNDEKİLER	II
ÖZET	IV
SUMMARY	V
ŞEKİLLER TABLOSU	VI
TABLolar LİSTESİ	VII
KISALTMALAR	VIII
1. GİRİŞ	1
2. SAYISAL GÖRÜNTÜ VE GÖRÜNTÜ FORMATLARI	3
2.1. Temel Kavramlar	3
2.1.1. Piksel	3
2.1.2. Çözünürlük	3
2.1.3. Nokta ve Nokta Aralığı	3
2.1.4. İnç Başına Düşen Çizgi Sayısı	3
2.1.5. İnç Başına Düşen Nokta Sayısı	3
2.2. Görüntü Formatları	4
2.2.1. BMP	5
2.2.2. GIF	5
2.2.3. PNG	5
2.2.4. JPEG	5
2.2.5. TIFF	6
2.2.6. PICT	6
2.2.7. EPS	6
2.2.8. PSD	7
3. STEGANOĞRAFİ VE STEGANOĞRAFİ YÖNTEMLERİ	8
3.1. Steganografi	8
3.2. Steganografinin Alt Alanları	11
3.3. Steganografinin Kullanıldığı Alanlar	11
3.3.1. Metin Steganografi	12
3.3.1.1. Açık Alan Yöntemleri	13

3.3.1.1.1. Satır Kaydırma Kodlaması	13
3.3.1.1.2. Kelime Kaydırma Kodlaması	13
3.3.1.1.3. Gelecek Kodlaması	14
3.3.1.2. Yazımsal Yöntemler	15
3.3.1.3. Anlamsal Yöntemler	15
3.3.2. Ses Kodlaması	15
3.3.2.1. En Az Ağırlıklı Bit Kodlama	15
3.3.2.2. Eşlik Biti Kodlama	15
3.3.2.3. Aşama Kodlaması	16
3.3.2.4. Tayf Yayılması	16
3.3.2.5. Yankı Veri Gizleme	17
3.3.3. Görüntü Steganografi	17
3.3.3.1. En Önemsiz Bite Ekleme	17
3.3.3.2. Maskeleye ve Filtreleme	19
3.3.3.3. Algoritmalar ve Dönüşümler	20
3.4. Steganografi Alanında Yapılan Çalışmalar	20
4. MATERYAL VE METOT	27
4.1. Bilgiyi Gizle Bölümü	29
4.2. Bilgiyi Göster Modülü	37
4.3. Analiz Modülü	38
5. BULGULAR	40
6. SONUÇLAR VE TARTIŞMA	44
KAYNAKLAR	45
ÖZGEÇMİŞ	50

ÖZET

Bilgi güvenliğine duyulan ihtiyaç her geçen gün artmaktadır. Bilginin güvenliği sadece kişisel bilgilerin güvenliğiyle kalmayıp aynı zamanda kurumların ve ülkelerin iletişiminde de kritik role sahiptir. Hatta savaş durumlarında ülkelerin en büyük sorunları yaşadığı alan bilgi güvenliğinin sağlanamamasıdır. İletişimin gizliliğinde ve mesajın güvenli bir şekilde iletilmesinde kullanılan iki önemli teknik; kriptografi ve steganografidir. Kriptografide temel amaç, bilginin yetkisiz kişiler tarafından ulaşıp çözülmesini şifreleyerek engellemektir. Steganografi çalışmalarında temel amaç ise gizlenen verinin yetkisiz kişilerce fark edilmesini engellemektir.

Kelime anlamı olarak “Örtülü Yazı” anlamına gelen Stenografi, Yunan sözcüklerinden türetilmiş bir kelimedir. Steganografi yöntemlerindeki temel amacın gerçekleşmesi için bilgi gizlenecek ortamın en az değişime uğraması gerekmektedir. Steganografi ortamları ses, görüntü, video veya metin dosyalarından herhangi biri olabilir. Bu tez çalışmasında, sayısal imgelere bilgi gizleme yöntemlerinden biri olan en önemsiz bite ekleme yöntemi üzerinde durulmuştur. En önemsiz bite ekleme yöntemi ile veri gizlenirken örtü dosyasında oluşan değişikliklerin en aza indirgenmesi ve gizlenen verinin elde edilmesinin zorlaştırılması tezin ana hedefini oluşturmaktadır. Bu nedenle, tez çalışması kapsamında, veri gizleme işlemini uygun bir maske kullanarak yapan yeni bir yaklaşım önerilmektedir. Önerilen yöntem farklı imgeler üzerinde test edilmiş ve elde edilen sonuçlar tezin sonunda sunulmuştur.

Anahtar Kelimeler: Steganografi, Görüntü, Ses, Video, Uygun Maske.

SUMMARY

Determination of the Appropriate Mask for Hiding Secure Data on Digital Images

The need for information security is increasing day by day. The security of information is not only the security of personal information but also has a critical role in the communication of institutions and countries. Even in the case of war, the greatest weakness of the countries is that they cannot secure the information. Two important technical cryptography and steganography used in communicating the confidentiality and secure transmission of the message. The main purpose in cryptography is to prevent information from being accessed and decrypted by unauthorized persons. The main purpose of steganography studies is to prevent the hidden data from being recognized by unauthorized persons.

Steganography is derived from Greek words, meaning "Covered Writing". In order to realize the basic purpose of the steganography methods, the environment in which information is to be hidden needs to be changed at least. Steganography media can be any of audio, video, video, or text files. In this thesis study, least significant bit method which is one of the methods of hiding information in digital imagery is examined. When least significant bit method is hidden, a new method of data hiding is proposed by creating optimum mask in order to minimize the changes occurring in the cover file and to make it difficult to obtain hidden data.

Key Words: Steganography, Image, Audio, Video, Appropriate Mask.

ŞEKİLLER TABLOSU

	<u>Sayfa No</u>
Şekil 2.1.	72 DPI ile 300 DPI resim arasındaki fark..... 4
Şekil 2.2.	Sayısal resmin yapısı 4
Şekil 2.3.	a) Orijinal görüntü, b) Yoğunluk, c) Mavi-Sarı ve d) Kırmızı-Yeşil 6
Şekil 3.1.	Steganografinin çalışma şekli 8
Şekil 3.2.	Fridrich üçgeni..... 9
Şekil 3.3.	a) Orijinal haber metni b) Kullanılan maske 10
Şekil 3.4.	Cardan ızgarası uygulanmış metin 10
Şekil 3.5.	Steganografi metotlarının sınıflandırılması 12
Şekil 3.6.	Satır kaydırma kodlaması örneği 13
Şekil 3.7.	a) Satır kodlaması yapılması b) Çizgiler olmadan görünüşü..... 14
Şekil 3.8.	a) Orijinal metin b) Kodlama yapılmış metin c) Abartılmış hali 14
Şekil 3.9.	Son iki bite veri gizleme işlemi 19
Şekil 4.1.	LSB yöntemiyle veri gizlenmesi 27
Şekil 4.2.	Program akış diyagramı..... 28
Şekil 4.3.	Program ana ekranı 29
Şekil 4.4.	Bilgi gizleme blok diyagramı 30
Şekil 4.5.	Resim dosyası renk kanalları..... 31
Şekil 4.6.	Matrisin NxN çerçeveye bölünmesi 31
Şekil 4.7.	Çerçevelerdeki 1 ve 0'ların gruplanması ve sayılması..... 32
Şekil 4.8.	Binary veri dizisi 32
Şekil 4.9.	Binary dizinin matris formatına çevrilmesi 33
Şekil 4.10.	Maske oluşturma algoritması 33
Şekil 4.11.	Veri gizlenecek sayısal imge (20x20 çözünürlüklü) 34
Şekil 4.12.	“Selam” bilgisinin binary sisteme çevrilmiş hali 34
Şekil 4.13.	Sayısal imgenin R kanalına ait pixel değerleri 35
Şekil 4.14.	Matris formatına çevrilmiş “Selam” verisi..... 35
Şekil 4.15.	"Selam" verisine ait bitlerin renk kanallarına dağılımı 35
Şekil 4.16.	Verilerin renk kanallarına dağıtılması ve gruplanması 36
Şekil 4.17.	Oluşturulan maske değerleri..... 36
Şekil 4.18.	Veriyi elde etme sözde kodları 38
Şekil 5.1.	Gizlemek için kullanılan veri 41

TABLÖLAR LİSTESİ

	<u>Sayfa No</u>
Tablo 4.1. Boyut Karşılaştırması	34
Tablo 4.2. Orijinal resim ve veri gizlenmiş resim.....	37
Tablo 5.1. LSb yöntemi yöntemi ile elde edilen sonuçlar.....	40
Tablo 5.2. Uygulamada kullanılan resim dosyaları	41
Tablo 5.3. Resimler ve bu resimlerin kanalları için oluşturulmuş maskeler.....	42
Tablo 5.4. Elde edilen PSNR ve MSE değerleri	43



KISALTMALAR

AVI	: Audio Video Interleave	Ses Görüntü Birleşimi
DCT	: Discrete Cosine Transform	Ayrık Kosinüs Dönüşümü
JPEG	: Joint Photographic Experts Group	Birleşmiş Fotoğraf Uzmanları Grubu
GIF	: Graphics Interchange Format	Grafik Değişim Biçimi
LSB	: Least Significant Bit	En Önemsiz Bit
MMS	: Multimedia Messaging Service	Multimedya Mesajlaşma Servisi
MPEG	: Motion Pictures Experts Group	Film Uzmanları Grubu
MP3	: MPEG Layer III	Film Uzmanlar Grubu Ses Katmanı 3
PCM	: Pulse Code Modulation	Darbe Kodu Modülasyonu
PNG	: Portable Network Graphics	Taşınabilir Ağ Grafikleri
WAV	: Waveform Audio File Format	Dalga Biçimli Ses Dosyası Biçimi
PSNR	: Peak signal-to-noise ratio	Tepe Sinyal-Gürültü Oranı
MSE	: Mean Squared Error	Ortalama Kare Hatası
PSD	: Photoshop Documents	Photoshop Belgeleri
BMP	: Bitmap	Resim Dosyası Türü
EPS	: Encapsulated Postscript	Vektörel Resim Formatı
RS	: Rescaled Range Analysis	Ölçeklendirilmiş Aralık analizi

1. GİRİŞ

Günümüzde teknolojinin hızla gelişmesiyle ve internetin yaygınlaşmasıyla internet ortamında veri paylaşımı oldukça artmıştır. Dünyanın her yerinden resim, video, ses, metin, vb. birçok veriyi içeren dosyalar saniyeler içinde paylaşılmaktadır. Hayatımızın kolaylaşmasını sağlayan bu iletişim ağı, içerisinde çok ciddi güvenlik açıklarıyla birlikte gelmektedir. Günümüzde bu açıklardan dolayı bilgi güvenliği dikkatle ele alınması gereken bir problem olarak önümüze gelmektedir. İletilen mesajlar üçüncü kişiler tarafından kolayca elde edilebilmekte ve değiştirilebilmektedir. Bunu engellemek için her geçen gün yeni yöntemler ve yazılımlar ortaya çıkmakta, çeşitli mekanizmalar oluşturulmaktadır. İletişimin gizliliğinde ve mesajın güvenli bir şekilde iletilmesinde kullanılan iki önemli teknik vardır [1]. Bunlardan birincisi kriptografi'dir. Kriptografi, iletilmek istenen verinin bir anahtar yardımıyla belirlenmiş bir algoritmayla şifrlenerek alıcıya iletilmesidir. Alıcı eğer anahtara sahipse gelen şifrelenmiş mesajı alır ve çözer. Eğer alıcı anahtara sahip değilse mesajı çözemeyecektir. Fakat geliştirilen yeni yöntemlerle birlikte şifreler de kolayca kırılmakta ve mesajlar okunabilmektedir.

Bilgisayar ortamında bilginin güvenli bir şekilde iletilmesi için kullanılan ikinci yöntem ise steganografidir. Steganografi, kelime anlamı olarak "Örtülü Yazı" anlamına gelen Yunan sözcüklerinden türetilmiştir [2]. Steganografide amaç önemli bilgiyi sıradan örtü nesnesi (resim, ses, video, metin vb.) içine görünmeyecek şekilde yerleştirerek stego nesneyi oluşturmak ve bunu alıcıya iletmektir. Gizlenen veri çıplak gözle belli olmadığı için yetkisiz kişiler tarafından görülmeyecektir. Steganografi; dilbilim steganografi ve teknik steganografi olmak üzere ikiye ayrılır. Dilbilim steganografi metin tabanlı steganografidir. Teknik steganografi ise resim, ses ve videolara uygulanan steganografidir. Habersiz izleyiciler için stego nesne, örtü nesnenin içeriğini sergiler. Alıcılar için stego nesne ise önemli bilgiyi gizler.

Bilgi gizleme tekniği çok eski yıllardan beri kullanılmaktadır ve hala kullanılmaya devam etmektedir. Günümüzde teknolojinin gelişmesiyle birlikte bilgi gizleme yöntemleri de gelişmekte ve yeni yöntemler ortaya konulmaktadır. Bilgi gizleme teknikleri günümüzde tıbbi görüntüler, ses dosyaları, metin belgeleri vb. birçok yerde sıklıkla kullanılmaktadır.

Bir steganografik sistemin güvenilirliği birden fazla ölçüte göre değerlendirilmektedir. Bunlar veri gizlemenin gizlenecek ortamı (cover object) değiştirme miktarı, ortamın içine

saklanabilecek bilginin miktarı ve ne kadar dayanıklılığı olduğudur. Dayanıklılık gücü, ortama yapılan görsel ve istatistiksel saldırılardan ne kadar az etkilendiği ile ölçülmektedir. Gelişen teknoloji ile birlikte birçok veri gizleme yöntemi ortaya çıkmıştır. Veri gizleme yöntemlerindeki çeşitliliğin artması birçok steganaliz yöntemlerinin de geliştirilmesini gerektirmiştir. Her veri gizleme yöntemi kendine özgü bir yol kullandığı için bunları fark etmede izlenecek yollar da farklıdır. Bir steganografik yöntemde özel geliştirilmiş steganaliz yöntemi bir başka steganografik yöntem için çalışmamaktadır [3].

Yapılan bu tez çalışmasında veriyi sayısal görüntü içerisine gizlerken görüntü üzerinde en az değişikliği sağlayacak maske kullanımını önererek, steganografi yöntemlerinden biri olan LSB tekniğinin güçlendirilmesi amaçlanmaktadır. Önerilen yöntemde elde edilen maskeyle aynı zamanda veri XOR işlemine tabi tutularak gizlenmekte ve bu sayede veri güvenliği artırılmaktadır. Önerilen yöntemi test edebilmek için veri dizisi belirlenmiştir. Belirlenen veriyi taşıyacak olan nesne, sayısal görüntü olup, bilgi gizlendikten sonra görüntü üzerindeki bozulmanın en az olması ve saldırılara karşı dayanıklı olması hedeflenmektedir. Tez içeriği genel hatlarıyla şu şekildedir: ikinci bölümde sayısal görüntü özellikleri, üçüncü bölümde steganografi kavramı ve veri gizleme yöntemleri üzerine yapılan çalışmalar, dördüncü bölümde materyal ve metot, beşinci bölümde elde edilen bulgular ve altıncı bölümde sonuç ve tartışma kısmına yer verilmiştir.

2. SAYISAL GÖRÜNTÜ VE GÖRÜNTÜ FORMATLARI

2.1. Temel Kavramlar

Sayısal görüntü, elektronik ortamlarda görüntü dosyalarına verilen isimdir. Bilgisayar ortamındaki bu görüntü dosyaları, farklı dosya türleri olarak dijital ortamlarda depolanmaktadır. Depolanan dosya türlerine geçmeden önce görüntü dosyalarıyla ilgili bilinmesi gereken temel kavramlar bulunmaktadır. Bu temel kavramlar piksel, çözünürlük, LPI, DPI ve nokta aralığıdır.

2.1.1. Piksel

Sayısal resimler noktalardan oluşmaktadır. Resim yakınlaştırıldığında veya büyütüldüğünde bu noktaların kare şeklinde olduğu görülmektedir. Bu kare şeklindeki noktalara piksel denilmektedir. Bir piksel kırmızı, mavi ve yeşil renklerin birleşiminden oluşmaktadır. Bir görüntü ne kadar çok pikselden oluşuyorsa görüntü de o kadar net olmaktadır [4, 5].

2.1.2. Çözünürlük

Görüntü teknolojilerinde görüntüyü oluşturan piksel sayısının tamamına çözünürlük denir. Çözünürlük ne kadar yüksek ise görüntü kalitesi de o kadar iyidir [4, 5].

2.1.3. Nokta ve Nokta Aralığı

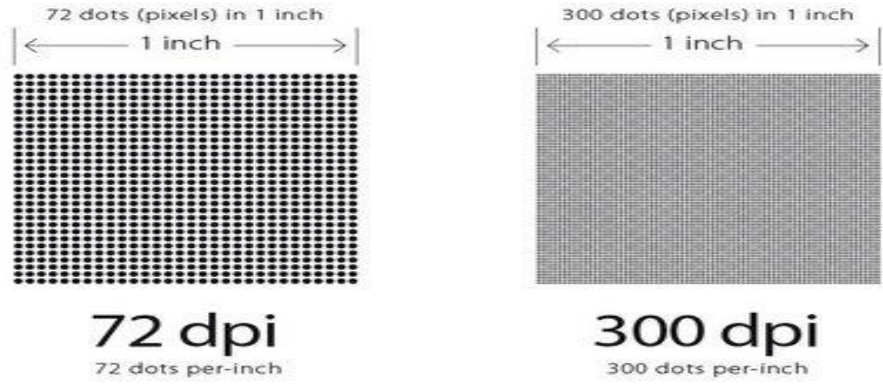
Pikselli oluşturan kırmızı, mavi ve yeşil renklerinin her birisine nokta denilmektedir. Pikselli oluşturan bu renklerin arasındaki mesafeye ise nokta aralığı denilmektedir. Nokta aralığı ne kadar az ise görüntü o kadar iyi kabul edilir.

2.1.4. İnç Başına Düşen Çizgi Sayısı

Bir inç'lik (2.54cm) alana düşen satır sayısına LPI(Line Per Inch) denilmektedir. Bu satırlar noktalardan oluşmaktadır. Satırları oluşturan nokta dizilerine tram denilmektedir. LPI çözünürlüğü tram sıklığını elde etmemizi sağlar [3, 6].

2.1.5. İnç Başına Düşen Nokta Sayısı

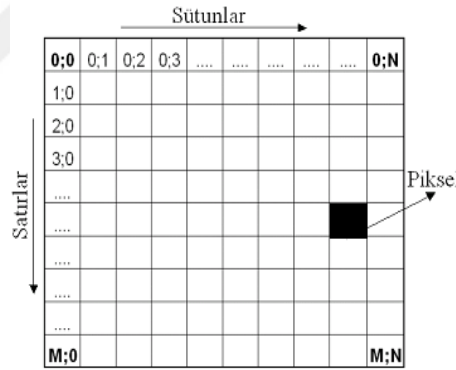
Bilgisayar ekranında bir inç olarak adlandırılan 2,54 cm x 2,54 cm kare içinde bulunan nokta sayısına DPI(Dot Per Inch) denir. 1 inç kareye daha fazla kare sığdırabilmek için noktaların gittikçe ufalması gerekir. Bir inç kareye ne kadar çok ufak nokta yerleşirse baskı da o kadar kaliteli olmuş olur. Farklı DPI arasındaki fark Şekil 2.1'de gösterilmiştir [4].



Şekil 2.1. 72 DPI ile 300 DPI resim arasındaki fark

2.2. Görüntü Formatları

Görüntü dosyaları, görüntüye ait yükseklik, genişlik, boyut gibi bilgilerin bulunduğu başlık ve görüntünün piksellerinden oluşan iki kısımdan oluşur. Görüntü x yükseklik ve y genişlik olmak üzere $x*y$ adet pikselden oluşmaktadır. Bir sayısal resmin yapısı Şekil 2.2’de gösterilmektedir [7].



Şekil 2.2. Sayısal resmin yapısı

Oluşturulan resimler diğer programlarla da paylaşılabilmesi veya kullanılabilmesi için farklı bir formatta saklanmalıdır. Görüntü dosyalarında yaygın olarak kullanılan görüntü dosyası biçimlerinden bazıları aşağıda açıklanmıştır.

2.2.1. BMP

Windows 3.1 ve Windows 95 ile birlikte gelen msPaint programının işlediği 1-24 bit renk derinliği içerebilen dosya türüdür. BMP formatında başlangıçta sıkıştırma bulunmamaktaydı fakat daha sonra opsiyonel olarak kayıpsız sıkıştırma eklendi. Bu sıkıştırmada herhangi bir veri kaybı bulunmamaktadır. BMP formatı Paintten başka görüntüleme programı olmayan bilgisayarlarda kullanılır [4].

2.2.2. GIF

GIF dosyası, CompuServe firması tarafından geliştirilmiştir. GIF dosya türü görüntüyü sıkıştırarak kaydeder. Sıkıştırmayı bulunan renk sayısını 256'ya ölçeklendirerek yapar. Eğer resimde 256 ve daha az farklı renk varsa kayıpsız olarak sıkıştırılabilir fakat 256'dan fazla farklı renk varsa kayıplı olarak sıkıştırılacaktır [5].

2.2.3. PNG

PNG dosyası, görüntüyü gerçek manada kayıpsız olarak sıkıştırabilen dosya türlerinden birisidir. Hufmman algoritmasını ve index tablosunu kullanarak sıkıştırma işlemini gerçekleştirir. Sıkıştırma için satır temelli olarak karar verilir [5, 6]. PNG dosyalarında saydamlık bilgileri alfa kanalı içinde saklanmaktadır. PNG dosyalarını kaydetmeden önce sıkıştırmak istendiğinde çeşitli sıkıştırma algoritmaları kullanılabilir.

2.2.4. JPEG

JPEG, Joint Photographic Experts Group (Birleşik Fotoğraf Uzmanları Grubu) tarafından standartlaştırılmış bir sayısal görüntü kodlama biçimidir. Bu biçim, 1994 yılında ISO10918-1 adıyla standartlaşmıştır [8, 9].

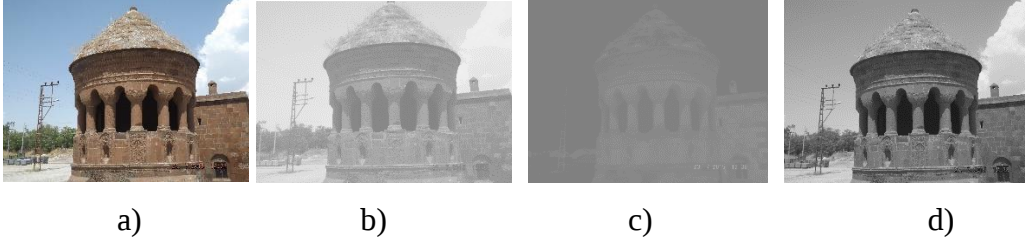
JPEG dosya biçimi internette kullanılan popüler bir dosya türüdür. Jpeg dosya türü kayıplı sıkıştırma yapar yani orijinal resmin tekrar elde edilmesi mümkün değildir. JPEG dosyası sıkıştırma işlemi oldukça karmaşıktır.

Sıkıştırma işleminin ilk adımı olarak RGB'nin Yoğunluk(Y), Mavi - Sarı(Cb), Kırmızı - Yeşil(Cr) değerlerine dönüştürülmesi gerekir. RGB'yi YCbCr'ye dönüştürürken kullanılan formül Denklem 2.1, Denklem 2.2 ve Denklem 2.3'te gösterilmiştir [9].

$$Y=(0.257*R)+(0.504*G)+(0.098*B)+16 \quad (2.1)$$

$$Cb= -(0.148*R)-(0.291*G)+(0.439*B)+128 \quad (2.2)$$

$$Cr=(0.439*R)-(0.368*G)-(0.071*B)+128 \quad (2.3)$$



Şekil 2.3. a) Orijinal görüntü, b) Yoğunluk, c) Mavi-Sarı ve d) Kırmızı-Yeşil

Dönüştürme işlemi yapılmış örnek Şekil 2.3'te gösterilmektedir. Dönüştürme işleminden sonra resim 8x8 bloklara ayrılır ve her bloğa DCT (Ayrık Kosinüs Dönüşümü) işlemi uygulanır. İşlemin bu basamağında resim eski haline kayıpsız olarak döndürülebilir. Bu basamaktan sonra her blok JPEG'in önerdiği nicemleme tablosundaki değerlere birebir bölünür ve yuvarlanır. Nicemleme, sayısal işaret ve görüntü işleme alanlarında sürekli ya da büyük bir değerler kümesinin daha az sayıda ve ayrık değerlere eşlenmesidir. Kayıplı sıkıştırma bu basamakta oluşur ve resim eski haline dönüştürülemez. Bunlar kullanarak sıkıştırma işlemi tamamlanır ve Huffman ağacı ile resim son halini alarak JPEG'e dönüşmüş olur [5].

2.2.5. TIFF

Bilgisayarlar arası ortak bir dosya formatı olan TIFF formatı tüm programlar tarafından desteklenmektedir. Bu formatta kaydedilmiş olan dosyalar, herhangi bir uygulama ile açıldığında görüntünün ve zeminin renk değerleri üzerinde azaltma ve değiştirme gibi olanaklar sağlamaktadır [6]. Örneğin, farklı renklerde kullanılacak olan bir görüntü ya da logo TIFF formatında kaydedilip sayfaya yapıştırıldığı zaman renkleri değiştirilebilir. Bu yöntemle her renk değişikliği yapıldığında kendisini oluşturan programda dosyayı açıp yeniden renkleri değiştirilerek yeni tiff dosyaları yaratılmamış olur.

2.2.6. PICT

PICT bilgisayar ortamında her programın kullanabildiği dosya formatıdır. Herhangi bir programa aktarıldığı zaman imge bilgisi imgeyle birlikte sayfaya dâhil olmaktadır. Görüntü dosyası bilgisayarınızdan silinse bile baskı yapma aracı sayfayı basarken PICT dosyasını aramaz.

2.2.7. EPS

EPS, çizim ve sayfa düzenleme programlarının hemen hemen hepsinin desteklediği görüntü dosyası formatıdır. Resim düzenleme programlarından biri olan Photoshop programında, görüntü BMP formatında kaydedilirken beyazları saydam yap seçeneği seçilerek kaydedildiği zaman, dosyada bulunan beyaz bölgeler saydam olarak

kaydedilecektir. EPS formatında kaydedilecek bir renkli görüntü dosyası CMYK modunda geçirilmiş olursa olası baskı problemleri önlenmiş olur.

2.2.8. PSD

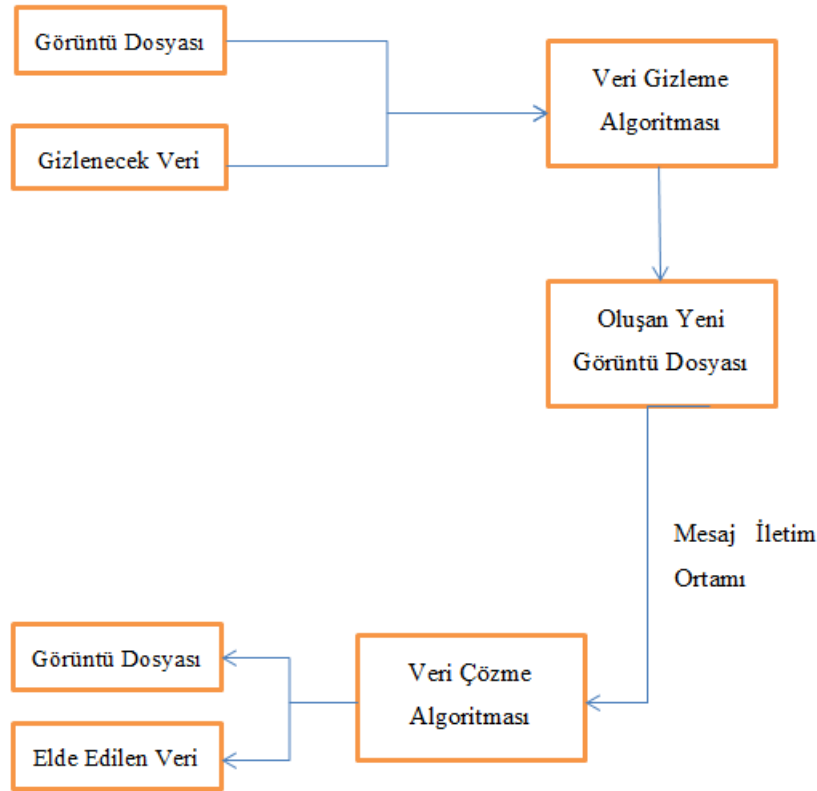
Resim düzenleme programlarından biri olan Photoshop programına özel formata PSD (Photoshop Document) denilmektedir. PSD çok sayıda alfa kanalını, yolu ve katmanı desteklemektedir. PSD dosyalarının desteklediği biçimlerden birkaçı şunlardır; ikili dosya, indekslenmiş renk, gerçek renk RGB, CMYK, Lab. Çalışma yapılan görüntü dosyalarına ait PSD dosyalarını saklamak, kişiye daha sonra bu görüntü dosyalarında yapılacak olan düzeltmelerde çok yardımcı olacaktır [10].



3. STEGANOGRAFI VE STEGANOGRAFI YÖNTEMLERİ

3.1. Steganografi

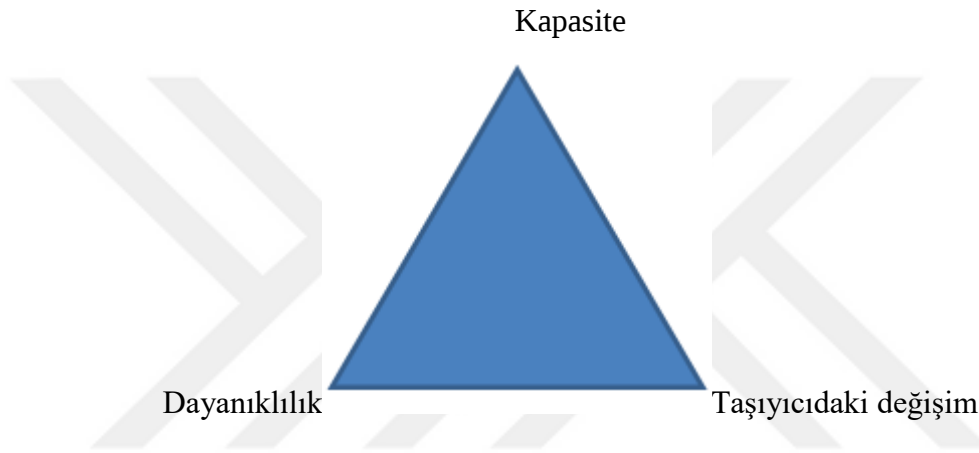
Steganografi, bir verinin başka bir veri içine gömülerek gizlenmesidir. Steganografi, kelime olarak “Örtülü Yazı” anlamına gelen Yunan sözcüklerinden türetilmiştir [2, 11]. Steganografide amaç, önemli bilgiyi sıradan örtü nesne (resim, ses, video, metin vb) içine gizleyerek stego nesneyi oluşturmaktır. Habersiz izleyiciler için stego nesne, örtü nesnenin içeriğini sergiler. Alıcılar için stego nesne önemli bilgiyi kendi içerisinde saklar. İşlem üç aşamadan oluşmaktadır. Önemli bilgi stego nesnesinin içine gizlenir, alıcıya iletilir ve alıcı stego nesnesinden doğru algoritmayla önemli veriyi elde eder. Bu işlem adımları Şekil 3.1’de gösterilmektedir [6].



Şekil 3.1. Steganografinin çalışma şekli

Algılanabilirlik, dayanıklılık ve kapasite steganografik yöntemler için yöntemin başarısını etkileyen faktörlerdir. Kapasite örtü dosyasına ne kadar veri gizleyebileceğimizi ifade eder ve bunun belirli bir sınırı vardır. Bu sınır ortam dosyasının türüne ve boyutuna göre değişmektedir [12]. Genel olarak bunlar birbirlerini olumsuz etkileyen faktörlerdir.

Bu ölçütleri steganografi ve steganaliz alanında birçok çalışmalar yapan Jessica Fridrich ortaya atmıştır. Fridrich üçgeni Şekil 3.2’de gösterilmiştir [13]. Algılanabilirlik, ortamda gizli veri taşınıp taşınmadığının yetkisi olmayan kişiler tarafından sezgilerle veya çeşitli araçlarla algılanamamasıdır. Dayanıklılık, stego nesne üzerinde yapılan işlemlere rağmen gizli bilginin hala elde edilebilir olmasını ifade eder. Kapasite ise ne kadar önemli bilgiyi gizleyebileceğimizi gösterir. Kapasite bilgi gizlenecek ortamın boyutuyla sınırlıdır. Ortamın boyutundan daha büyük bir bilginin eklenilmesi mümkün değildir. Ayrıca ne kadar çok bilgi eklemeye çalışılırsa bilgi gizlenen ortam bozulacağından bilginin fark edilmesi de o kadar kolay olacaktır [12].



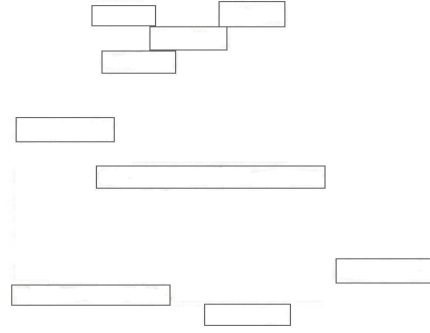
Şekil 3.2. Fridrich üçgeni

Günümüzde birçok steganografik yöntem bulunmaktadır ve bu metotların başarısının ölçülmesi için yukarıdaki üçgen referans alınmaktadır [13].

Steganografi ilk olarak M.Ö. 1500 yıllarında kullanıldığı bilinmektedir [14]. Histaiacus göz hapsinde iken kölesinin saçlarını kazıtmış ve mesajı kölesinin kafasına dövme olarak yapmıştır. Kölenin saçları uzadıktan sonra kölesini mesajı göndereceği yere göndermiştir. Böylece mesaj güvenli bir şekilde alıcıya ulaşmıştır. İtalyan Matematikçi Jerome Cardan, 500 yıl önce antik Çin'den kalma gizli yazı yöntemini keşfetti. Üzerinde delikler bulunan maske ve mesaj birlikte kullanılarak gizli metin ortaya çıkarılıyordu. Bu yöntem Cardan ızgarası olarak adlandırılmaktadır [2, 6, 15].

KIDEM Tazminatı Fonu'nda uzlaşma çıkmadı. Çalışma ve Sosyal Politikalar Bakanı **Faruk Çelik**, bugün taraflarla son kez bir araya geliyor. Sosyal taraflar kendilerine verilen 20 günlük sürede çalışmalarını tamamladı. "İşçinin kıdemine dokunmak grev sebebidir" diyen Türk-İş kararından geri adım atmıyor, Hak-İş de devlet garantisinde ısrarını koruyor. Bugün masada uzlaşma sağlanamazsa, kıdem tazminatı fonu çalışma yaşamına yönelik paketten çıkarılacak.

Çalışma Bakanı Çelik, toplantıda uzlaşma sağlanırsa konuyu Başbakan Erdoğan'a götüreceğini, olmazsa da düzenlemeyi getirmeyeceklerini bildirdi.



Şekil 3.3. a) Orijinal haber metni b) Kullanılan maske

Şekil 3.3'te a) Orijinal haber metni ve b) kullanılan maske belirtilmektedir. Belirlenen maske metnin üzerine konulduğu zaman, dışarıdan bakıldığında masum bir haber gibi görünen metinden gizlenen veri ortaya çıkarılmaktadır. Maske uygulanmış ve gizli metnin ortaya çıkarılmış hali Şekil 3.4'te gösterildiği gibidir.

Faruk bugün
geliyor.
verilen
kararında
uzlaşma sağlanamazsa
götüreceklerini Erdoğan'a
bildirdi.

Şekil 3.4. Cardan ızgarası uygulanmış metin

Arabistan'da eserlerin İngilizceye çevrilmesi için başlatılan bir projede ise 1200 yıl önce yazıldığına inanılan gizli yazışmaların olduğu Arapça el yazmaları bulunmuştur. Bu yazıların bir kısmı da Türkiye ve Almanya'da bulunmuştur [2, 16]. Günümüzde de terör örgütleri ve istihbarat birimleri haberleşirken veya istihbarat toplarken steganografik yöntemlerden faydalanmaktadır. Bu bağlamda hem tekniklerin nasıl uygulandığını hem de uygulanan yöntemi nasıl çözeceklerini bilmeleri büyük önem arz etmektedir. Bu yöntemleri sağlayan uygulamalara ulaşmak ve uygulamak internetin yaygınlaşmasıyla oldukça kolaylaşmıştır.

3.2. Steganografinin Alt Alanları

Steganografi; dilbilim steganografi ve teknik steganografi olmak üzere ikiye ayrılır. Dilbilim steganografi metin tabanlı steganografidir. Dilbilim steganografi yöntemlerinde değişiklik yapmanın birçok yolu bulunmaktadır. Bunlardan bazıları aşağıda sıralanmıştır:

- Grafik kullanma yoluyla yapılabilir,
- Metnin yapısının değiştirilmesi yoluyla yapılabilir,
- Tek amacı gizlenecek veriyi saklamak olan yeni bir metin oluşturularak yapılabilir.

Dilbilim Steganografinin kullanmış olduğu yöntemler aşağıda belirtilmiştir [5, 6]:

– Açık kodlar: Gizlenmesi istenilen mesaj, açıkça okunabilecek normal bir metin haline gelmektedir. Mesajı gizlemek için; maskeleyme, boş Şifreler ve grid (ızgara) gibi yöntemler kullanılmaktadır.

– Şemagramlar: Gizlenmiş olan bilgi, açıkça okunan yazının küçük ama gizli bir ayrıntısının içine gizlenmektedir. Mesajın bu yöntemle gizlenebilmesi için grafiksel değişiklikler yapılmaktadır. Bu değişiklikleri yapabilmek için kullanılan metotlardan bazıları şunlardır [3, 6];

- Farklı yazı tiplerini kullanmak
- Eski daktilo yazılarını kullanmak
- İmgeler içinde boşluklar kullanmak.

Teknik Steganografi, birçok konuyu içinde barındıran çok geniş bir kavramdır. Bu konuları uygulama ortam ve yöntemlerine göre gruplandırabiliriz;

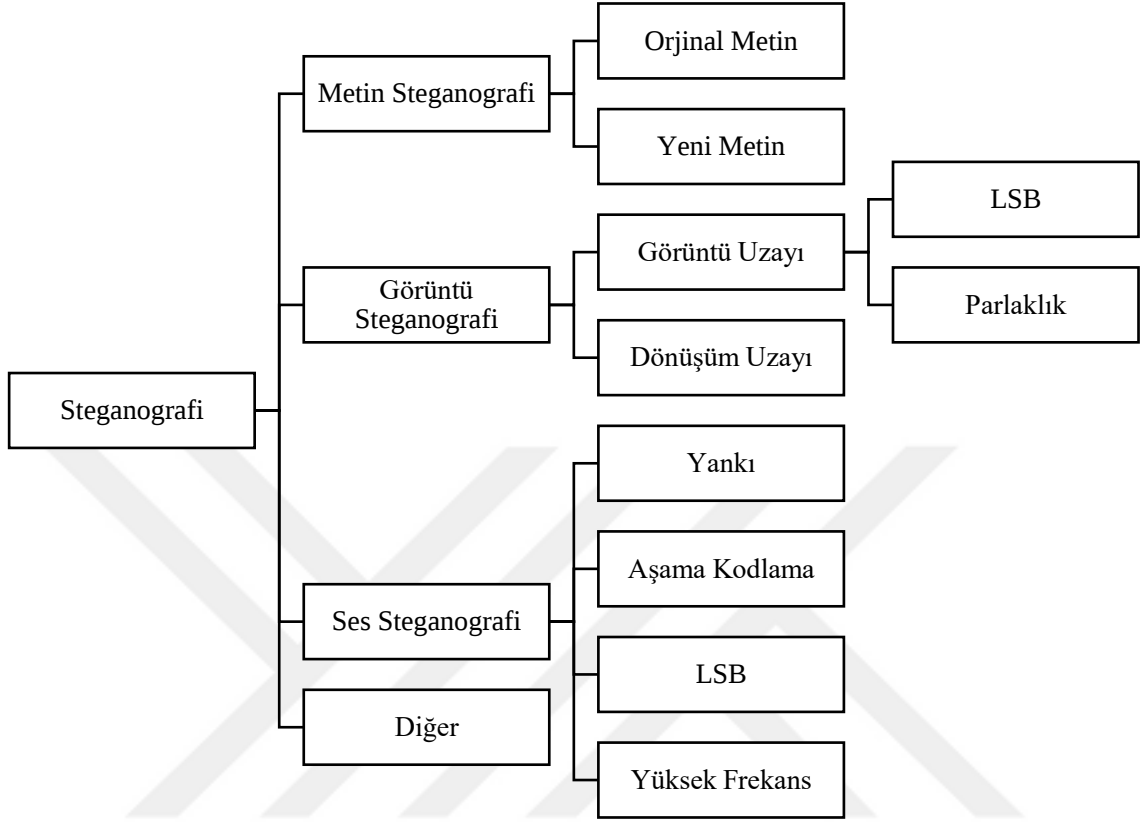
- Görünmez mürekkep: Gizlenecek olan mesajların görünmez mürekkeple yazılmasıdır.
- Gizli yerler: Mesajların kimsenin kolayca bulamayacağı gizli yerlere saklama (bavul, kasa vb.)
- Microdot'lar: Bilgiyi çok küçük noktalar halinde sayfa içerisine gizleme
- Bilgisayar tabanlı yöntemler: Metin, ses, görüntü dosyalarını kullanarak bu dosyaların içerisine mesaj gizleme yöntemleridir.

3.3. Steganografinin Kullanıldığı Alanlar

Sayısal steganografi kullanılan ortamlara göre gruplandığında genel olarak üç gruba ayrılmaktadır. Bu gruplar aşağıda listelenmiştir [4, 5, 6]:

- Metin (text) steganografi
- Ses (audio) steganografi
- Görüntü (image) steganografi

Sayısal steganografide yaygın olarak kullanılan yöntemlerin sınıflara ayrılması Şekil 3.5'te gösterilmektedir.



Şekil 3.5. Steganografi metotlarının sınıflandırılması

3.3.1. Metin Steganografi

Metin steganografi, verinin gizleneceği ortamın yazıdan oluştuğu bilgi gizleme çeşididir. Metin steganografisini uygulamak için aşağıdaki yöntemler kullanılmaktadır.

1. Açık alan yöntemleri
 - Satır Kaydırma Kodlaması
 - Kelime Kaydırma Kodlaması
 - Gelecek Kodlaması
2. Yazımsal yöntemler
3. Anlamsal Yöntemler

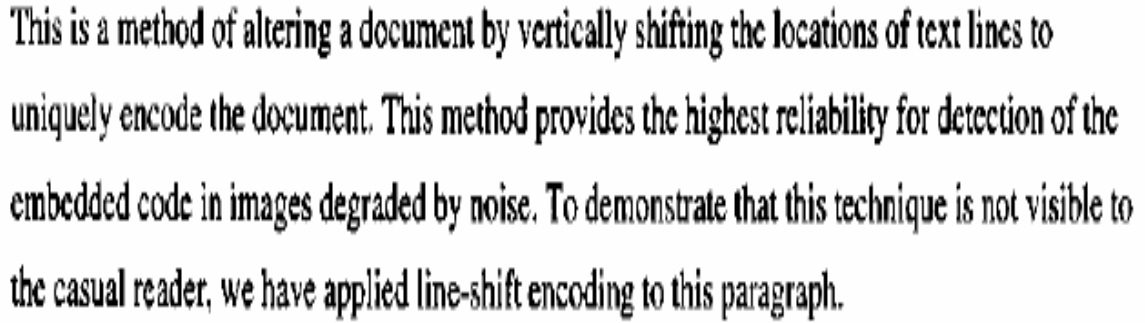
Metin steganografide genelde az miktarda veri gömülmektedir. Bunun nedeni taşıyıcı ortamın gereksiz alanları ve boşlukları az barındırmasıdır. Metin tabanlı veri gizleme yöntemlerinin hepsi, gizli mesajı tekrar elde edebilmek için ya orijinal metne ya da orijinal metnin biçimlendirme bilgisine ihtiyaç duyarlar.

3.3.1.1. Açık Alan Yöntemleri

Açık alan yöntemi, metin içerisinde normal gözüken iki kelime arasında ekstra boşluklar bırakarak ya da satır sonunda boşluklar bırakarak çalışmaktadır. Açık alan yöntemlerinin kullanmış olduğu kodlama yöntemleri aşağıda listelenmektedir [3].

3.3.1.1.1. Satır Kaydırma Kodlaması

Satır kaydırma kodlaması yönteminde yazının satırları düşey düzlemde aşağı veya yukarı ötelenerek gizlenecek olan mesajın kodlanması sağlanmış olur. Gizlenmiş olan mesaj metin dosyası ya da Windows Bitmap (BMP) dosyası olarak açılabilir. Şekil 3.6'daki metinde ikinci satır 1/300 inch yukarı doğru ötelenmiştir. Fakat çıplak gözle anlaşılabılır fark olmadığından anlaşılamamaktadır. Bu şekilde yapılan kodlama "0" ya da "1" ile tanımlanarak kodlama işlemi gerçekleştirilmiş olur [4, 5, 6].



Şekil 3.6. Satır kaydırma kodlaması örneği

3.3.1.1.2. Kelime Kaydırma Kodlaması

Kelime kaydırma kodlaması yönteminde metnin satırları yatay düzlemde sağa veya sola kaydırılarak dokümanın kodlanması sağlanır. Gizlenmiş olan mesaj tekrar metin dosyası ya da BMP formatında açılabilir. Kelime kaydırma kodlaması yönteminde birbirine yakın iki kelime arasında çıplak gözle fark edilemeyen boşluklar meydana gelmektedir. Dokümanın kodunun çözülebilmesi için boşlukların nerede olduğu bilinmesi gerekmektedir. Boşlukların nerede olduğunu bilinebilmesi için de eski dokümana ihtiyaç vardır [2, 5, 6]. Şekil 3.7'de kelime kaydırma kodlamasının nasıl yapıldığı gösterilmektedir.

Now is the time for all men/women to ...

Now is the time for all men/women to ...

(a)

Now is the time for all men/women to ...

Now is the time for all men/women to ...

(b)

Şekil 3.7. a) Satır kodlaması yapılması b) Çizgiler olmadan görünüşü

Şekil 3.7 a’da ilk satırda “for” kelimesinden önce bir boşluk bulunmaktadır, ikinci satırda **for** kelimesi ile **all** kelimesi arasında daha fazla boşluk bulunur. Şekil 3.7 b’de kelimeler arasında dikey çizgiler olmadan metnin nasıl gözüktüğü görülmektedir.

3.3.1.1.3. Gelecek Kodlaması

Gelecek kodlaması yöntemi metin dosyalarına uygulanabildiği gibi görüntü dosyalarına da uygulanabilir. Bu yöntemde bazı harflerin boyları, kelimelerin yerleri ve karakterlerin ASCII kod karşılıklarında değişiklik yapılmaktadır [4, 6]. Şekil 3.8’de gelecek kodlamasının nasıl yapıldığı gösterilmektedir.

:S AND 1 Incremental Mod

a

:S AND 1 Incremental Mod

b

:S AND 1 Incremental Mod

c

Şekil 3.8. a) Orijinal metin b) Kodlama yapılmış metin c) Abartılmış hali

Şekil 3.8 a’da orijinal metin gösterilmekte olup, Şekil 3.8 b’de gelecek kodlaması yapılmış hali gösterilmektedir. Şekil 3.8 c’de ise gelecek kodlamasının abartıldığında nasıl sonuçlar elde edileceği gösterilmektedir.

3.3.1.2. Yazımsal Yöntemler

Yazımsal yöntemlerde belgeyi kodlayabilmek için noktalama işaretleri kullanılmaktadır [17]. Örneğin aşağıda verilmiş olan iki cümleye ilk defa bakıldığında birbirinin aynısı gibi gözükmemektedir, fakat cümlelere daha dikkatli bakıldığında zaman ilk cümlelerin diğerinden farklı olarak bir ‘,’ işareti fazladan bulundurduğu görülmektedir. Bu yapıların biri “1”, diğeri de “0” olarak belirlenmekte ve kodlama işlemi bu şekilde gerçekleştirilmektedir.

“bread, butter, and milk”

“bread, butter and milk”

3.3.1.3. Anlamsal Yöntemler

W. Bender tarafından ortaya atılan bu yöntemde eşanlamlı kelimelere birincil ve ikincil değerler atanmaktadır. Daha sonra atanan bu değerler “1” ve “0” olarak binary değerlere dönüştürülmektedir.

Örneğin “big” kelimesi birincil olarak “large” kelimesi de ikincil olarak işaretlenmiş olsun. Birincil “1” olarak ikincil de “0” olarak binary’e çevrilmektedir.

3.3.2. Ses Kodlaması

Dijital seslere bilgi gizlemek dijital resimler gibi medya unsurlarına bilgi gizlemekten daha zordur. Gizli bilgi yerleştirmek amacıyla kullanılan farklı yöntemler mevcuttur. Bu yöntemler mesajı gürültü içine gizleyen basit algoritmalar olabileceği gibi karışık sinyal işleme yöntemlerini kullanarak mesajı gizleyen çok güçlü algoritmalar da olabilmektedir.

Ses içerisine veri gizlemede kullanılan yöntemlerden bazıları, LSB kodlaması, eşlik kodlaması, aşama kodlaması, tayf yayılımı, yankı veri gizlemesidir [3, 4].

3.3.2.1. En Az Ağırlıklı Bit Kodlama

Dijital ses dosyaları içerisine mesaj gizlemenin en kolay yoludur. LSB kodlama en önemsiz bitlerin örnek noktalarının her birine bir tane binary verinin gizlenmesiyle oluşmaktadır. LSB kodlama, yapılabilecek en fazla miktarda mesajın gizlenmesine imkân vermektedir. LSB kodlama için en ideal bilgi iletim oranı 1Khz da 1Kbps’dır [6].

3.3.2.2. Eşlik Biti Kodlama

Bir sinyali bireysel örneklerle değiştirmek yerine, bu sinyali örnekleme bölgelerine ayırır ve daha sonra gizlenecek olan verinin her bitini benzer bölgenin eşlik bitine gizler [6]. Seçilen bölgenin eşlik biti gizlenecek olan mesajın biti ile eşleşmiyor ise seçilen bölgenin

içinde LSB kodlama yöntemi kullanılır. Bu yöntem sayesinde mesajı gizleyerek gönderen kişi gizli biti eklemede daha fazla imkânlarla sahip olmaktadır. Ayrıca sinyal de daha az dikkat çekici bir hale de gelmektedir.

LSB veya eşlik biti kodlama yöntemini kullanmanın temel iki dezavantajı vardır. Bunlardan ilki insan kulağının çok duyarlı olmasından dolayı ses dosyası üzerindeki oluşabilecek en küçük bozulmayı bile saptayabilmesidir. Yine de eşlik biti kodlama LSB kodlamayla karşılaştırıldığında ses dosyası üzerinde daha az bozulma meydana getirmektedir. Bu yöntemlerin ikinci dezavantajı ise her iki kodlama yönteminin de sağlam ve güçlü bir kodlama seçeneği olmamasıdır. Eğer eşlik biti veya LSB kodlama yöntemiyle kodlanmış gizli bir veri içeren ses dosyası tekrar oluşturulursa, gizlenmiş olan mesaj kaybolacaktır. Farklı yöntemler kullanılarak bozulmalara karşı daha güçlü bir kodlama oluşturulabilir fakat bu kodlamayla birlikte gizlenecek olan veri miktarında da önemli miktarda azalma olacaktır [3, 4, 5].

3.3.2.3. Aşama Kodlaması

Aşama kodlaması yöntemi görüntü dosyalarında uygulanmakta olan JPEG algoritmasına benzer bir yapıyla oluşmaktadır [4, 5, 6]. Gizleme işlemi esnasında ses dosyası küçük parçalara ayrılır ve her parçaya ait faz, gizlenecek olan mesaja ait faz referansı ile yer değiştirilir. Aşama Kodlaması yöntemi insan kulağının duyamayacağı sesler üzerinde veri gizlemeyi yapmayı amaçlamaktadır. Aşama kodlamanın sahip olduğu tek dezavantaj, düşük mesaj gizleme oranına sahip olmasıdır. Bu dezavantaj bölünen her parçanın uzunluğu artırılarak giderilebilir. Fakat bu yöntem kullanıldığı zaman gizlenen mesajın tespiti de kolaylaşmış olur. Aşama kodlaması yöntemi sonuç olarak maskeleye gibi küçük bilgilerin gizlenerek iletilmesinde kullanılır.

3.3.2.4. Tayf Yayılması

Ses Stenografisinde tayf yayılması yöntemi, gizli mesajı, ses sinyalinin frekans spektrumuna mümkün olduğu kadar homojen olarak yaymaktır. Bu yöntem, gizlenecek mesajın bitlerini ses dosyasının tamamına rastgele yayan ve bu yayma işlemini LSB kodlama yöntemini kullanarak yapan sistemin analog biçimidir [4, 5]. Bu yöntemin LSB kodlama yönteminden farkı, tayf yayılması yöntemi gizlenecek bilgiyi ses dosyasının frekans spektrumuna yayar ve orijinal sinyalden bağımsız olan bir kod kullanır. Gizleme işlemi sonucunda oluşan sinyal, gerekli olan bant genişliğinden daha fazla bant genişliğine yol açmaktadır. Ses dosyasına veri gizlerken iki tip tayf yayılması kullanılır [4]. Bunlar:

- Düz Sıralı Dağınık Spektrum (Direct-Sequence Spread Spectrum)

- Frekans Atlamalı Dağınık Spektrum (Frequency-Hopping Spread Spectrum)

Düz Sıralı Dağınık Spektrum (Direct- Sequence Spread Spectrum)

Gizlenecek olan her bit için çok miktarda bitlerden oluşan bir kalıp üretilir. Bu bit kalıbına “chip” ya da “chipping code” adı verilmektedir. “Chip” ne kadar uzun olursa orijinal bilginin tekrar elde edilmesi o kadar yüksek olur. Fakat bu yöntem gerekenden daha fazla bir bant genişliği gerektirir. Eğer Chip’in içindeki bir veya iki bit haberleşme esnasında bozulur ise cihaz istatistiksel tekniklerle orijinal bilgiyi haberleşmeyi tekrarlamadan kurtarabilir.

Frekans Atlamalı Dağınık Spektrum (Frequency-Hopping Spread Spectrum)

Ses dosyasındaki frekans spektrumu değiştirilir ve bu yöntemle gizlenecek olan mesajın bitleri frekanslar arasında sekerek gizlenir. Tayf yayılması yöntemi bilgi iletim oranı ayarlama ve silme tekniklerine karşı sağlamlık açısından daha iyi performans göstermesi açısından LSB kodlama, eşlik biti kodlama ve aşama kodlama tekniklerinden daha iyi sonuçlar verebilmektedir. Ancak Tayf yayılması yöntemi ses dosyasında oluşturduğu gürültü ile LSB ve eşlik biti kodlama yöntemiyle aynı sorunu yaşamaktadır.

3.3.2.5. Yankı Veri Gizleme

Yankı veri gizleme yöntemi, insan kulağının ses dosyası üzerindeki kısa süreli yankıları algılayamaması özelliğini kullanmaktadır [4]. Bu yöntemde gizlenecek bilgiler ayrı bir sinyal içerisinde yankı olarak oluşturulur ve bu yankı ses dosyasına gömülür. Tayf yayılması yöntemi gibi, Yankı veri gizleme yöntemi de yüksek miktarda veri iletim oranına imkân sağlar. Gürültüye neden olan yöntemlerle kıyaslandığında üstün sağlamlık sağlar. Bilgiyi başarıyla saklayabilmek için genlik, düşüş oranı, orijinal sinyalden sapması(gecikme süresi) gibi yankının üç parametresi değiştirilir. Bu üç parametre de insan kulağının algılama eşiğinin altında ayarlanır.

3.3.3. Görüntü Steganografi

Bilgi gizlenecek ortamın resim olduğu steganografi çeşididir. Görüntü steganografi çeşitleri şunlardır:

1. En Önemsiz Bite Ekleme
2. Maskeleye ve Filtreleme
3. Algoritmalar ve Dönüşümler

3.3.3.1. En Önemsiz Bite Ekleme

En önemsiz bite ekleme yöntemi görüntü steganografisinde en çok kullanılan yöntemdir. LSB yönteminde gizlenecek olan mesajın her bir biti, resim dosyasını oluşturan

verinin bir baytının en son bitine yazılır. Örnek olarak 'A' harfini 24 bit BMP resim dosyasına gizlemek için 8 byte gerekmektedir. Gizlenecek 'A' harfinin ikilik sayı sistemindeki karşılığı bu 8 byte'lık verinin son bitlerine gizlenecektir. Veri gizlemek için gerekli olan sekiz biti elde edebilmek için üç piksel seçmeliyiz. Bu işlem adımları aşağıdaki gösterilmektedir [18].

Bilginin gömüleceği üç pikselin içerdiği değerler ve bunların ikilik sistemdeki karşılıkları:

(00100111 11101001 11001000)(39 233 200)

(00100111 11001000 11101001)(39 200 233)

(11001000 00100111 11101001)(200 233 39)

Gizlenecek bilgi olan 'A' harfinin ikilik sistemdeki değeri:

(1 0 0 0 0 1 1)

Piksel değerlerinin veri gömüldükten sonraki durumu:

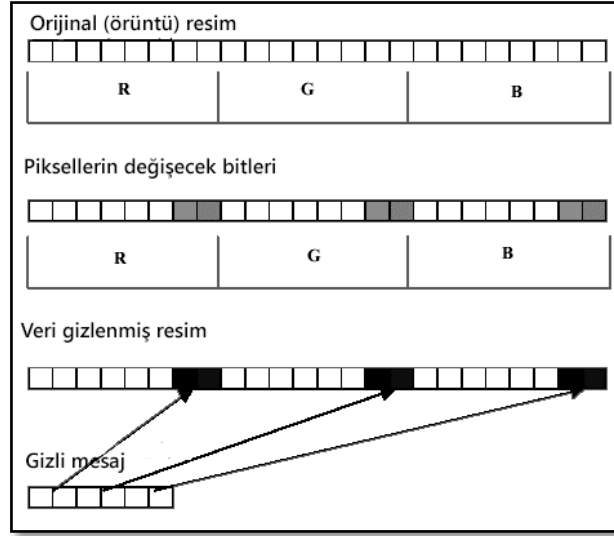
(00100111 11101000 **11001000**) (39 232 200)

(0010011**0** 11001000 1110100**0**) (38 232 232)

(11001000 00100111 11101001) (200 233 39)

Örnekte olduğu gibi gizleme işleminde altı çizili olan bit dizisi kullanılmamıştır. Veri gizlenirken ilk bayttan başlayarak son bayta kadar devam eden doğrusal bir sıralama izleme zorunluluğunun olmadığı buradan görülebilmektedir. Verinin gizlenmesi işleminde kullanılacak olan baytların sırası stego-key denilen bir anahtar aracılığıyla düzenlenmekte ve tamamen karışık sırayla veri gizlenmesi yapılabilmektedir.

Bilginin gizlenmesinden sonra değerler incelendiğinde bitlerin eklendiği byte dizisinden sadece üçünün son bitinde değişiklik olduğu görülmektedir. Son bitlere gizleme yapıldığında ortam verisindeki değişme ihtimali %50 olmaktadır. Bu prensipten yola çıkarak sadece son bitlere ekleme yapmak yerine son iki bite ekleme yapmak bir resimde saklanabilecek veri miktarını iki katına çıkarmaktadır. Son iki bite ekleme işlemi gerçekleştirildiğinde, 24 bit renk derinliğine sahip BMP formatlı bir resim dosyasında bu fark hala gözün fark edemeyeceği düzeyde olmaktadır. Şekil 3.9.'da son iki bite ekleme yönteminin örneği görülmektedir.



Şekil 3.9. Son iki bite veri gizleme işlemi

24 bit BMP, 8 bit BMP, GIF ve gri tonlamalı resimlerde LSB ekleme yöntemi uygulanabilir. Her formatın kendisine özgü avantaj ve dezavantajları bulunmakla birlikte daha önce de belirtildiği gibi en uygun format 256 renk gri tonlamalı resimlerdir [3].

Kullanabileceğimiz formatları tek tek değerlendirmek gerekirse 24 bit BMP formatındaki resimler LSB yöntemi için en uygun yapıdaki ve en geniş alanı sağlayan resimlerdir. Kullandığı geniş renk spectrumu (16 milyon renk) sayesinde gömme işleminin yaratacağı değişimler fark edilmemektedir. Ancak bu formattaki resimler çok büyük alan kaplamaktadır ve dağıtımını diğerlerine göre daha zor olmaktadır.

Kayıpsız sıkıştırma kullanılan 8 bit BMP ve GIF formatları her piksel için 1 bayt kullanarak dosya boyutunu küçültmektedir. Kullandıkları palet ise gömme işlemini zorlaştırmaktadır. Paletteki renk sıralaması nedeniyle piksel değerindeki bir artış veya azalış renkte büyük farklılıklara neden olabilmektedir. Steganography yazılımları bu sorunu aşmak için renk farklılıklarını mümkün olduğu kadar azaltmak üzere paleti düzenlemektedirler.

3.3.3.2. Maskeleye ve Filtreleme

Genellikle 24 bit ve gri-seviye görüntü dosyaları üzerinde işaretleme (marking) ve filigran yapılarak uygulanmakta olan tekniğe Maskeleye ve filtreleme teknikleri denilmektedir [5, 6]. Filigran teknik olarak bir steganografi biçimi değildir. İşaretleme ya da filigran tekniklerinin görüntülere sıkça uygulanması nedeniyle, görüntü üzerindeki değişikliğin fark edilmesi korkusu olmadan uygulanabilmektedir. Maskeleye işlemi, son

bite ekleme yöntemine göre sıkıştırma, kırpma ve bazı görüntü işlemleri açısından daha güçlüdür.

3.3.3.3. Algoritmalar ve Dönüşümler

En önemsiz bite ekleme yönteminde bilgiyi gizlemek oldukça kolay ve hızlı bir yöntemdir. Fakat görüntü dosyasına uygulanan işlemler (kırpma, döndürme, v.b) ya da uygulanan kayıplı sıkıştırmalar esnasında bilgi zarar görebilmektedir. Yüksek kalitedeki resim dosyalarının internet ortamında sıkıştırılarak, örneğin JPEG formatı kullanılarak gönderilmesi daha uygun olacaktır. Bu sebepten dolayı gizlenen mesajın kaybolmaması ve görüntünün sıkıştırılmasını sağlayan algoritmalar ve dönüşümler kullanılmaktadır [3].

3.4. Steganografi Alanında Yapılan Çalışmalar

Steganografi alanında birçok çalışma yapılmıştır. Bu çalışmalara ait literatür taraması yapıp yapılan çalışmalara aşağıda yer verilmiştir.

Bender 1996 da yaptığı çalışmasında resim, ses ve metin içerisine veri saklamayı detaylı bir şekilde anlatmıştır. Ses içerisine, düşük bit kodlaması, faz kodlaması, yayılmış spektrum ve yankı veri saklaması yöntemleriyle veri saklanması bu çalışmada incelenmiştir. Ayrıca boşluk kullanımı, konuşma dilinin yapısı ve eş anlamlı kelimelerden faydalanarak metin içerisine veri saklama yöntemleri bu çalışmada ayrıntılı bir şekilde incelenmiştir [5, 17].

Natthawut 2009 yılında kaleme aldığı çalışmasında cep telefonlarından kısa mesajlar (SMS) içerisine bilginin nasıl gizlendiğine dair detaylı bir açıklama yapmıştır [19]. Yapılan çalışmada Thai dili kullanılarak sms içerisine bilgi gizlemeye çalışılmıştır. Thai dili kullanılmasının sebebi olarak bu dilde yazılan mesaj dönüştürüldüğünde daha çok bit barındırmasını öne sürmüştür. Bu sayede kısa mesaj içerisine bilgi gizlenebildiği sonucuna varmıştır.

Alwan ve arkadaşlarının 2008 yılında resim steganografide piksellerin nasıl seçildiğine yönelik uygulamalar yapmışlardır [20]. Bu çalışmada kenar bulma ve kenar piksel bağıntısı kullanılmıştır. Bu bağıntı kullanılarak bilgi gizlenecek pikselin seçimi detaylı olarak anlatılmıştır.

Fridrich ve arkadaşları ele aldıkları makalelerinde gri seviye ve renkli resimlerde LSB ile yapılmış steganografinin sezilmesini detaylı bir şekilde anlatmışlar ve uygulama yapmışlardır [21]. Yapılan çalışmada Cover image, Steganos, Hide4PGP gibi programlarla veri gizlenen resimlerden, RS (Rescaled Range Analysis) analiz tekniği kullanarak gizlenen bilgiyi elde etmeye çalışmışlardır.

Cheddad ve arkadaşları 2010 yılında ele aldıkları makalelerinde steganografi yöntemlerini ve steganografik yazılımlarını incelemişlerdir [2]. Çalışmada steganografik yazılımların veri gizlerken hangi yöntemleri kullandıkları, avantajları ve dezavantajlarını detaylı bir şekilde anlatmışlardır.

Sui ve Luo (2004), internet sayfasındaki metinler içerisine, boşluklar ekleme yerine biçimleme etiketlerinin (HTML tags) pozisyonlarını değiştirerek veri saklayan bir çalışma yapmışlardır [22]. Bu çalışmadan elde ettikleri sonuçlara bakarak, uyguladıkları yöntemin şu an için, resim ve ses içerisine veri saklama yöntemlerine göre henüz zayıf olsa da, ilerleyen zaman içerisinde bilgi güvenliği sistemleri açısından potansiyel değerinin olduğunu bildirmişlerdir [22].

Anderson ve arkadaşları 1998 yılında ele aldıkları makalelerinde bilgi gizlemek için kullanılan dosyaların ne kadar bilgi gizlenebileceğine dair çalışma yapmışlardır [12]. Yapılan çalışmada bilgi gizlenirken teorik olarak ve pratikte bilginin limitini belirlemeye çalışmışlardır.

Rahul Joshi ve arkadaşları 2013 yılında yazdıkları makalelerinde resim steganografide LSB yöntemini ele almışlardır [23]. Bu çalışma sonucunda, key oluşturarak LSB yöntemiyle resim dosyasının içine bilgi gizlenmiştir.

Lisa M. Marvel ve arkadaşları 1999 yılında yazdıkları makalelerinde yayılmış spectrum tekniğiyle resim steganografi yapmışlardır [24]. Bu çalışmada mesajın resim, müzik, resim veya herhangi bir sayısal sinyale gizlenebileceğini belirtmişlerdir. Bu çalışma sonucunda, bu tekniğin ataklara karşı daha dayanıklı olduğu, gizlenen bilginin elde edilirken orijinal resime gerek olmadığı rapor etmişlerdir.

Rukiye Karakış ve İnan Güler 2014 yılında yaptıkları çalışmalarında medikal dicom görüntüler için steganografi uygulaması yapmışlardır [25]. Aynı yıl ayrıca bulanık mantık tabanlı steganografi çalışması da yapmışlardır [26]. Bu çalışmalar sonucunda, hastaların kişisel bilgileri ve doktorun yapmış olduğu teşhis, uyguladığı tedavi hastaya ait görüntü içine gizlenerek gerekli gizliliğin sağlanabileceği rapor edilmiştir.

Zekeriya Şentürk ve arkadaşları 2015 yılında LSB yöntemiyle renk değişimi esaslı yeni bir steganografik yaklaşım önermişlerdir [27]. Bu çalışmada ilk 7 bitin değişmediği komşu pixseller belirlenerek bu piksellerin son bitine mesajın bitleri sırasıyla yerleştirilmektedir. Çalışma sonucunda, renklerin değişmediği bölgelere bilgi gizlenerek, gizlenmiş bilginin algılanabilirliğinin düşürüldüğü rapor edilmiştir.

Erdal Güvenoğlu ve E. Murat Esin yazdıkları makalelerinde LSB yöntemini SUFFLE algoritmasıyla iyileştirmeye çalışmışlardır [28]. Bu çalışmada, suffle algoritması ve anahtar yöntemi kullanmışlardır. Çalışmada sonuç olarak, kullanılan yöntemle bilginin elde edilmesi için anahtarın gerekli olduğunu ve bu yöntemle mesaj gizlenen resimde gözle görülür bir fark olmadığını, piksel çakışması bulunmadığını belirtmişlerdir.

Yıldıray Yalman ve arkadaşları 2012 yılında ayırık kosinüs dönüşümlü steganografik yaklaşımının görsel ataklara karşı dayanıklılığını araştıran bir makale yayınlamışlardır [29]. Bu çalışmada, ayırık kosinüs dönüşüm bloklarına ait katsayıların veri gizleme için kullanılması ve ikili damganın her bir bitinin ayrı bir bloğa gömülmesi ile ataklar sonucunda gizli bilgide oluşan bozulmanın minimize edile bildiği rapor edilmektedir.

Engin Avcı ve arkadaşları 2011 yılında yazdıkları makalelerinde renkli imgelerin görsel ataklara karşı dayanıklılığını inceleyen bir uygulama sunmuşlardır [30]. Bu çalışmada sıralı LSB yerine, resmin tüm boyutlarına ve rastgele bilgi gizlendiği bu sayede bilginin elde edilmesi daha zor hale getirildiği belirtilmiştir. Çalışma sonucunda, yapılan görsel ataklara karşı daha dayanıklı olduğu ve benzer uygulamalara oranla daha yüksek PSNR değerlerine sahip olduğu rapor edilmiştir.

Meltem Kurt ve Nevcihan Duru 2014 yılında yazdıkları makalelerinde menezes vanstone eliptik eğri şifreleme algoritması kullanılarak video dosyaları üzerinde steganografi uygulaması geliştirmişlerdir [31]. Bu çalışmada, menezes vanstone algoritmasını ve eliptik eğri şifreleme algoritmasını kullanmışlardır. Kullanılan menezes vanstone algoritmasıyla bilgiyi gizleyecekleri video çerçevesini belirlemişlerdir. Sonuç olarak, kullanılan şifreleme yöntemi ve kriptografi yöntemiyle güvenli bir şekilde bilgiyi gizlediklerini belirtmişlerdir.

Meltem Kurt Pehlivanoğlu ve arkadaşları 2015 yılında yazdıkları makalelerinde koblitz metodu kullanılarak video dosyaları üzerinde lsb tabanlı steganografi uygulaması geliştirmişlerdir [32]. Bu çalışmalarında, koblitz metodunu kullanarak bilgi gizleyecekleri video çerçevesini belirlemişlerdir. Uygulanan yöntemle yapılan deneysel sonuçlarda PSNR değerleri 30 db den yüksek olduğu için videoda gözle görülür bir fark olmadığı rapor edilmiştir.

Yasemin Yıldız ve Ahmet T. Özcerit 2014 yılında yaptıkları çalışmalarında 24 bit renkli hareketli resimler (video) üzerinde veri gizleme çalışmalar yapmışlardır. Yaptıkları bu çalışmalarında AVI formatındaki video dosyalarının üzerine şifrelenen mesaj klasik veri gömme tekniği olan LSB'den farklı olarak RGB ağırlık tabanlı veri gizleme ile

gerçekleştirmişlerdir. LSB kodlamaya oranla nispeten daha fazla veri gizleyebildiklerini ifade etmişlerdir [33].

Andaç Şahin ve arkadaşları yaptıkları çalışmalarında renkli resimlerde veri gizlenip gizlenmediğini anlamak için RS steganaliz uygulaması sunmuşlardır. Yaptıkları çalışmada steganalizde maske seçiminin öneminden bahsetmişlerdir. Maske seçimi doğru olmadığı takdirde, örüntü dosyasının içerisinde veri olmadığı halde veri olduğunu gösterdiğini ifade etmişlerdir. Seçilen maskeye göre steganaliz sonuçlarının başarımlarını da değiştirmekte olduğunu bildirmişlerdir [34].

Mohammad-Saleh Nambakhsh, ve arkadaşları 2011 yılında yaptıkları çalışmalarında elektrokardiyograf (EKG) ve demografik metin verilerini çift filigran olarak kullanan yeni bir veri gizleme yöntemi sunmaktadırlar. Bu verileri yüksek çözünürlüklü dalgalı ayırması kullanılarak bir PET görüntüsünün seçilmiş doku bölgelerine gömüldüğünü ifade etmişlerdir. Yaptıkları çalışmada hasta bilgilerinin korunduğunu ve eşleşmeyen teşhis bilgilerini engellediğini bildirmişlerdir [35].

Şengül Doğan 2016 yılında yaptığı çalışmada kaotik haritaları ve genetik algoritmayı kullanarak yeni bir veri gizleme yöntemi sunmuştur. Bu çalışmada genetik algoritmayı geliştirmek için kaotik haritalardan faydalanılmıştır. Yapılan çalışmada kullanılan yöntemin gauss, logistic ve tent maps rasgele fonksiyonlarından daha hızlı olduğunun gözlemlendiğini bildirmiştir [36].

Cheng-Hsing Yang ve arkadaşları yaptıkları çalışmalarında PVD (Piksel Değer Farkı) kullanan daha fazla veri gömebilme imkânı sunan ve algılanamayan görüntü steganografi yöntemi sunduklarını bildirmişlerdir. Yöntem, iki piksele kaç tane gizli bit ekleneceğini tahmin etmek için ardışık iki piksel arasındaki fark değerini kullanmaktadır. Fark değerleri aralığı, adaptif olarak daha düşük, orta ve daha yüksek seviyelere ayrılmıştır. Birbirini izleyen herhangi bir çift piksel için, her iki piksel de -bit LSB değiştirme yöntemiyle yerleştirilir. Wu ve ark.'nın PVD ve LSB değiştirme yöntemine ilişkin geçmiş çalışmalarıyla karşılaştırıldığında, deney sonuçlarının, önerdikleri yöntemin hem daha büyük yerleştirme kapasitesi hem de daha yüksek görüntü kalitesi sağladığını gösterdiğini bildirmişlerdir [37].

Padmapriya Praveenkumar ve arkadaşları 2016 yılında yaptıkları çalışmalarında kuantum polarizasyonlarını ve Rubik küpleri şifrelemesini kullanarak görüntünün belirli bloğunu kaydırarak çoklu şifreleme sağlamış ve bunu medikal verileri dicom görüntülere gizlemede kullanmışlardır. Yaptıkları çalışmada elde edilen sonuçları, Novel hybrid image

compression–encryption algorithm based on compressive sensing ve Novel image compression–encryption hybrid algorithm based On key controlled measurement matrix in compressive sensing çalışmalarıyla karşılaştırıldığında daha iyi sonuçlar elde ettiklerini bildirmişlerdir [38].

Andaç Şahin Mesut ve arkadaşları yaptıkları çalışmalarında Bu çalışmada, görüntü paylaşım yöntemlerinde gizli paylaşım şemalarının nasıl kullanıldığını ve bu yapının güvenliğini nasıl etkilediğini incelemiştir. Yapılan çalışma sonucunda elde edilen veriler paylaşım şemalarının kullanılmasının verinin güvenliğini artırdığını ve yetkisiz kişiler tarafından elde edilmesinin zorlaştırıldığını rapor etmişlerdir [39].

Andaç Şahin ve arkadaşları yaptıkları çalışmalarında sayı teorisi ile rastgele LSB kodlama yöntemini birleştirip gri seviyeli resimlerde veri gizlemek için bir yöntem önermişlerdir. Yaptıkları çalışmalarında rastgele LSB yöntemini kullanmışlardır. Kullandıkları bu yöntem sayesinde klasik sıralı LSB yöntemine göre daha dayanıklı olduğunu ifade etmişlerdir. Rastgele yerleşimi sağlamak için ise ayrık logaritma fonksiyonu kullandıklarını ifade etmişlerdir [40].

Sefa Tunçer ve Cihan Karakuzu 2016 yılında yaptıkları çalışmalarında Gizlenecek veriyi önce RSA algoritmasıyla şifrelemiş daha sonra görüntü dosyasının son üç bitine veriyi gizlemişlerdir. RSA algoritmasının kullanılmasının en önemli sebebi olarak veri güvenliği olduğunu bildirmişlerdir. Kullanılan steganografi ve kriptografi yöntemleri sayesinde verinin güvenli bir şekilde saklanabilir veya iletilebilir hale geldiğinin görüldüğü rapor edilmiştir [41].

Ömer Kurtuldu ve Nafiz Arıca yaptıkları çalışmada imgeleri karelere bölmüşler ve bu karelere veriyi gizlemeye çalışmışlardır. Önerdikleri yöntem, örtü imgesini karelere bölerek her kareye kare boyutuna bağlı uzunlukta mesaj bitini saklamaktadır. Kare içinde değişik yönlerdeki satır/sütunlardaki piksellerin en önemsiz bitlerinde (EÖB) arama yaparak mesaj bit dizisine en yakın satır/sütun bulunmakta ve bulunan satır/sütunun en önemsiz bitleri mesaj bit dizisiyle değiştirildikten sonra mesajın geri elde edilebilmesi amacıyla değiştirilen satır/sütunun çerçeve bitinin işaretlendiğini bildirmişlerdir. Yapılan çalışmada elde edilen sonuçların literatürdeki çalışmalara oranla imge üzerinde daha az bozulmaya sahip olduğunu bildirmişlerdir [42].

Sunil Kumar 2014’de yaptığı çalışmasında LSB yer değiştirme tekniğini kullanarak medikal görüntülerde veri gizleme işlemi yapmıştır. Yaptığı çalışmada elde edilen sonuçları iki ölçüte göre değerlendirmiştir. Bu ölçütlerden birincisi fark edilebilirlik diğeri

ise dayanıklılıktır. Yapılan çalışmada elde edilen sonuçlara bakılarak önerilen yöntemin başarılı olduğunu rapor etmiştir [43].

Arash Habibi Lashkari ve arkadaşları 2011 yılında yaptıkları çalışmalarında görüntü steganografisinde kullanılan teknikleri incelemişler ve sonuçları tablolar halinde sunarak karşılaştırma yapmışlardır. Patchwork yaklaşımının tablolandırılmasından, çoğu saldırı türüne karşı çok yüksek bir sağlamlığa sahip olduğu, ancak sadece çok az miktarda bilgiyi gizleyebildiğini bildirmişlerdir. Hem BMP hem de GIF'deki LSB, bunun üstesinden gelebilmesine rağmen, her iki durumda da sonuç, üçüncü bir kişi tarafından tespit edilmeye kolayca yatkın olan bir yöntem olduğunu ifade etmişlerdir [44].

Adnan Abdul-Aziz Gutub ve arkadaşları 2010 yılında yaptıkları çalışmalarında Arapçanın işaretlerinden faydalanarak metin içerisine veri gizlemişlerdir. Çalışmalarında iki adet algoritma sunmuşlardır. Yaptıkları çalışmalarında önerdikleri her iki algoritmayı da gerek kapasite gerekse sağlamlık ve dosya boyutu ölçümleri açısından ayrıntılı bir performans analizi yaparak her iki algoritmayı avantajları ve dezavantajları olarak detaylı şekilde incelemişlerdir [45].

Fabien A. P. Petitcolas ve arkadaşları 1999 yılında veri gizleme üzerine araştırma çalışması yapmışlardır. Bu çalışmada veri gizleme yöntemleriyle ilgili yeni ve eski yöntemleri araştırmışlar ve aralarındaki bağlantıyı ortaya koymaya çalışmışlardır. Ayrıca alt algısal bozulma ekleyerek birçok markalamayı kıran ve eko gizlemeye özel bir saldırı tanımladıkları bir araç geliştirmişlerdir. Yaptıkları çalışmanın sonucunda genel bir araçla tüm markalamanın kırılmayacağını bildirmişlerdir [46].

Ersin Esen ve İlkay Atıl yaptıkları çalışmalarında standart görüntü işleme işlemlerinin steganaliz sonuçlarına etkisini araştırmışlardır. Yaptıkları çalışmada oluşturulan 1000 imgelik veri kümeleri ile dokuz farklı steganaliz özneliği incelenmiş, sonuçlar tablolar ile aktarmışlardır. Elde ettikleri sonuçlara göre görüntü işleme yöntemlerinin büyük ölçüde steganaliz sonuçlarını etkilediklerini ve her türlü işleme dayanıklı (yanlış alarm vermeyecek) bir eğitim kümesinin oluşturulamayacağı sonucunu rapor etmişlerdir [47].

Emir ÖZTÜRK ve arkadaşları yaptıkları çalışmalarında en çok kullanılan yöntemlerden biri olan LSB yöntemini BMP resminin tüm kanallarını kullanmayıp sadece seçilen tek bir renk kanalına uygulayarak güvenilirliğe etkisini araştırmışlardır. Bu durumun güvenliği negatif yönde etkilememekle birlikte saldırganın işini daha zorlaştırmak amacıyla kolaylıkla uygulanabilir olduğunu ve saldırganın elinde orijinal resim olmadığı için bilgiyi sezme ve elde etmek için daha fazla çaba sarf etmesi gerektiğini rapor etmişlerdir. Ayrıca

bilgi gizleme işleminin sıralı değil de bir anahtar değere göre rastgele yapılması steganalitik saldırılara karşı daha güçlü olmasını sağlayacağını bildirmişlerdir [48].

Ferdi Doğan ve arkadaşları yaptıkları çalışmalarında hastaların mahrem bilgilerinin gizli kalması istenen durumlarda, yeni bir veri gizleme/gömme yaklaşımı ortaya koymaktadır. Ayrıca yapılan çalışmada kişisel bilgilerin hafızada büyük alan kaplayarak görüntünün orijinalliğinin bozulmaması için geliştirilen veri sıkıştırma tekniği de veri gizlemeyle birlikte kullanılmıştır. Yaptıkları deneysel çalışmalar sonucunda, güvenliği sağlamak için medikal görüntüler üzerinde yeni bir sır örtme yaklaşımı sunulurken, yüksek veri kapasitesine sahip gizli bilgilerin ise sıkıştırılarak arşivlenmesi önerilmiştir. [49].

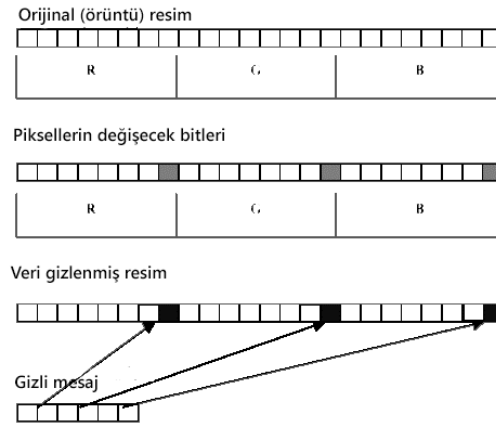
Rukiye Karakış ve arkadaşları medikal görüntüler içerisine bilgi saklayabilmek için bulanık mantık tabanlı steganografi yöntemini kullanmışlardır. Yaptıkları çalışmada önerdikleri yöntemin görüntü piksellerinin en az ağırlıklı bitlerinin (LSB) rastgele seçilmesini sağlamakta olduğunu bildirmişlerdir. Hastaya ait kişisel bilgiler ve doktor yorumundan oluşturulan mesaj, olası saldırılara karşı şifreleme ve sıkıştırma yöntemleri ile dayanıklı hale getirilmekte olduğunu ifade etmişlerdir. Geliştirilen uygulamayla veriyi 128 bitlik bir anahtarla şifrelediklerini bu sayede hastaya ait gizli bilgilerin sadece uzmanlar tarafından görülebildiğini rapor etmişlerdir [50].

Petros L. K. Mantos ve Ilias Maglogiannis 2016 yılında yaptıkları çalışmalarında ROI tablolarını kullanarak hastalarının hassas bilgilerini gizlemeye çalışan bir steganografi uygulaması yapmışlardır. Deneysel sonuçlar, önerilen düzenin sınırlı düz alanlar içeren resimler kullanıldığında verimli bir steganografi şeması olarak kullanılabileceğini kanıtladığını ifade etmişlerdir. Sunulan yöntemin, Kapasite, Sağlamlık, Algılanamazlık ve Mahremiyet ile ilgili değerlendirmenin tüm yönlerinde çok iyi sonuçlar verdiğini ayrıca daha fazla yürütme süresi ve veri genel giderleri düşük olduğunu ve mevcut dağıtılmış tıbbi bilgi sistemlerinde kolay ve şeffaf uyuma olanak tanıdığını bildirmişlerdir [51].

P.Thiyagarajann ve Aghila G, grafik renklendirmeyi kullanarak medikal görüntüler için veri gizleme uygulaması yapmışlardır. Bu çalışmalarında, grafik renklendirme problemi ile oluşturulan dinamik bir anahtar kullanılarak tıbbi bir görüntü içerisinde hasta bilgilerinin gizlendiği yeni bir steganografi tekniği önerilmektedir. Deneysel sonuçlar, önerilen yöntemin kırpma ve döndürme gibi tekdüze affine dönüşümlerine karşı dirençli olduğunu gösterdiğini bildirmişlerdir. Önerilen yöntem, iletilen tıbbi görüntüleri kirletebilecek iletim hatası ile ilgili davalar göz önüne alınarak tasarlanmakta olduğunu rapor etmişlerdir [52].

4. MATERYAL VE METOT

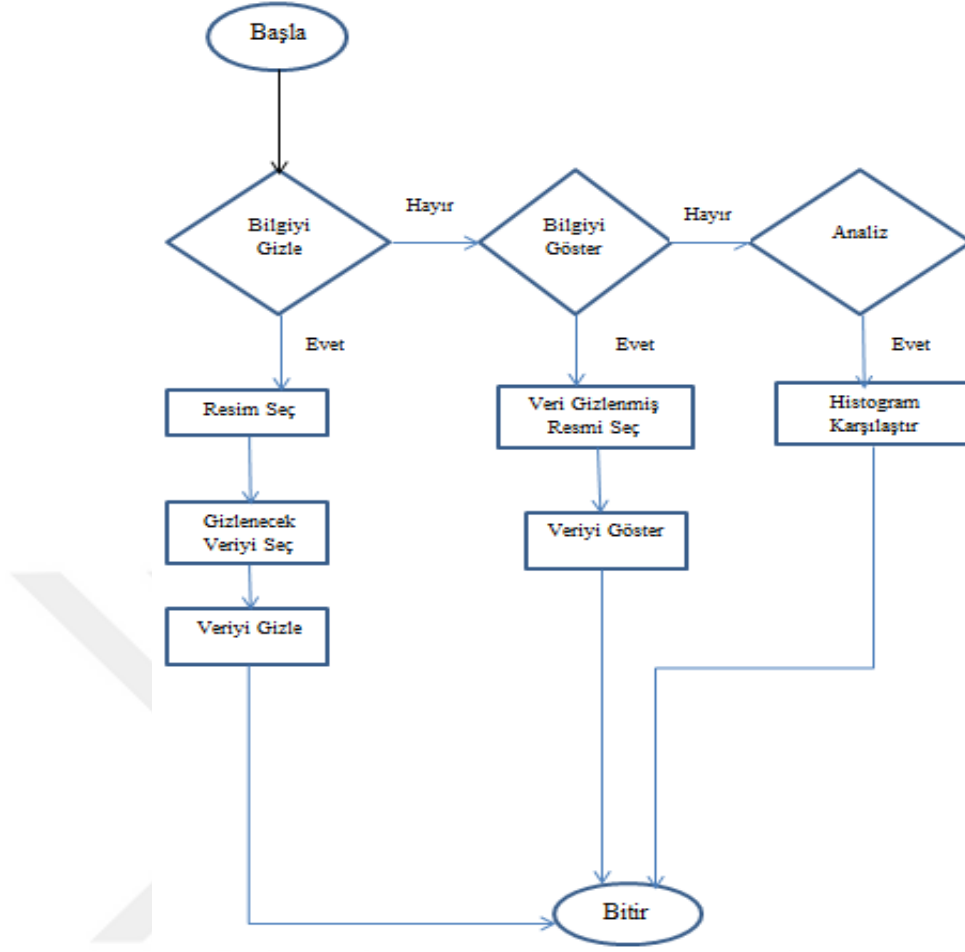
Steganografi bir verinin başka bir veri içine gömülerek gizlenmesidir. LSB yöntemi görüntü steganografisinde en çok kullanılan yöntemdir. LSB yönteminde gizlenecek olan mesajın her bir biti, resim dosyasını oluşturan verinin bir baytının en son bitine yazılır. LSB yöntemiyle veri gizleme işlemi Şekil 4.1’de gösterildiği gibidir.



Şekil 4.1. LSB yöntemiyle veri gizlenmesi

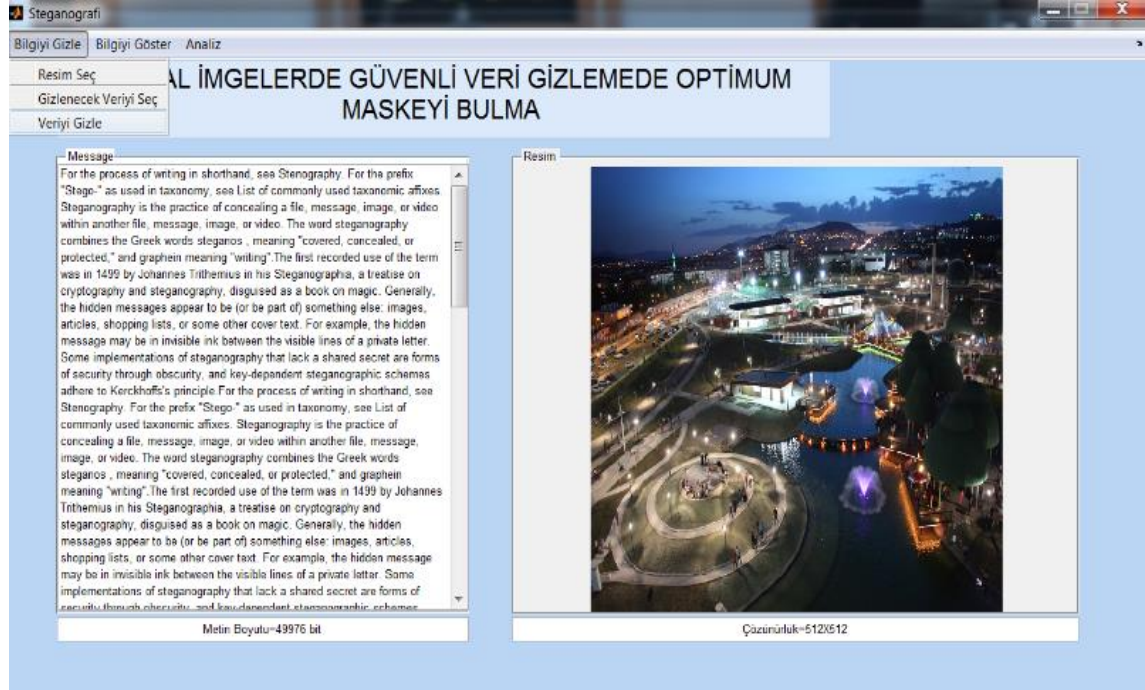
Veri gizlenirken ilk bayttan başlayarak son bayta kadar devam eden doğrusal bir sıralama izleme zorunluluğu bulunmamaktadır. Verinin gizlenmesi işleminde kullanılacak olan baytların sırası stego-key denilen bir anahtar aracılığıyla düzenlenmekte ve tamamen karışık sırayla veri gizlenmesi yapılabilmektedir. LSB yönteminde son bite eklemek yerine son iki bite eklemek gizlenebilecek veri miktarını iki katına çıkarmaktadır. Son iki bite veri gizleme işlemi yapıldığında örtü nesnesi üzerinde meydana gelen değişiklik hâlâ gözle görülemez iken, steganaliz yöntemlerle belirlenme olasılığı daha da artmaktadır.

Yapılan bu çalışmada, veri gizlenirken görüntü dosyasında oluşan değişikliği minimum seviyeye indirgenmesi ve gizlenen verinin güvenliğinin artırılması için uygun bir maske oluşturularak bu maskenin kullanılması sağlanmıştır. Kullanılan bu maske yardımıyla yetkisiz kişilerin veri dizisi elde etse bile bu verilerin çözülmesi engellenmektedir. Bu tez çalışması kapsamında geliştirilen uygulama, Matlab R2013.a programı kullanılarak GUI ara yüzünde geliştirilmiştir. Geliştirilen algoritmaya uygun oluşturulan akış diyagramı Şekil 4.2’de gösterilmektedir.



Şekil 4.2. Program akış diyagramı

Şekil 4.2’de verilen akış diyagramına bağlı kalınarak bir uygulama geliştirilmiştir. Geliştirilen uygulamanın ara yüzünde 3 adet menü ve bu menülere ait alt menüler kullanılmıştır. Menülerin kullanılmasının temel amacı uygulamanın kullanıcı dostu ve sade bir tasarıma sahip olmasına çalışılmasıdır. Veri gizleme, veri elde etme ve analiz işlemleri bu menüler aracılığıyla yapılmaktadır. Uygulama ana ekranı Şekil 4.3’te gösterilmektedir.



Şekil 4.3. Program ana ekranı

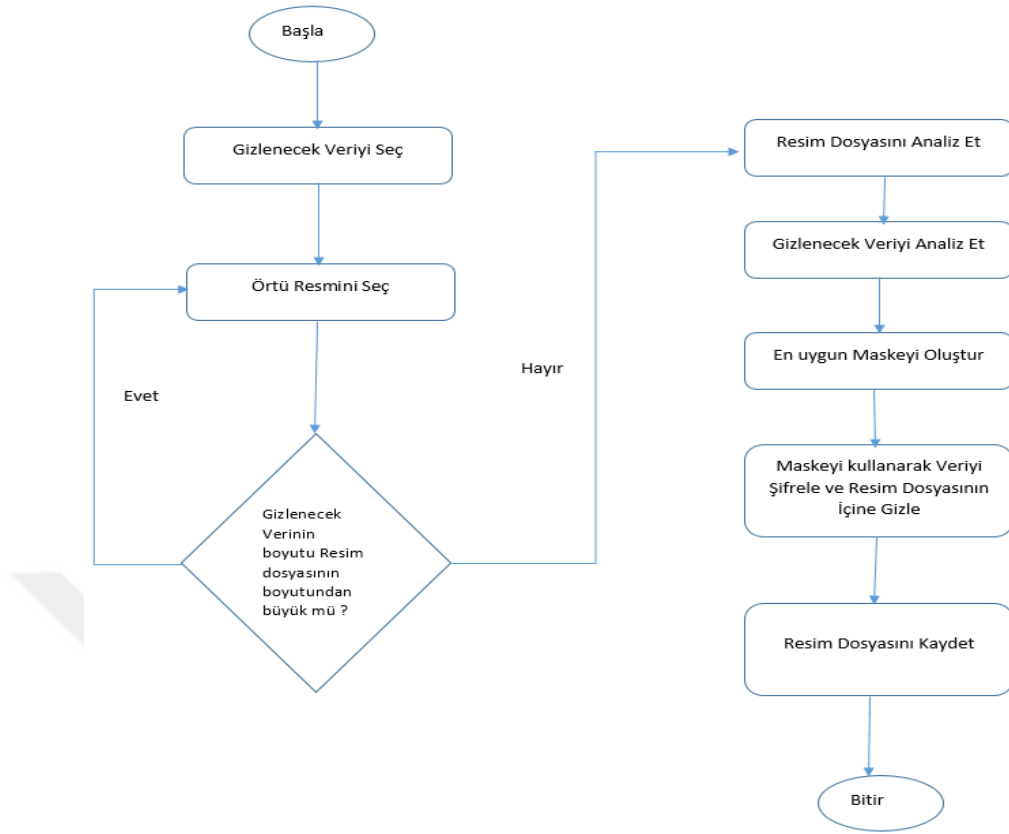
Geliştirilen uygulama Şekil 4.3'te görülmekte olduğu gibi 3 ana bölümden oluşmaktadır.

Bu bölümler;

1. Bilgiyi Gizle Bölümü
2. Bilgiyi Göster Bölümü
3. Analiz Bölümü

4.1. Bilgiyi Gizle Bölümü

Bilgiyi Gizle bölümünde örüntü dosyası seçimi, veri seçimi, maske oluşturulması, verilerin maske kullanılarak XOR işlemine tabi tutulması ve verinin örüntü dosyası içerisine gizlenmesi işlemleri yapılmaktadır. Veri gizleme bölümünün blok diyagramı Şekil 4.4'te gösterilmektedir.



Şekil 4.4. Bilgi gizleme blok diyagramı

Bilgiyi gizleme bölümünde, “**Resim Seç**” menüsü seçilerek dosya seçme penceresi açılır, buradan örtü nesnesi olan resim dosyası seçilerek, program ara yüzünde görüntülenir. Resim Seç menüsü tıklandığında çalışan fonksiyon **Open_Image** fonksiyonudur. Bu fonksiyon, resmin seçilmesini sağlar. Daha sonra gizlenecek olan metin dosyası “**Gizlenecek Veriyi Seç**” menüsü seçilerek dosya seçme penceresi açılır. Dosya seçme penceresinden gizlenecek olan veri metin dosyasından seçilir. Seçilen metin dosyasının içeriği ve boyutu program ara yüzünde görüntülenir. “**Gizlenecek Veriyi Seç**” menüsü altında yapılan işlemleri gerçekleştiren fonksiyonun ismi **OpenText**’tir. Bu fonksiyon, dosya açma penceresinin açılmasını ve dosyanın seçilmesini sağlar. Son olarak “**Veriyi Gizle**” butonu seçilerek veri gizleme işlemi gerçekleştirilmektedir. “Veriyi Gizle” butonu altında yapılan işlemleri gerçekleştiren fonksiyon “**HideText**” ve “**MaskeOlustur**” fonksiyonlarıdır. “HideText” fonksiyonunda öncelikle resim dosyasının boyutu ile gizlenecek olan verinin boyutu karşılaştırılır. $M \times N$ çözünürlüğe sahip örtü nesnesine gizlenebilecek veri miktarı denklem 4.1’de gösterilen formül yardımıyla hesaplanmaktadır.

$$Boyut = (M * N * 3) - 54 \quad (4.1)$$

Denklem 4.1’de toplam boyuttan 54 bitin çıkarılmasının sebebi her bir renk kanalı için maske değerlerini ve gizlenecek mesaj uzunluğu verisini bu bitlere saklanmasıdır. Eğer boyut yeterli ise resim dosyası ve gizlenecek veri “**MaskeOlustur**” fonksiyonuna gönderilir. “MaskeOlustur” fonksiyonunda, ilk adım olarak veri gizleme işlemi yapılırken örtü dosyasındaki bozulmayı en aza indirgeyecek maskeyi elde edebilmek için analiz işlemi yapılmaktadır. Analiz işlemi esnasında sayısal görüntüyü oluşturan renk kanalları (R,G,B) birbirinden ayrıştırılmaktadır. Şekil 4.5’te bir resim dosyasını oluşturan renk kanalları gösterilmektedir.

125	134	67	63	79	167	221	234	245	255
253	129	39	48	149	231	229	112	116	65
37	94	125	134	67	63	79	167	221	234
39	105	253	129	39	48	149	231	229	112
98	81	37	94	125	134	67	63	79	167
116	38	39	105	253	129	39	48	149	231
197	97	98	81	37	94	67	83	153	176
116	38	116	38	39	105	183	179	135	191
197	97	197	97	98	81	125	38	221	125
137	85	116	38	116	38	37	91	223	253

125	134	67	63	79	167	221	234	245	255
253	129	39	48	149	231	229	112	116	65
37	94	125	134	67	63	79	167	221	234
39	105	253	129	39	48	149	231	229	112
98	81	37	94	125	134	67	63	79	167
116	38	39	105	253	129	39	48	149	231
197	97	98	81	37	94	67	83	153	176
116	38	116	38	39	105	183	179	135	191
197	97	197	97	98	81	125	38	221	125
137	85	116	38	116	38	37	91	223	253

125	134	67	63	79	167	221	234	245	255
253	129	39	48	149	231	229	112	116	65
37	94	125	134	67	63	79	167	221	234
39	105	253	129	39	48	149	231	229	112
98	81	37	94	125	134	67	63	79	167
116	38	39	105	253	129	39	48	149	231
197	97	98	81	37	94	67	83	153	176
116	38	116	38	39	105	183	179	135	191
197	97	197	97	98	81	125	38	221	125
137	85	116	38	116	38	37	91	223	253

Şekil 4.5. Resim dosyası renk kanalları

Örüntü nesnesini oluşturan resim dosyasını renk kanallarına ayrıştırma işleminden sonra her bir renk kanalı ayrı ayrı NxN çerçevelere bölünmektedir. Resim dosyalarını oluşturan renk kanallarının NxN çerçevelere bölünmesi Şekil 4.6’da gösterilmektedir.

N

125	134	67	63	79	167	221	234	245	255
253	129	39	48	149	231	229	112	116	65
37	94	67	83	153	176	192	137	219	117
39	105	183	179	135	191	29	27	23	19
98	81	125	38	221	125	134	67	63	79
116	38	37	91	223	253	129	39	48	149
197	97	93	76	64	37	94	67	83	153
116	38	37	91	223	253	129	39	48	149
197	97	93	76	64	37	94	67	83	153
137	85	37	78	81	39	105	183	179	135

N

Şekil 4.6. Matrisin NxN çerçeveye bölünmesi

$N \times N$ çerçevenin her karesindeki pikselin son biti incelenip gruplanmakta ve bu grupların içerisinde kaç tane 1 ve 0 olduğu sayılmaktadır. $N \times N$ çerçevenin içindeki her bir karenin gruplanması ve sayılması Şekil 4.7’de gösterilmektedir.

125	134	67	63	79	167	221	234	245	Maske (1,1)	125	63	221
253	129	39	48	149	231	229	112	116	Maske (1,2)	134	79	234
37	94	67	83	153	176	192	137	219	Maske (1,3)	67	167	245
39	105	183	179	135	191	29	27	23	Maske (2,1)	253	48	229
98	81	125	38	221	125	134	67	63	Maske (2,2)	129	149	112
116	38	37	91	223	253	129	39	48	Maske (2,3)	39	231	116
197	97	93	76	64	37	94	67	83	Maske (3,1)	37	83	192
116	38	37	91	223	253	129	39	48	Maske (3,2)	94	153	137
197	97	93	76	64	37	94	67	83	Maske (3,3)	67	176	219
137	85	37	78	81	39	105	183	179				

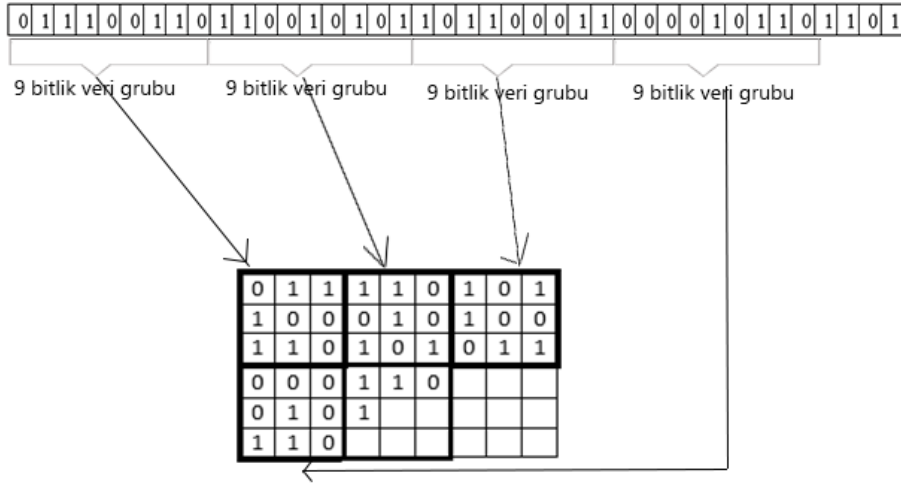
Şekil 4.7. Çerçevelerdeki 1 ve 0’ların gruplanması ve sayılması

Örüntü dosyasını oluşturan renk kanallarının ayrı ayrı gruplama ve sayma işlemi bittikten sonra gizlenecek verinin analiz aşaması başlamaktadır. Gizlenecek verinin analizinde ilk önce veriler 1 ve 0’lardan oluşan binary (ikilik) veri dizisine çevrilmektedir. Şekil 4.8’de “selam” verisinin binary sisteme çevrilmiş hali gösterilmektedir.

S	E	L	A	M
0 1 1 1 0 0 1 1 0 1 1 0 0 1 0 1 0 1 1 0 0 0 1 1 0 0 0 0 1 0 1 1 0 1 1 0 1				

Şekil 4.8. Binary veri dizisi

Gizlenecek veri gruplama ve sayma işleminin yapılabilmesi ve örüntü dosyası ile aynı formatta olabilmesi için matris formatına dönüştürülmektedir. Şekil 4.9’da veri dizisinin matris formatına nasıl dönüştürüldüğü gösterilmektedir.



Şekil 4.9. Binary dizinin matris formatına çevrilmesi

Şekil 4.9’da gösterilmekte olduğu gibi matris formatına dönüştürülen veri, Şekil 4.6’da gösterilmekte olan NxN çerçevelere bölünmektedir. Çerçevelere bölünen veri matrisi, Şekil 4.7’de gösterilmekte olduğu gibi gruplanmakta ve 1 ve 0 bitleri sayılmaktadır. Gizlenecek veri ve örtü nesnesinin gruplandırma ve sayma işlemi bittikten sonra oluşturulan gruptaki 1 ve 0’ların sayısı karşılaştırılarak her renk kanalı için ayrı bir maske oluşturulmaktadır. Maske oluşturma algoritma adımları Şekil 4.10’da gösterilmektedir.

1. **Adım** Başla
2. **Adım** Örtü nesnesini seç
3. **Adım** Veriyi seç
4. **Adım** Veriyi binary sisteme çevir
5. **Adım** Resim boyutunu ve binary dizi boyutunu karşılaştır
6. **Adım** Eğer boyut yeterli değilse 2. adıma git
7. **Adım** Renk kanalını oluşturan pikselleri NxN çerçevelere böl
8. **Adım** Bölünen çerçevelerdeki piksellerin son bitine bak ve 1 ve 0 say
9. **Adım** Binary dizisini matrise çevir
10. **Adım** Matris formatındaki verinin 1 ve 0 bitlerini say
11. **Adım** Örtü dosyasının ve verinin çerçevelerindeki bir ve sıfır sayılarını karşılaştır
12. **Adım** Maskeyi oluştur
13. **Adım** Maskeyi ana programa gönder
14. **Adım** Bitir

Şekil 4.10. Maske oluşturma algoritması

Şekil 4.10’da verilen algoritma adımlarıyla her bir renk kanalı için ayrı ayrı oluşturulan maskeler ana programa gönderilmektedir. Ana programa gelen maskeler kullanılarak veriler XOR işlemine tabi tutulmakta ve örüntü dosyasının içerisine en az değişikliği sağlayarak gizleme işlemi gerçekleştirilmektedir. Örnek olarak, “Selam” bilgisinin önerilen

yöntem ile Şekil 4.11’de gösterilen sayısal imgenin içerisine gizlenmek istendiği varsayılmaktadır.



Şekil 4.11. Veri gizlenecek sayısal imge (20x20 çözünürlüklü)

İlk adım olarak bilgi, binary formata çevrilmelidir. Binary formata çevrildiğinde oluşacak veri dizisi Şekil 4.12’de gösterilmektedir.

S								E								L								A								M								
0	1	1	1	0	0	1	1	0	1	1	0	0	1	0	1	0	1	1	0	1	1	0	0	0	0	1	1	0	0	0	0	1	0	1	1	0	1	1	0	1

Şekil 4.12. “Selam” bilgisinin binary sisteme çevrilmiş hali

İkinci adım olarak örtü nesnesi ile gizlenecek olan verinin boyutu karşılaştırılır. Sayısal imgenin ve gizlenmek istenen verinin boyutu Tablo 4.1’de gösterilmektedir.

Tablo 4.1. Boyut Karşılaştırması

Resmin Boyutu	Verinin Boyutu
20*20*3=1200 bit	40 bit

Örtü nesnesinin boyutu veriyi gizleyecek kadar büyük ise örtü nesnesi ve gizlenecek veri MaskeOlustur fonksiyonuna gönderilmek için hazırlanmaktadır. Renk kanallarına ayrılan örtü nesnesinin R kanalına ait pixel değerlerini içeren matris Şekil 4.13'te gösterilmektedir.

190	186	183	181	184	178	181	186	200	191	187	180	184	181	182	179	183	183	182	180
185	185	183	186	175	186	186	199	171	168	170	166	183	192	178	185	184	184	182	180
183	183	184	180	180	184	193	154	169	236	171	177	160	192	190	191	189	186	183	180
181	182	182	185	185	192	163	240	117	242	179	194	173	198	196	186	189	186	183	180
183	179	187	188	180	171	174	237	115	239	176	194	181	196	189	190	186	184	182	182
183	180	187	194	163	193	152	241	127	242	180	191	173	200	186	185	182	182	182	182
182	183	184	194	167	238	129	244	141	239	186	200	180	206	188	182	179	179	182	183
183	183	184	192	173	240	142	240	144	243	141	235	190	193	189	182	179	179	182	183
179	185	185	192	167	242	163	214	250	254	252	246	216	172	187	178	181	183	182	183
186	181	180	196	160	240	204	253	252	249	254	249	234	156	191	184	200	163	192	184
181	183	184	199	169	234	250	250	249	250	255	249	238	159	198	204	142	223	138	194
183	183	188	193	161	240	248	248	252	250	251	255	249	168	219	137	239	203	171	197
183	180	185	192	178	219	242	255	255	255	255	251	253	224	168	241	217	151	205	185
181	183	181	189	208	185	250	252	254	255	255	255	255	253	251	226	148	203	186	182
186	185	181	182	199	174	245	251	255	255	254	251	255	247	248	139	202	185	187	191
181	183	180	181	195	151	248	247	254	255	255	251	248	245	159	193	187	179	184	183
183	181	181	178	186	195	166	248	250	255	255	253	248	175	178	188	184	183	183	184
186	181	179	180	183	195	167	159	211	228	208	180	157	178	193	182	183	183	183	183
188	187	183	184	181	182	194	194	168	161	179	206	194	193	187	185	183	183	183	183
184	188	186	182	188	182	179	181	185	179	186	181	183	182	182	183	183	183	183	183

Şekil 4.13. Sayısal imgenin R kanalına ait pixel değerleri

Oluşan veri dizisinin örtü nesnesiyle aynı formatta olması için, veri dizisi iki boyutlu matris formatına çevrilmektedir. Şekil 4.14’te matris formatına çevrilmiş “Selam” verisi gösterilmektedir.

0	1	0	1	0	0	1	1
0	1	1	1	0	1	0	0
0	1	1	1	1	1	0	1
0	1	1	1	0	0	0	1
0	1	1	1	1	1	0	1

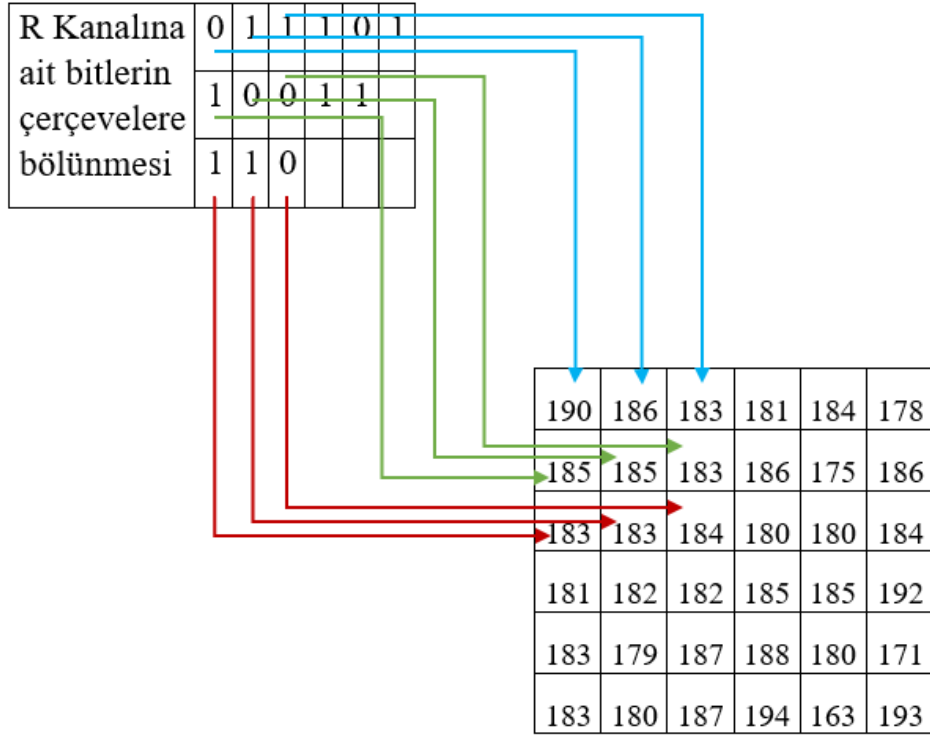
Şekil 4.14. Matris formatına çevrilmiş “Selam” verisi

Örtü nesnesi ve şekil 4.14’te gösterildiği gibi matris formatına çevrilen veri “MaskeOlustur” fonksiyonuna gönderilmektedir. “MaskeOlustur” fonksiyonunda örtü nesnesi ve veri matrisi gruplama ve sayma işlemine tabi tutulmaktadır. Bu işlem esnasında verinin kaçınıcı bitlerinin hangi kanala gömüleceği belirlenmektedir. Selam verisine ait bitlerin renk kanallarına dağılımı Şekil 4.15’te gösterilmektedir.

R Kanalı	0	1	1	1	0	0	1	1	0	1	0	1	1	1
G Kanalı	1	0	1	1	1	0	1	0	1	0	1	1	1	
B Kanalı	0	0	0	1	0	1	1	1	1	0	0	1	0	

Şekil 4.15. "Selam" verisine ait bitlerin renk kanallarına dağılımı

Bitlerin renk kanallarına dağılımından sonra her renk kanalına ait bitler ve renk kanallarına ait pixel değerleri kendi içerisinde NxN çerçevelere bölünmektedir. Şekil 4.16’da R kanalına gömülecek bitlerin ve R kanalının çerçevelere bölünmüş hali gösterilmektedir.



Şekil 4.16. Verilerin renk kanallarına dağıtılması ve gruplanması

Renk kanallarına gömülecek veriler belirlendikten sonra renk kanalları ve veriler aynı boyutta çerçevelere bölünür ve her çerçevedeki bir ve sıfır bitleri sayılır. Bitlerin sayma işlemi bittikten sonra renk kanalı ile veri çerçevelerindeki bir ve sıfırlar karşılaştırılarak en az değişimi sağlayacak maske oluşturulur. Oluşturulan maskeler ana programa gönderilir. Ana programa gönderilen bu maskeler sayesinde hem veriler XOR işlemine tabi tutularak gizlenmekte hem de örüntü dosyasının en az değişime uğraması sağlanmaktadır. “**MaskeOlustur**” fonksiyonunda oluşturulan ve ana programa gönderilen maskeler Şekil 4.17’de gösterilmektedir.

R KANALI	G KANALI	B KANALI
0 0 0	0 0 1	0 0 0
1 1 0	0 0 0	1 0 0
0 1 1	1 0 1	1 0 1

Şekil 4.17. Oluşturulan maske değerleri

Veri XOR işleminden geçirilip, gizlendikten sonra oluşan yeni resim dosyası isim verilerek kaydedilir. Veri gizlenmiş dosya ile veri gizlenmemiş dosya arasında gözle

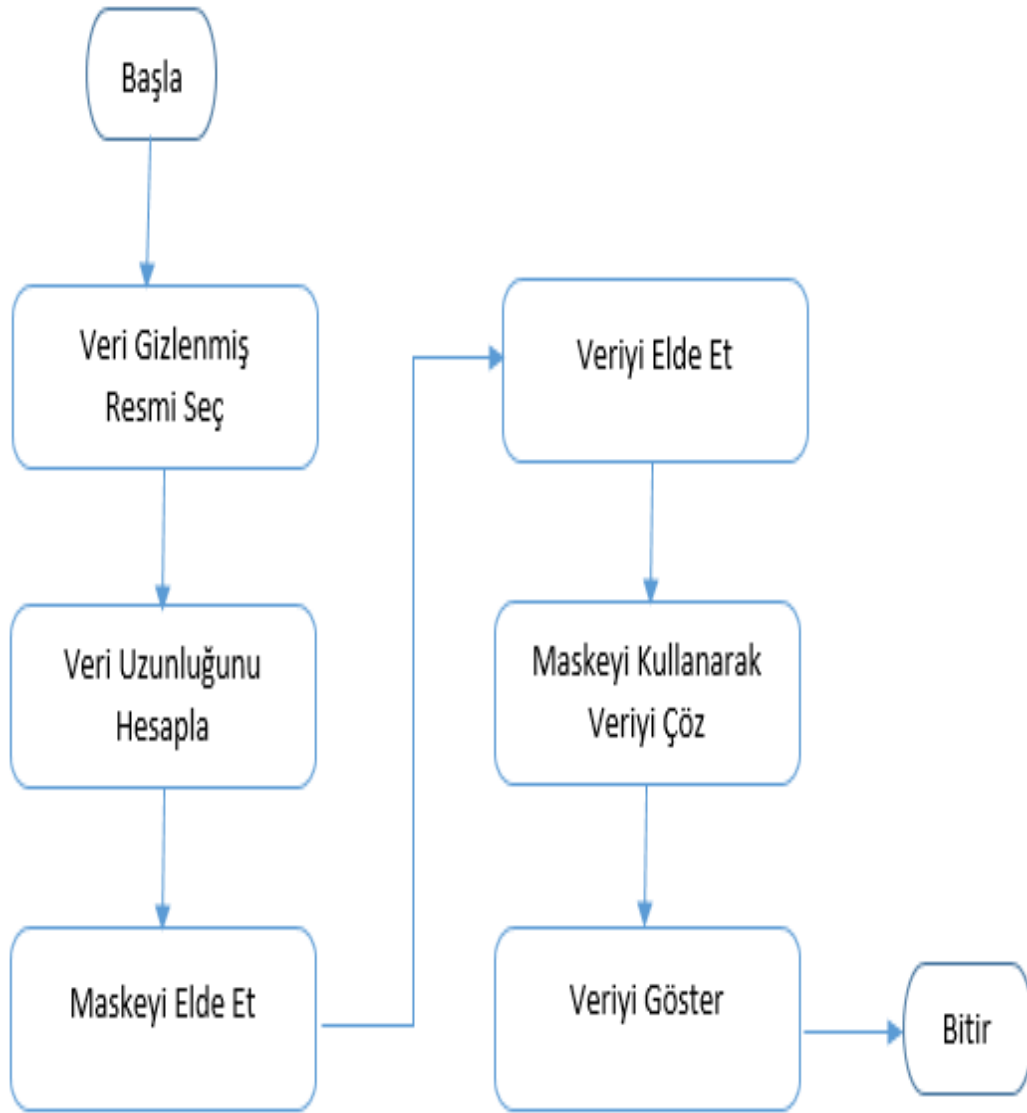
görülür bir fark oluşmamaktadır. Veri gizlenmiş resim ile orijinal resim Tablo 4.2’de birlikte verilmektedir.

Tablo 4.2. Orijinal resim ve veri gizlenmiş resim

		Boyut	PSNR	MSE
Orijinal Resim		1254 byte	63,7903	4,1864E-07
Veri Gizlenmiş Resim		1254 byte		

4.2. Bilgiyi Göster Modülü

Bilgiyi göster bölümünde veri gizlenmiş olan resim dosyasını açmak için “**Resim Seç**” menüsü kullanılır. Resim Seç menüsü altında çalışan fonksiyon “**Open_Image**” fonksiyonudur. Open_Image fonksiyonu dosya seçme penceresini açarak ilgili örtü nesnesinin seçilmesini sağlamaktadır. Seçilen örtü nesnesi ve örtü nesnesine ait çözünürlük bilgisi program ara yüzünde gösterilmektedir. Daha sonra “**Veriyi Göster**” butonu tıklanmaktadır. “**Veriyi Göster**” butonu altında yapılan işlemleri gerçekleştiren fonksiyon ise “**ShowText**” fonksiyonudur. “**ShowText**” fonksiyonu, resim dosyasını renk kanallarına (R,G,B) ayırmaktadır. Ayırıştırılan renk kanallarından gizlenmiş olan verinin uzunluğunu ve maske değerini elde etmektedir. İkinci adım olarak renk kanallarından gizlenmiş olan veri, bit dizisi olarak elde edilmektedir. Son adım olarak elde edilen bit dizisi maske değerleri kullanılarak ters XOR işlemine tabi tutularak çözülmekte ve çözülmüş olan veri dizisi metine dönüştürülmektedir. Son adım olarak elde edilen metin program ara yüzünde gösterilmektedir.



Şekil 4.18. Veriyi elde etme sözde kodları

Şekil 4.18’de veri gizlenmiş olan resim dosyasından verinin elde edilmesi ve verinin çözümlenerek metin haline dönüştürülmesini sağlayan bölümün blok diyagramı yer almaktadır.

4.3. Analiz Modülü

Bu bölümde yapılan deneysel çalışmada, elde edilen sonuçları karşılaştırmak amacıyla PSNR(Peak signal-to-noise ratio) ve MSE(Mean Squared Error) değerleri kullanılarak, orijinal resim ile veri gizlenmiş resim arasındaki farklılıkların miktarı hesaplanmaktadır. PSNR değeri ne kadar yüksek çıkarsa iki resim arasındaki fark da o kadar az olduğu kabul edilmektedir. PSNR değerinin yüksek çıkması için de MSE değerinin sıfıra yakın olması

gerekmektedir. Yani resmin orijinal hali ile veri gizlenmiş hali arasında fark yoktur demektir. PSNR değeri denklem 4.3'te yazılan formül yardımıyla hesaplanmaktadır.

$$MSE = \frac{1}{m*n} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i,j) - K(i,j)]^2 \quad (4.2)$$

$$PSNR = 10 * \log_{10} \left(\frac{MAX_I^2}{MSE} \right) \quad (4.3)$$

Kullanılan resimler 24 bit renkli resim olduğundan renkli resimler için PSNR değeri hesaplanması gerekmektedir. Renkli resimlerde PSNR değeri her bir renk kanalı için ayrı ayrı hesaplanmakta ve bu üç değer aritmetik ortalaması alınmaktadır. Renkli resim dosyaları için PSNR değeri denklem 4.4, denklem 4.5, denklem 4.6, denklem 4.7'deki formüller aracılığıyla hesaplanmaktadır.

$$PSNR_r = 20 * \log_{10} \left(\frac{255}{MSE_r} \right) \quad (4.4)$$

$$PSNR_g = 20 * \log_{10} \left(\frac{255}{MSE_g} \right) \quad (4.5)$$

$$PSNR_b = 20 * \log_{10} \left(\frac{255}{MSE_b} \right) \quad (4.6)$$

$$PSNR = \frac{PSNR_r + PSNR_g + PSNR_b}{3} \quad (4.7)$$

5. BULGULAR

Yapılan tez çalışması kapsamında geliştirilen uygulama içerisinde BMP resim dosyası türü kullanılmaktadır. BMP dosya türünün seçilmesinin nedeni, bu dosya formatının sıkıştırılmamış 24 bit renkli resim dosyası olmasıdır. BMP Dosya türü sıkıştırılmadığından dolayı veri gizlendiğinde resmin boyutunda herhangi bir değişiklik olmamaktadır. Örüntü dosyanın boyutunun değişmemesi de fark edilebilirliği azaltmaktadır. Geliştirilen uygulama, veri gizlerken en uygun maskeyi kullanarak örüntü dosyasında en az değişikliği yapması ve verileri XOR işleminden geçirdikten sonra gizlemesiyle LSB yönteminden ayrılmaktadır. Klasik LSB yönteminde resmin sadece son bitine ve ardışık olarak veri ekleme yapılmaktadır. Bu yöntemle yapılan gizleme işlemi kolaylıkla çözülebilmekte ve veriler elde edilebilmektedir.

Önerilen yöntemde ise veriler hem NxN çerçeveler halinde hem de XOR işleminden sonra gizlendiğinden veriler elde edilse bile çözümlenememektedir. Tablo 5.1’de önerilen yöntem ile gizlenmiş bir verinin, klasik LSB yöntemiyle elde edilmeye çalışılmış ve elde edilen sonuçlar Tablo 5.1’de gösterilmektedir.

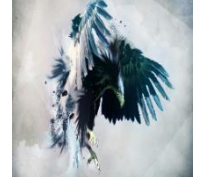
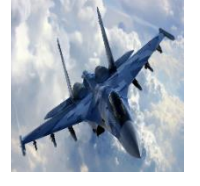
Tablo 5.1. LSB yöntemi ile elde edilen sonuçlar

Orijinal Metin	Elde Edilen Metin
Bu gün hava çok güzel	^ëñp·Â¼Åhð³áúÝ-ÞØi_ëÒ-QSv ûiÇÁÔÊ!È½ªchÈÉ¶,0ù¼-ú^Ëø·s+6]D& UÃžŸzt”ÅfÐiÖÖñcô&½GÂÂÐ=]-h}þdK È¹ÃE®N«â/÷;ëò;ù

Tablo 5.1’deki elde edilen sonuç ele alındığında önerilen yöntemle gizlenmiş olan bir veri klasik LSB yöntemiyle elde edilmeye çalışıldığında herhangi bir anlam ifade etmeyen veriler elde edilmektedir.

Yapılan deneysel çalışmalarda önerilen yöntemi klasik LSB yöntemiyle karşılaştırmak için internet ortamında rastgele seçilmiş olan 24 bitlik BMP formatında 12 adet resim dosyası kullanılmaktadır. Rastgele seçilen 12 adet resim dosyası Tablo 5.2’de gösterilmektedir.

Tablo 5.2. Uygulamada kullanılan resim dosyaları

Kümbet	Harput	Malatya Ulu Camii	Sultan Ahmet Camii
			
Elazığ Park	İstanbul	Kız Kulesi	Sokak
			
Kartal	Teknoloji	Uçak	Uzay
			

Deneysel çalışmalarda gizlenecek veri olarak kullanılmak üzere Mevlâna Celâleddin-i Rûmî’nin dizeleri seçilmiştir. Gizlenmek için seçilen dizeler Şekil 5.1’de gösterilmektedir.

*"Bilemezsin sana verecek bir armağanı ne çok aradığımı.
Hiçbir şey içime sinmedi.
Altın madenine altın sunmanın ne anlamı var, ya da okyanusa su..?
Düşündüğüm her şey, Doğu'ya baharat götürmek gibiydi.
Kalbimi ve ruhumu vermemin bir yararı yok,
çünkü sen zaten bunlara sahipsin.
O yüzden sana bir ayna getirdim.
Kendine bak ve beni hatırla..!"*
Mevlâna Celâleddin-i Rûmî

Şekil 5.1. Gizlemek için kullanılan veri

Yapılan deneysel çalışmada Tablo 5.2’de gösterilmekte olan internet üzerinden rastgele seçilen 12 adet 24 bitlik BMP formatında resim dosyasına Şekil 5.1’de gösterilmekte olan veri dosyası, hem klasik LSB yöntemiyle hem de önerilen yöntem ile gizlenmiştir. Gizleme işlemi sırasında RGB renk kanalları için ayrı ayrı 3x3 boyutunda maskeler oluşturulmuştur. Her renk kanalı için ayrı maske oluşturulmasının sebebi her renk kanalındaki piksellere yerleşecek olan bitlerin birbirinden bağımsız olması ve her bir pixeli farklı etkilemesidir. Elde edilen maske değerleri Tablo 5.3’te gösterilmektedir.

Tablo 5.3. Resimler ve bu resimlerin kanalları için oluşturulmuş maskeler

Resim	Rmask	Gmask	Bmask
Harput	1 1 1 0 0 0 0 1 1	1 1 0 1 0 0 1 0 0	0 1 0 0 1 1 0 1 0
Kartal	1 0 1 0 0 0 1 0 0	0 0 1 1 1 0 1 1 1	1 1 1 1 0 1 1 1 1
İstanbul	1 0 1 1 0 1 0 1 1	0 0 0 0 0 1 0 0 0	1 0 0 1 0 0 1 0 0
Elazığ Park	1 0 1 1 0 1 1 0 1	1 1 0 1 1 1 0 0 1	0 0 0 1 0 0 0 0 0
Sultan Ahmet Camii	1 0 1 1 0 1 1 0 1	0 0 0 0 0 0 0 0 0	1 0 0 1 0 0 1 0 0
Kümbet	0 0 1 1 0 1 0 0 1	1 0 1 1 1 1 1 1 0	0 1 1 0 1 1 0 1 1
Teknoloji	0 1 0 0 1 0 0 1 0	0 1 0 1 1 1 0 0 1	0 0 0 0 0 0 0 0 0
Uçak	1 1 0 1 0 0 0 1 1	1 1 0 1 0 1 1 0 1	0 1 1 0 0 1 0 0 0
Uzay	0 0 0 1 0 0 1 0 0	0 1 0 0 1 0 0 1 0	0 1 0 1 0 0 1 1 0
Malatya Ulu Camii	0 1 0 0 1 0 0 1 0	1 1 1 1 1 1 1 1 1	1 0 0 1 0 0 1 0 0
Kız Kulesi	1 0 1 1 0 0 1 0 1	1 1 1 0 0 0 1 1 1	0 1 1 0 0 1 1 0 0
Sokak	0 1 0 0 1 0 0 1 0	0 0 0 0 0 0 0 0 0	1 0 0 1 0 0 1 0 0

Tablo 5.3’te verilen maske değerleri resim dosyası ve verinin birlikte analiz edilmesiyle oluşturulmuştur. Analiz aşamasının ayrıntıları Bölüm 4.1 Bilgiyi Gizle Bölümü’nde

anlatılmaktadır. Önerilen yöntemle oluşturulan maske değerleri, örtü nesnesi olarak kullanılan sayısal imge dosyası veya gizlenecek olan veri dosyası değiştiği zaman otomatik olarak değişmektedir. Tablo 5.3'teki maske değerleri kullanılarak 12 adet sayısal imge dosyasına uygulanan veri gizleme işleminden sonra resimler orijinal halleriyle karşılaştırılmaktadır. Hem önerilen yöntem için hem de klasik LSB yöntemini değerlendirmek için PSNR ve MSE değerleri hesaplanmıştır. Yapılan analizler sonucunda elde edilen PSNR ve MSE değerleri Tablo 5.4'te gösterilmektedir.

Tablo 5.4. Elde edilen PSNR ve MSE değerleri

RESİMLER	Önerilen Yöntem				LSB			
	PSNR	MSE			PSNR	MSE		
		R	G	B		R	G	B
Kümbet	69.7290	0.0090	0.0069	0.0090	45.1817	0.0041	0.2478	0.2918
Harput	69.6398	0.0014	0.0010	0.0090	45.5759	0.0044	0.3506	0.2584
Elazığ park	69.6806	0.0043	0.0091	0.0099	32.4187	0.0042	0.3735	0.3769
İstanbul	69.5916	0.0090	0.0017	0.0014	38.7758	0.0041	0.3728	0.3587
Kız kulesi	69.5546	0.0047	0.0090	0.0011	65.3277	0.0043	0.0137	0.0036
Malatya Ulu Camii	69.7715	0.0021	0.0047	0.0012	50.9231	0.0088	0.2525	0.2713
Sokak	69.7450	0.0019	0.0018	0.0014	35.4337	0.0041	0.3159	0.3149
Sultan Ahmet Camii	69.6665	0.0043	0.0018	0.0014	45.1750	0.0042	0.2793	0.1152
Teknoloji	69.7635	0.0018	0.0012	0.0012	35.7114	0.0041	0.2250	0.2384
Uçak	69.6869	0.0012	0.0087	0.0099	46.9677	0.0042	0.0252	0.0175
Kartal	69.6492	0.0073	0.0094	0.0013	35.7280	0.0040	0.3753	0.3841
Uzay	69.5908	0.0080	0.0014	0.0013	44.3532	0.0042	0.0688	0.1329

Tablo 5.4'teki sonuçlara bakıldığında, önerilen yöntemin klasik LSB yöntemine göre çok daha yüksek PSNR değerlerine sahip olduğu görülmektedir. Yüksek PSNR değerine sahip olması, önerilen yöntem ile yapılan veri gizleme işleminde resim dosyasındaki bozulmanın klasik LSB yöntemine göre daha az olduğunu göstermektedir.

6. SONUÇLAR VE TARTIŞMA

Yapılan bu tez çalışmasında klasik LSB yöntemi incelenmiş ve LSB yönteminin dezavantajlarından iki tanesi iyileştirilmeye çalışılmıştır. Bu dezavantajlardan birincisi, örtü dosyası üzerindeki bozulma miktarı ikincisi ise gizlenmekte olan verinin kolaylıkla elde edilmesidir. Önerilen yöntem ile veri gizlenirken örüntü dosyası üzerinde en az değişikliği sağlayacak maske üretilmiş ve bu maske hem veriyi gizlerken en az bit değişikliğini sağlamakta hem de veriyi XOR işlemine tabi tutmak için anahtar olarak kullanılmaktadır. Önerilen yönteminin bir diğer avantajı ise LSB yönteminde olduğu gibi veriyi ardışık bir sırayla değil resim dosyasını $N \times N$ çerçevelere bölerek veriyi gizlemesidir. Bu tez çalışmasında örüntü nesnelerine önerilen yöntem ile ve klasik LSB yöntemi ile veri gizlenmiş ve PSNR değerleri kullanılarak karşılaştırma yapılmıştır. PSNR değerleri karşılaştırıldığında elde edilen sonuçlar klasik LSB yönteminden daha iyi olduğu görülmüştür. Elde edilen sonuçlar Tablo 5.4'te karşılaştırmalı olarak listelenmiştir. Bu çalışmada aynı zamanda maske anahtar olarak kullanıldığından elde etme saldırıları da başarısız olmuştur.

Yapılan çalışmada klasik LSB yöntemine göre daha iyi sonuçlar elde edilmesine rağmen önerilen yöntemin iki tane önemli dezavantajı da bulunmaktadır. Önerilen yöntemin dezavantajlarından birincisi klasik LSB yönteminde olduğu gibi gizlenecek veri boyutunun örtü dosyasının boyutuyla sınırlı olmasıdır. Örtü dosyasının RGB renk kanallarının üçü de kullanılarak gizlenecek veri miktarı üç katına çıkarılarak gizlenecek veri boyutu artırılarak bu dezavantaj azaltılmaya çalışılmıştır. Önerilen yönteminin bir diğer dezavantajı ise örtü nesnesinin 54 biti gizlenecek verinin uzunluğu ve maskelerin saklanması için kullanıldığından gizlenecek verinin boyutu azalmaktadır. Verinin güvenliği ele alındığında kapasite azalır iken gizlenen verinin elde edilmesi zorlaştığından bu dezavantaj görmezden gelinebilecek düzeydedir. Önerilen yöntem ile veri gizlenirken kullanılan maskelerin boyutu değiştirilmek suretiyle gizlenecek bilgi daha da karmaşık bir şekilde değiştirilerek gizleme yönteminin geliştirilmesi mümkündür.

KAYNAKLAR

- [1] **C. Koçak**, Kriptografi ve stenografi yöntemlerini birlikte kullanarak yüksek güvenli veri gizleme, Erciyes Üniversitesi Fen Bilimleri Enstitüsü Dergisi, cilt 31, no. 2, pp. 115-123, 2015.
- [2] **A. Cheddad, J. Condell, P. M. Kevitt ve K. Curran**, Digital Image Steganography: Survey and Analysis of Current Methods, Signal processing, cilt 90, no. 3, pp. 727-752, 2010.
- [3] **A. Şahin**, Görüntü Steganografide Kullanılan Yeni Metotlar ve Bu Metodların Güvenirlikleri, Doktora Tezi, Edirne, 2007.
- [4] **T. Tuncer**, Resimler için Veri Gizleme Tabanlı Bilgi Güvenliği Uygulamaları, Doktora Tezi, Elazığ, 2011.
- [5] **M. A. Atıcı**, Steganografik Yaklaşımların İncelenmesi, Tasarımı ve Geliştirilmesi, Yüksek Lisans Tezi, Ankara, 2007.
- [6] **O. Ünlü**, Ortam Ve Yöntem Bağımsız Steganografik Kütüphane Tasarımı, Yüksek Lisans Tezi, Ankara, 2012.
- [7] **F. Akar**, Bmp Resimler için Veri Gizleme Tabanlı Bilgi Güvenliği Uygulamaları.
- [8] **Wikipedia**, Wikipedia, Wikimedia Foundation Inc., 24 05 2013. [Çevrimiçi]. Available: <https://tr.wikipedia.org/wiki/JPEG>. [Erişildi: 20 01 2016].
- [9] **L. Rastislav ve N. P. Konstantinos**, Color Image Processing: Methods and Applications, Newyork: CRC Press Taylor & Francis Group, 2007.
- [10] **İ. Avcıbaş, M. Kharrazi, N. Memon ve B. Sankur**, Image Steganalysis with Binary Similarity Measures, EURASIP Journal on Advances in Signal Processing, pp. 2749-2757, 2005.
- [11] **Wikipedia**, Wikipedia, Wikimedia Foundation, Inc, [Çevrimiçi]. Available: <https://en.wikipedia.org/wiki/Steganography>. [Erişildi: 28 09 2017].
- [12] **R. J. Anderson ve F. A. P. Petitcolas**, On the Limits of Steganography, IEEE Journal On Selected Areas In Communications, cilt 16, no. 4, pp. 474-481, 1998.
- [13] **E. Avcı, T. Tuncer ve F. Ertam**, Çok Katmanlı Görüntü Steganografi, 7th Internatinal Conference on Information Security and Cryptology, İstanbul, 2014.

- [14] **Y. Yalman ve İ. Ertürk**, Kişisel Bilgi Güvenliğinin Sağlanmasında Steganografi Biliminin Kullanımı, *Bilgi Çağında Varoluş: “Fırsatlar ve Tehditler” Sempozyumu*, İstanbul, 2009.
- [15] **Wikipedia**, Wikipedia, Wikimedia Foundation Inc., 25 Şubat 2015. [Çevrimiçi]. Available: https://en.wikipedia.org/wiki/Cardan_grille. [Erişildi: 15 Ocak 2016].
- [16] **S. B. Sadkhan**, Cryptography: Current status and future trends, *Information and Communication Technologies: From Theory to Applications*, 2004, Proceedings, pp. 417- 418, 19 Nisan 2004.
- [17] **W. Bender, D. Gruhl ve N. Morimoto**, Techniques for data hiding, *IBM systems journal*, cilt 3, no. 4, pp. 313-336, 1996.
- [18] **H. Kutucu ve M. Kaya**, Cryptography And Network Security, 2002. [Çevrimiçi]. Available: web.karabuk.edu.tr/hakankutucu/documents/stegano.doc. [Erişildi: 04 Nisan 2017].
- [19] **N. Samphaiboon**, Steganography via running short text messages, *Multimed Tools Appl*, no. 52, pp. 569-596, 12 2009.
- [20] **R. H. Alwan, F. J. Kadhim ve A. T. Al-Taani**, Data Embedding Based on Better Use of Bits in Image Pixels, *International Journal of signal processing*, cilt 2, no. 1, pp. 104-107, 2005.
- [21] **J. Fridrich, M. Goljan ve R. Du**, Detecting LSB Steganography in Color and GrayScale Images, *IEEE MultiMedia*, cilt 8, no. 4, pp. 22-28, 2001.
- [22] **S. Xin-Giang ve Luo Hui**, A new steganography method based on hypertext, *Radio Science Conference*, 2004. Proceedings. 2004 Asia-Pacific, Qingdao, China, 2004.
- [23] **R. Joshi, L. Gagnani ve S. Pandey**, Image Steganography With LSB, *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, cilt 2, no. 1, pp. 228-229, 2013.
- [24] **L. M. Marvel, C. G. Boncelet ve C. T. Retter**, Spread spectrum image steganography, *IEEE Transactions On Image Processing*, cilt 8, no. 8, pp. 1075-1083, 08 1999.
- [25] **R. Karakış ve İ. Güler**, Medikal Dicom Görüntüler için Steganografi Uygulaması, 7. Uluslararası Bilgi Güvenliği ve Kriptoloji Konferansı, İstanbul, 2014.

- [26] **R. Karakış ve İ. Güler**, Bulanık Mantık Tabanlı Görüntü Steganografi Uygulaması, Signal Processing and Communications Applications Conference (SIU), 2014 22nd. IEEE, Trabzon, 2014.
- [27] **Z. Şentürk, Y. Vatankulu ve Y. Durna**, LSB Yöntemiyle Renk Değişimi Esaslı Yeni Bir Steganografik Yaklaşım, Signal Processing and Communications Applications Conference (SIU), Malatya, 2015.
- [28] **E. M. Esin ve E. Güvenoğlu**, Resim İçine Yazı Gizlenmesi Amacıyla Kullanılan LSB Ekleme Yönteminin Shuffle Algoritmasıyla İyileştirilmesi, Türkiye Bilişim Vakfı Bilgisayar Bilimleri ve Mühendisliği Dergisi, cilt 2, no. 2, 2006.
- [29] **M. Yeşilyurt, A. T. Özcerit, Y. Yalman ve İ. Ertürk**, Sayısal İmgeler İçin Ayrık Kosinüs Dönüşümü Esaslı Veri Gizlemenin Ataklara Dayanıklılığı, Akademik Bilişim 2012, Uşak, 2012.
- [30] **T. Tuncer, Ş. Doğan ve E. Avcı**, Renkli İmgelerde Gizlenen Verilerin Görsel Ataklara Karşı Dayanıklılığının Tespiti İçin Bir Steganografi Uygulaması, 6th International Advanced Technologies Symposium (IATS'11), ELAZIĞ, 2011.
- [31] **M. Kurt ve N. Duru**, Menezes Vanstone Eliptik Eğri Şifreleme Algoritması Kullanılarak Video Dosyaları Üzerinde Steganografi Uygulaması, 2014 IEEE 22nd Signal Processing and Communications Applications Conference (SIU 2014), Trabzon, 2014.
- [32] **M. Kurt Pehlivanoglu, B. Kır Savaş ve N. Duru**, Koblitz Metodu Kullanılarak Video Dosyaları Üzerinde LSB Tabanlı Steganografi Uygulaması, 23. Sinyal İşleme ve İletişim Uygulamaları Kurultayı, Malatya, 2015.
- [33] **Y. Yıldız ve A. T. Özcerit**, 24 bit renkli hareketli resimler (video) üzerinde geliştirilen sırtörme yöntemi, Sakarya Üniversitesi Fen Bilimleri Dergisi, cilt 19, no. 1, pp. 1-6, 2015.
- [34] **A. Şahin, E. Buluş, H. N. Buluş ve M. T. Sakallı**, 24-Bit Renkli Resimler Üzerinde Uygulanan Rs Steganalizde Maske Seçimlerinin Etkileri, Elektrik Elektronik Bilgisayar Mühendisliği Sempozyumu, BURSA, 2006.
- [35] **N. Mohammad-Saleh, A. Alireza ve Z. Habib**, A contextual based double watermarking of PET images by patient ID and ECG signal, Computer Methods and Programs in Biomedicine, no. 104, pp. 418-425, 2011.

- [36] **Ş. Doğan**, A new data hiding method based on chaos embedded genetic algorithm for color image, *Artificial Intelligence Review*, cilt 1, no. 46, pp. 129-143, 2016.
- [37] **Y. Cheng-Hsing, W. Chi-Yao ve W. Shiuh-Jeng**, Adaptive Data Hiding in Edge Areas of Images With Spatial LSB Domain Systems, *IEEE Transactions on Information Forensics and Security*, cilt 3, no. 3, pp. 488-497, 2008.
- [38] **P. Praveenkumar, P. Priya, T. K., R. J.B.B. ve A. Rengarajan**, Convolutud Rubik's Encoded Quantum Polarized-A Secure DICOM Image, 2016 International Conference on Computer Communication and Informatics, Coimbatore, 2016.
- [39] **M. T. Sakallı, A. Mesut ve A. Şahin Mesut**, Görüntü Steganografide Gizlilik Paylaşım Şemalarının Kellanılması ve Güvenliğe Etkileri.
- [40] **A. Şahin, E. Buluş ve M. T. Sakallı**, Gri Seviye Resimler Üzerinde Rasgele LSB Yöntemini ve Sayı Teorisini Kullanarak Bilgi Gizleme ve Steganaliz, *Akademik Bilişim Konferansları*, 2006.
- [41] **S. Tunçer ve C. Karakuzu**, Veri Güvenliğini Artırmak Amacıyla Bilgiyi Şifreleme ve Steganografik Yöntemlerle Görüntüye Gizleme, *Elektrik-Elektronik ve Bilgisayar Sempozyumu*, Tokat, 2016.
- [42] **Ö. Kurtuldu ve N. Arıca**, İmge Kareleri Kullanan Yeni Bir Steganografi Yöntemi, *Deniz Bilimleri ve Mühendisliği Dergisi*, cilt 5, no. 1, pp. 107-118, 2009.
- [43] **K. Sunil**, A Lossless Data Hiding Technique For Medical Images Using Lsb Substitution, 14th sem M.Tech(L.S.P)SJCIT, Chikballapur, 2014.
- [44] **A. H. Lashkari, A. A. Manaf, M. Masrom ve S. M. Daud**, A survey on image steganography algorithms and evaluation., *Digital Information Processing and Communications*. Springer Berlin Heidelberg, pp. 406-418, 2011.
- [45] **A. A.-A. Gutub, Y. S. Elarian, S. M. Awaideh ve A. K. Alvi**, Utilizing Diacritic Marks for Arabic Text Steganography, *Kuwait journal of Ecience and Engineering*, cilt 1, no. 37, pp. 89-109, 2010.
- [46] **F. A. P. Petitcolas, A. Ross J. ve K. Markus G.**, Information Hiding A Survey, *Proceedings of the IEEE*, cilt 87, no. 7, pp. 1062-1078, 1999.
- [47] **E. Esen ve İ. Atıl**, Standart Görüntü İşleme Yöntemlerinin Steganalize Etkisi,» 2015 23rd Signal Processing and Communications Applications Conference (SIU),

Malatya, 2015.

- [48] **E. Öztürk, A. Mesut Şahin ve A. Mesut**, LSB Ekleme Yönteminde Bilgi Gizleme için Tek Renk Kanal Kullanımının Güvenliğe Etkileri, IV. Ağ ve Bilgi Güvenliği Sempozyumu, ANKARA, 2011.
- [49] **F. Doğan, E. A. Güzeldereli ve Ç. Özdemir**, Medikal Görüntü İçerisine Tıbbi Bilgilerin Gömülmesi için Yeni Bir Yaklaşım, SAÜ. Fen Bilimleri Dergisi, cilt 17, no. 2, pp. 277-286, 2013.
- [50] **R. Karakış, İ. Güler, İ. Çapraz ve E. Bilir**, Medikal Görüntülerin Güvenliği için Bulanık Mantık Tabanlı Steganografi Yöntemi, 23rd Signal Processing and Communications Applications Conference, Malatya, 2015.
- [51] **M. Petros L. K. ve M. Ilias**, Sensitive Patient Data Hiding using a ROI Reversible Steganography Scheme for DICOM Images, Journal of Medical Systems, no. 40, pp. 156-173, 2016.
- [52] **T. P. ve G. .Aghila**, Reversible dynamic secure steganography for medical image using graph coloring, Health PolicyandTechnology, pp. 151-161, 13 Haziran 2013.

ÖZGEÇMİŞ

Adı Soyadı	: Yıldıray YİĞİT
Doğum Tarihi	: 12.01.1984
Tel	: 0507 093 26 76
Eposta	: yildirayyigit@gmail.com

ÖĞRENİM DURUMU:

Lisans	Üniversite/Okul Adı	: Fırat Üniversitesi
2008	Fakülte/Enstitü/Birim Adı	: Teknik Eğitim Fakültesi
	Bölüm/Program/Ana Bilim Dalı	: Bilgisayar Öğretmenliği
	Adı	: Trafik Sigorta Kasko Otomasyonu
	Tez / Proje Adı	
Yüksek Lisans	Üniversite/Okul Adı	: Fırat Üniversitesi
	Fakülte/Enstitü/Birim Adı	: Teknoloji Fakültesi
	Bölüm/Program/Ana Bilim Dalı	: Yazılım Mühendisliği
	Adı	: Sayısal İmgelerde Güvenli Veri Gizlemeye
	Tez / Proje Adı	: Yönelik Uygun Maske Belirleme

BİLDİĞİ DİLLER

Yabancı Dil	Sınav Adı	Puanı
İngilizce	Yökdil	52,5

AKADEMİK GÖREVLER:

Öğretim Görevlisi Ahlat M.Y.O. Bitlis Eren Üniversitesi 2013 - Devam Ediyor

İDARİ GÖREVLER

Görev Tanımı	Tarih Aralığı
Ahlat Meslek Yüksek Okulu Bölüm Başkanı	01.09.2014 - 01.09.2015

AKADEMİK FAALİYETLER

Uluslararası Bir Kurum/Kuruluş Tarafından Düzenlenmiş, Hakemli Bir Kongre/Sempozyumda Sunulmuş ve Kongre Bildiriler Kitabında Veya Bildiriler Cd Sinde Yer Alan

1. Yıldıray YİĞİT, Murat KARABATAK, The Perspectives Of Individuals About Cyber Crimes In Turkey, "26 Th TheIIEr International Conference ", , (2015), s.94
2. Yıldıray YİĞİT, Murat KARABATAK, Developing LSB Method Using Mask in Colored Images, "6 Th International Symposium on Digital Forensic and Security ",Antalya , (2018)