T.C.

SELÇUK ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

BİLGİ GÜVENLİĞİ İÇİN METİN STEGANOGRAFİSİNDE YENİ BİR YAKLAŞIM

Esra ŞATIR

DOKTORA TEZİ

BİLGİSAYAR MÜHENDİSLİĞİ

Nisan-2013
KONYA

TEZ KABUL VE ONAYI

Esra ŞATIR tarafından hazırlanan "Bilgi Güvenliği için Metin Steganografisinde Yeni bir Yaklaşım" adlı tez çalışması 11/04/2013 tarihinde aşağıdaki jüri tarafından oy birliği / ~~oy çokluğu~~ ile Selçuk Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı'nda DOKTORA TEZİ olarak kabul edilmiştir.

Jüri Üyeleri	**İmza**
Başkan Prof. Dr. İnan GÜLER	
Danışman Prof. Dr. Hakan IŞIK	
Üye Prof. Dr. Ahmet ARSLAN	
Üye Prof. Dr. Novruz ALLAHVERDİ	
Üye Doç. Dr. Harun UĞUZ	

Yukarıdaki sonucu onaylarım.

Prof. Dr. Aşır GENÇ
FBE Müdürü

Bu tez çalışması BAP tarafından 11101037 nolu proje ile desteklenmiştir.

TEZ BİLDİRİMİ

Bu tezdeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edildiğini ve tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

DECLARATION PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Esra ŞATIR
Tarih:11/04.2013

ÖZET

DOKTORA TEZİ

BİLGİ GÜVENLİĞİ İÇİN METİN STEGANOGRAFİSİNDE YENİ BİR YAKLAŞIM

Esra ŞATIR

Selçuk Üniversitesi Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Prof. Dr. Hakan IŞIK
2013, 91 Sayfa

Jüri
Danışman: Prof. Dr. Hakan IŞIK
Prof. Dr. İnan GÜLER
Prof. Dr. Ahmet ARSLAN
Prof. Dr. Novruz ALLAHVERDİ
Doç. Dr. Harun UĞUZ

Algılanamazlık, kapasite ve sağlamlık, bilgi gizleme sistemlerinin üç ana gereksinimleridir. Bu gereksinimler birbiriyle ödünleşim halindedir. Bu çalışmada, yeni bir metin steganografi yaklaşımı önerilerek algılanamazlık ve kapasite konuları ele alınmış, güvenlik konusu desteklenmiştir. Çalışmanın yeniliği ve katkısı; algılanamazlık korunurken, örten ortama saklanabilen veri miktarını artırmak ve güvenliği sağlamaktır. Algılanamazlık, stego ortamı forward mail platformu olarak düzenleyerek ve iki taraf arasındaki iletişim için bu ortam kullanılarak sağlanmıştır. Gizli bilgi, grup hitabında kullanılabilecek ve dilbilgisi kurallarına uyularak oluşturulmuş metinlerden oluşan metin tabanından seçilen bir metin içerisine saklanmıştır. Saklama, örten metnin orijinalliği korunarak gerçekleştirilmiştir. Güvenlik, çıkarım aşamasının karmaşıklaştırılması için veri sıkıştırma teknikleri ve steganografik ortamı analiz etmeye çalışan bir gözlemci için rastgeleselliği sağlamak amacıyla kombinatorik tabanlı kodlama kullanılarak sağlanmıştır. Metinsel veriyle çalışıldığı için sıkıştırma sonucu bilgi kaybı olmamalıdır. Dolayısıyla, literatürdeki yaygın kullanımları ve dikkate değer sıkıştırma oranları sebebiyle LZW ve Huffman kayıpsız sıkıştırma algoritmaları tercih edilmiştir. Ayrıca güvenliği artırmak amacıyla simetrik şifreleme usulü paylaşılan stego anahtar kullanılmıştır. Gerçekleştirilen deneyler neticesinde 300 karakterlik gizli mesaj için, LZW kodlaması kullanıldığında elde edilen kapasite değeri %8.37, Huffman kodlaması kullanıldığında elde edilen kapasite değeri %9.34 olmaktadır. Önerilen yaklaşımın güvenlik analizi, gizli mesajın çıkarılması için gereken kombinasyon sayısı formüle edilip hesaplanarak gerçekleştirilmiştir. Son olarak, önerilen metot, literatürdeki diğer güncel metotlarla en yaygın ölçüt olan kapasite, açısından karşılaştırılmıştır. Elde edilen kapasite değerleri oldukça yüksek iken, gizli mesajın uzunluğu arttıkça kapasite değerlerinin de arttığı görülmüştür. Böylece uzunluk artışının kapasite üzerindeki dezavantajı, avantaja çevrilmiştir. Önerilen metotta işlemler sayılarla gerçekleştirildiğinden ötürü, dile bağımlılık asgari düzeydedir.

Anahtar Kelimeler: Huffman kodlaması, kayıpsız veri sıkıştırma, LZW kodlaması, metin steganografisi, steganografi

ABSTRACT

Ph.D THESIS

A NEW TEXT STEGANOGRAPHY APPROACH FOR INFORMATION SECURITY

Esra ŞATIR

THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCEOF
SELÇUK UNIVERSITY
DEPARTMENT OF COMPUTER ENGINEERING

Advisor: Prof. Dr. Hakan IŞIK
2013, 91 Pages

Jury
Advisor: Prof.Dr. Hakan IŞIK
Prof. Dr. İnan GÜLER
Prof. Dr. Ahmet ARSLAN
Prof. Dr. Novruz ALLAHVERDİ
Assoc. Prof. Dr. Harun UĞUZ

Imperceptibility, capacity and security are main requirements of information hiding systems. There is a trade-off between these requirements. Here, imperceptibility and capacity were handled, security was supported. Namely, novelty and contribution are increasing the amount of hidden data while protecting imperceptibility and providing security. Imperceptibility was provided by constituting stego-cover as forward mail platform and employing it for communication. Secret information was hidden by benefitting chosen text from the previously constructed text base consisting of naturally generated texts which are suitable for group speech. Hiding operation was performed by protecting the originality of cover text. Security was provided by employing data compression to complicate extraction procedure and combinatorics-based coding to provide randomness for an observer. Information loss isn't desired because of handling textual data. Therefore, LZW and Huffman lossless compression algorithms were chosen due to their frequent usages in the literature and significant compression ratios. The stego-key shared in the manner of symmetric encryption was employed to increase security. According to the experiments, for the secret message with 300 characters, capacity was computed as 8.37% and 9.34% for LZW and Huffman codings, respectively. Security was examined by formulating and measuring the combination number to extract secret message. Finally, comparison of the proposed method with the contemporary methods in the literature was carried out in terms of the most widespread requirement; capacity. Obtained capacity values are quite high and they increased as the character length increased. Thus, the disadvantage of length on capacity has been turned into an advantage. Moreover, language dependency is in minimum level, since operations are performed via numbers.

Keywords: Huffman coding, lossless data compression, LZW coding, steganography, text steganography

ÖNSÖZ

Günümüzde internet ve ağ teknolojilerinin hızlı gelişimi sonucu bilgi güvenliği oldukça önem arz etmektedir. Bilgi güvenliğinin alt dalı olan steganografi biliminin üstünlüğü ise gözlemcinin okumakta ya da bakmakta olduğu ortamda herhangi bir bilginin varlığını algılayamamasıdır. Steganografi biliminin uygulamaları tarihte sık sık karşımıza çıkmaktadır. Burada da ortak nokta şüphesiz saklama yoluyla iletilecek bilginin kritik önem taşımasıdır.

Özellikle metin steganografisi zor ve yeni bir dal olmasına karşın, uygulamada oldukça dinamik olmakta, farklı dillerle göre değişik uygulama yaklaşımlarına yer verebilmekte ve üzerinde beyin fırtınası gerçekleştirmeye gayet yatkın olmaktadır. Tüm bu anlatılanlar neticesinde ise ortaya uygulama bakımından basit ya da karmaşık ancak oldukça etkili sonuçlar veren yaklaşımların çıkması kaçınılmazdır.

Yeni ve uygulama bakımından daha iddialı olan bu alanda çalışmanın dönem dönem zorlukları olsa da, çok şey kazandırdığı görüşündeyim. Bu alanda çalışmam için bana fırsat veren tüm hocalarıma teşekkürlerimi sunmak isterim.

Öncelikle lisans eğitimimden bu yana bilgileri ve akademik görüşünden faydalandığım, kendime örnek edindiğim ve özellikle tez çalışması esnasında yaptığı paha biçilmez bilimsel katkıları ve yönlendirmelerinden ötürü sayın hocam Prof. Dr. İnan GÜLER'e teşekkürü borç bilirim.

Doktora tez danışmanlığımı üstlenerek; çalışmaların yürütülmesi sırasında ilgi ve desteğini esirgemeyen sayın hocam Prof. Dr. Hakan IŞIK'a teşekkürlerimi sunarım.

Uygulamalı çalışmalarımda yardımlarını esirgemeyen Prof. Dr. Ahmet ARSLAN, Yrd. Doç. Dr. Rıdvan SARAÇOĞLU ve matematiksel alandaki katkılarından ötürü Araştırma Görevlisi Bahar SAYIN, matematik öğretmeni Nurhan KENDİRLİ'ye teşekkür ederim.

Son olarak bu günlere gelmemde büyük pay sahibi olan aileme ve dostlarıma teşekkürlerimi sunarım.

Esra ŞATIR
KONYA-2013

İÇİNDEKİLER

SİMGELER VE KISALTMALAR

Simgeler

n	: Eleman sayısı
a	: Eleman
n!	: Farklı permütasyon sayısı
m	: Farklı eleman sayısı
w	: Tekrar sayısı
P	: Farklı permütasyon sayısı
Ψ	: n' ye kadar olan pozitif tamsayılar kümesi
D	: Kaynak bilgisi
Δ(D)	: Kaynak bilgisi sıkıştırılarak oluşturulan içerik
C	: Sıkıştırma oranı
So	: Orijinal dosya boyutu,
Sc	: Sıkıştırılan dosya boyutu
R	: Orijinal veri miktarındaki azalma
P	: Örüntü
T	: Metin
x	: Örüntü kümesi
Z	: T metninden elde edilen sıkıştırılmış dizi
P	: Kod kelimesi
C	: Veri dizisindeki bir sonraki karakter
a_i	: Harf
$P[a_i]$	: a_i harfinin ilgili örüntü içindeki olasılığı
l_i	: a_i için üretilen kod kelimesi içindeki bit sayısıdır
S	: Gizli mesaj
T	: Metin Tabanı
Text	: Metin Tabanındaki bir metin
$\overrightarrow{\Delta D}$	: Göreli uzaklık vektörü
A	: E - posta adres uzantı kümesi
$\vec{F}$	: Huffman kodlama frekansları
NT	: Metin tabanındaki metin sayısı
D	: Göreli uzaklık matrisi
E	: Taşma matrisi
R	: Yeniden yapılandırılan göreli uzaklık matrisi
K_1	: Global Stego Anahtar
K_2	: Seçilen ve stego anahtar olarak düzenlenen e-posta adres kümesi
MaxC	: Maksimum karakter sayısı
a	: Gizli mesajın karakterleri
b	: Metin tabanındaki her bir metnin karakteri
c	: $\overrightarrow{\Delta D}$ elemanları
d	: D matrisinin elemanı
e	: E matrisinin elemanı
r	: R matrisinin elemanı
P	: İkili örüntü vektörü
p	: P vektörünün elemanı
T*	: Seçilen örten metin

$\overrightarrow{R'}$: T*; seçilen örten metne karşılık gelen vektör - yeniden yapılandırılan $\overrightarrow{\Delta D}$

$\overrightarrow{R'}$: $\overrightarrow{R'}$ vektörünün sıkıştırılması ile elde edilen vektör

$(\overrightarrow{R'})_2$: $\overrightarrow{R'}$ vektörünün ikili tabandaki karşılığı

G_1 : İlk 9 bit

G_2 : Son 3 bit

x : G_1 bit dizisinin onluk tabandaki karşılığının 26' ya tam bölümü sonucu bulunan değer

y : G_1 bit dizisinin onluk tabandaki karşılığının 26' ya göre modu

z : G_2 bit dizisinin onluk tabandaki karşılığı

$\vec{E}$: Taşma vektörü

s : Kümenin eleman sayısı

C : Kapasite

N : E-posta adresi sayısı

m : E-posta adresindeki rakam sayısı

Π : Çarpım sembolü

LZW_C : LZW kodlaması sonucu elde edilen karmaşıklık değeri

H_C : Huffman kodlaması sonucu elde edilen karmaşıklık değeri

$MaxLZW_C$: LZW kodlaması sonucu elde edilen maksimum karmaşıklık değeri

$MaxH_C$: Huffman kodlaması sonucu elde edilen maksimum karmaşıklık değeri

Kısaltmalar

AES : Advanced Encryption Standard - Gelişmiş Şifreleme Standardı

ASCII : American Standard Code For Information Interchange- Bilgi Değişimi Amaçlı Amerikan Standart Kodlama Sistemi

C# : C Sharp

DASH : Dot and Arrow Attack – Nokta ve Ok İşareti Saldırısı

GB : Giga Byte

GHz : Giga Hertz

HTML : Hyper Text Markup Language – Hareketli Metin İşaretleme Dili

ISO : International Organization for Standardization

LZ : Lempel Ziv

LZW : Lempel Ziv Welch

MT : Machine Translation – Makine Çevirisi

OCR : Optical Character Recognition - Optik Karakter Tanıma

PDF : Portable Document Format-Taşınabilir Belge Formatı

RAM : Random Access Memory

RGB : Red Green Blue – Kırmızı Yeşil Mavi

Unicode : Evrensel kod

XML : Extensible Markup Language - Genişleyebilir İşaretleme Dili

XOR : Exclusive OR - Özel Veya

1. GİRİŞ

Ağ teknolojileri ve dijital cihazların artışı sayısal çoklu ortam iletimini hızlı ve kolay kılmıştır. Ancak, internet gibi genel kanallar üzerinden yapılan dijital bilgi dağıtımı; telif hakkı ihlali, sahtecilik ve dolandırıcılık gibi sebeplerden ötürü güvenli değildir. Bu nedenle, dijital veriyi, özellikle hassas veriyi korumak amacıyla geliştirilen metotlar oldukça önem kazanmaktadır (Chang ve Kieu, 2010). Elektronik dokümanların kullanımı yaygın olmasına rağmen, çok az kişi bu dokümanların gizli veri içerdiğini fark edebilmektedir. Burada "gizli" kelimesinin kullanım amacı, bu verinin normal bir şekilde bir dosyanın içerisine yerleştirilmiş olmasına karşın belli başlı metotlar kullanılmadan fark edilememesidir. Gizli veri iki türde sınıflandırılabilir. Birinci türde, gizli veri uygulama tarafından otomatik olarak oluşturulmakta, ikinci türde ise bir birey tarafından belli bir amaç doğrultusunda saklanmaktadır (Park ve Lee, 2009).

Geleneksel olarak, gizli veri kriptolojik metotlar ile korunabilmektedir. Ancak şifrelenmiş verinin kripto sistem aracılığı ile iletimi bazı devletler tarafından yasaklanmaktadır veya şifrelenmiş verinin anlamsız şekli ve görünümü herhangi bir gizli iletişimi durdurmak amacıyla tasarlanan önleyicilerin (sensörler gibi) dikkatini çekebilir (Chang ve Kieu, 2010). Alternatif olarak gizli veri, bilgi saklama teknikleri kullanılarak korunabilir. Genellikle bilgi saklama teknikleri filigranlama ve steganografiyi içermektedir (Chang ve Kieu, 2010). Filigranlama, esas amacı bakımından steganografiden farklıdır. Filigranlama, telif hakkı korunumu, yayın takibi, işlem izleme gibi aktiviteler için kullanılmaktadır. Bir filigranlama metodu, bir ortamı, bu ortamla ilgili bir bilgiyi (sahiplik bilgisi, kimlik vb.) gömmek amacıyla algılanabilir ya da algılanamaz şekilde değiştirir (Gutub ve Fattani, 2007).

Bu bölümün devamında kriptografi ve steganografi arasındaki farklar, filigranlama ve steganografi arasındaki farklar ve gerçekleştirilen tez çalışmasının amacı açıklanacaktır.

1.1. Kriptografi ve Steganografi Arasındaki Farklar

Eski Yunancada gizlenmiş yazı anlamına gelen steganografi, bilginin görünürlüğünü gizleme bilimine verilen isimdir. Günümüzde karşılaşılan en büyük yanlış anlama steganografinin kriptografi ile karıştırılmasıdır. Veriyi gizleme sanatı olarak bilinen bu bilimin kriptografiye göre en büyük üstünlüğü bilgiyi gören bir kimsenin gördüğü şeyin içinde önemli bir bilgi olduğunu fark edemiyor olmasıdır ve böylece görülen kısmın içinde bir bilgi aramaz. Oysa bir şifreli mesaj, çözmesi zor olsa bile gizemi dolayısıyla ilgi çeker çünkü bir bilginin gizlendiği bellidir. Günümüzde steganografi bilimi sayesinde ses, video, resim dosyalarına ve haberleşme kanallarına istenilen veri gizlenebilmektedir.

Steganografi, veriyi gizleme ve zararsız taşıyıcılarla veriyi taşıma yöntemidir. Bu yöntem, var olan veriyi gizlemede birçok gizli haberleşme tekniği kullanmaktadır. Steganografi eski bir el sanatı olmasına rağmen günümüzde bilgisayar teknolojisiyle yeni bir içerik kazanmıştır. Bilgisayar tabanlı steganografi metotlarıyla yeni gizleme teknikleri geliştirilmiştir.

Bilgiler metin formunda, sayısal 1 ve 0 olarak ya da başka çeşitlerde iletime geçerken verinin kime ait olduğunu belirten bir çeşit parmak izi bırakırlar. Steganografi bir çeşit kriptografi yöntemi gibi düşünülebilir. Her ikisi de haberleşme sırasında kaydedilmiş veriye bilgi ekleyerek çalışırlar. Kriptografi teknikleri bilgiyi belirli algoritmalara dayanarak şifreleyip güvenli bir örtü yaratmayı amaçlar. Steganografi ise kriptografiden farklı olarak veriyi örterek gizlemeyi sağlar. Kriptografide şifreli metin olarak adlandırdığımız örtülü yapı dikkat çekebilirken steganografide bu yapı kendini gizlediğinden dikkat çekmemeyi sağlar. Bu da verinin güvenli bir şekilde taşınması açısından önemli ve yararlı bir durum oluşturmaktadır (Elci ve ark., 2008).

1.2. Filigranlama ve Steganografi Arasındaki Farklar

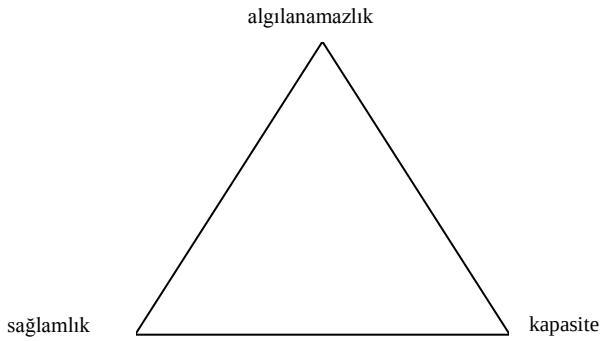
Filigranlama, steganografi ile yakından alakalıdır ancak bu ikisi arasında bazı farklar da bulunmaktadır. Filigranlama esasen ortamın kimliklendirilmesiyle uğraşırken steganografi, veriyi gizleme ile uğraşmaktadır. Gömülü filigran mesajları genellikle ortama ilişkin bilgiyle (telif hakkı gibi) ilişkilidirler ve bu nedenle ortam ile sınırlandırılmışlardır. Steganografide gizli mesajlar genellikle ortam ile alakalı

değildirler. Steganografik teknikler, aşırı derecede önemli bir bilgiyi herhangi bir engelleyiciye karşı fark edilemez bir hale getirmek için tasarlanmaktadırlar.

Filigranlama tekniğinde gömülü bilgi, taşıyıcının herhangi bir niteliği ile alakalı olabilmektedir ve taşıyıcıya ilişkin ekstra bilgi ya da özellik iletir. İletişim kanalının başlıca nesnesi, taşıyıcının kendisidir. Steganografide genellikle gömülü bilginin, bilgiyi geçirmek amacıyla bir mekanizma olarak basitçe kullanılan taşıyıcıyla gerçekleştireceği herhangi bir etkileşim yoktur. Burada iletişim kanalının başlıca nesnesi gizli bilgidir. Filigranlama uygulaması olarak, ortamın algı kalitesi ve sağlamlığı arasındaki denge korunmaktadır. Ortam kalitesini korumadaki kısıtlamalar, gömülü bilgi kapasitesini azaltmaya meyillidirler. Steganografinin uygulaması farklı olduğundan, gizli bilgi transferi ve gömme kapasitesi de sağlamlık ve ortam kalitesi kadar önemli görülmektedir (Shih, 2005)

1.3. Çalışmanın Amacı

Bir veri gizleme sisteminin en önemli gereksinimleri algılanamazlık, sağlamlık ve kapasite olarak bilinmektedir. Şekil 1.1'de gösterildiği gibi bu gereksinimlerin her biri, bir veri gizleme sistemindeki sihirli üçgenin köşelerini temsil etmektedir ve bu çakışan gereksinimler arasında her zaman bir ödünleşim mevcuttur (Zaker ve Hamzeh, 2011).



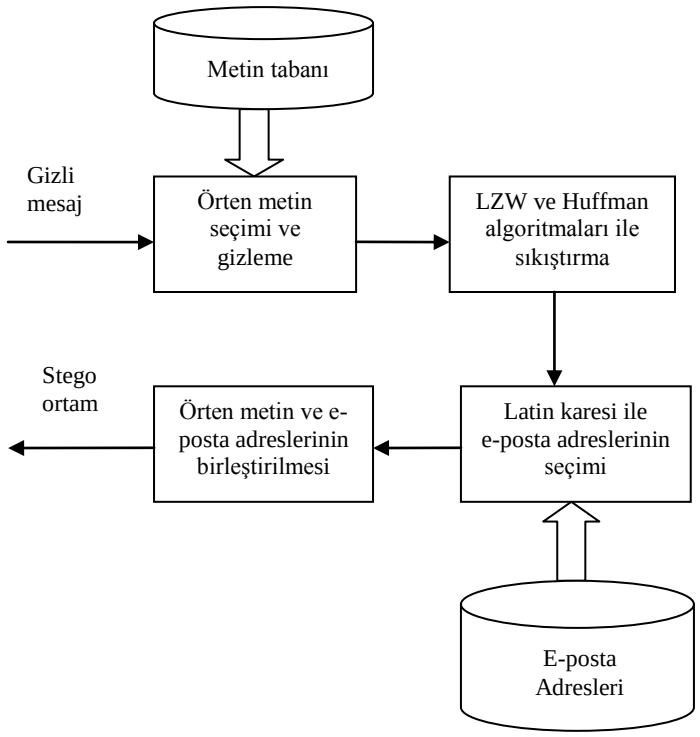
Şekil 1.1. Veri gizleme sistemindeki sihirli üçgen (Zaker ve Hamzeh, 2011)

Kapasite, örten ortama gömülebilen verinin bit miktarını ifade etmektedir. Güvenlik, bir gözlemcinin saklı bilgiyi çıkarma becerisiyle ilgilidir. Sağlamlık ise saklı

bilgiyi modifiye etme veya yok etmeye karşı direnme imkânı ile alakalıdır (Gutub ve Fattani, 2007).

Metinsel dokümanların karakteristik özelliklerini ele alırsak, burada bilgi gizleme iki gereksinimi karşılamalıdır. Birincisi; metinsel bilginin okunabilirliği gizleme işleminden etkilenmemeli, ikincisi ise, içerikte görsel bakımdan herhangi bir anormalliğe yer verilmemesidir (Shu ve ark., 2011). Bu çalışmada, sıkıştırma tabanlı bir metin steganografi yaklaşımı önerilerek kapasite ve güvenlik problemleri ele alınmıştır. Gutub ve Fattani'ye göre kapasite, örten ortama gizlenebilen verinin bit miktarını ifade etmektedir. Güvenlik, bir gözlemcinin gizli bilgiyi kolaylıkla tespit edip edemediği ile alakalıdır (Gutub ve Fattani, 2007). Yani, bu çalışmanın amacı, gizlenmiş verinin çıkarım işlemini karmaşıklaştırırken, örten ortama gizlenen veri miktarını artırabilmektir. Bu amaçla veri, önceden oluşturulmuş metin tabanından seçilen bir metin içerisine gizlenmektedir. Bu metin tabanı, hatırlatma, bildiri metinleri, makale özetleri gibi grup hitaplarında kullanılabilecek metinleri içermektedir.

Gömme esnasında, gizli bilgi sadece kamufle edilerek seçilen metnin orijinalliği korunmaktadır. İletişim iki taraf arasında gerçekleştiğinden, iletişim kanalı olarak e-posta seçilmiştir. Bu nedenle stego ortam bir forward mail platformu şeklinde düzenlenmiştir. Stego ortam forward mail platformu olarak düzenlenirken, önceden oluşturulmuş e-posta adres listesinden yararlanılmıştır. Şekil 1.2'de önerilen metodun blok diyagramı gösterilmektedir.



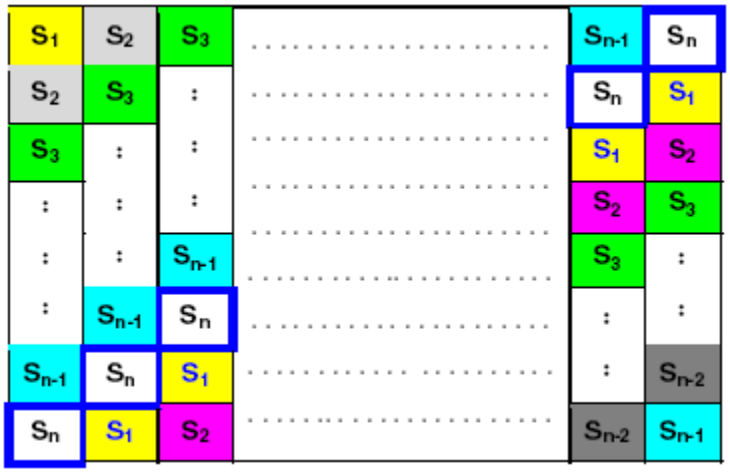
Şekil 1.2. Önerilen metodun blok diyagramı

Kapasite artışı olan ilk amaç için, veri sıkıştırma tekniklerinin kullanılması tercih edilmiştir. Bir veri sıkıştırma işleminde amaç, verilen veri tanımlamasındaki fazlalığı azaltmaktır (Galambos ve Bekesi, 2002). Sıkıştırma, iki bileşenin kombinasyonudur. Birisi kodlama, diğer birisi ise kod çözme algoritmasıdır. Kodlama algoritması, mesajın sıkıştırılmasını sağlamaktadır. Kod çözme algoritması ise sıkıştırılmış mesajdan, mesajın orijinal ya da orijinale yakın bir tahminin çıkarılmasını sağlamaktadır. Sıkıştırma algoritmaları; kayıplı ya da kayıpsız olarak iki sınıfa ayrılmaktadır (Begum ve Venkataramani, 2012). Kayıpsız veri sıkıştırma, orijinal ve çözülmüş dosyaların aynı olması gerektiği zaman kullanılmakta ve orijinal veri setinden, çözme işlemi sonucunda orijinal verinin birebir elde edilmesinin mümkün olduğu bir dönüşümü içermektedir. Kayıplı veri sıkıştırma ise orijinal veri setinden, çözme işlemi sonucunda orijinal verinin birebir elde edilmesinin mümkün olmadığı ancak yakın bir temsilin yapılabildiği bir dönüşümü içermektedir (Al-Bahadili, 2008). Metinsel bilgi ile çalışılması durumunda, sıkıştırma veya çözme işlemi gerçekleştirilirken orijinal verinin tamamı korunmalıdır. Resim ya da ses verisi ile çalışılması durumunda ise, çok büyük bir probleme uğramadan, orijinal bilgiye yakın bir tahmine izin verilebilir (Galambos ve Bekesi, 2002). Problemimizde, metinsel veri ile çalışıldığı için orijinalliğin korunması

gerekmektedir. Dolayısıyla, kayıpsız veri sıkıştırma teknikleri tercih edilmelidir. Bu nedenle, literatürde sıkça kullanımları ve verimli sıkıştırma oranlarından ötürü LZW (Lempel Ziv Welch) ve Huffman sıkıştırma algoritmalarından yararlanılmıştır. Huffman kodu, olasılıklar kümesinden elde edilen optimal bir ön koddur. Huffman kodlamasında, önce bir olasılık modeli elde etmek için kaynak veri taranır, daha sonra elde edilen bu olasılık modeli kullanılarak bir kodlama ağacı oluşturulur. LZW algoritması ilk önce veriyi okur. Sonrasında sözlükten kodlanmış bir karakter dizisi ve mümkün olan en geniş veri bitleri serisiyle eşleşen bir seri bulmaya çalışır. Eşleşen veri serisi ve bir sonraki karakteri bir arada gruplandırılarak, daha sonraki veri serilerinin kodlanması amacıyla sözlüğe eklenir (Liang ve ark., 2008).

Güvenliğin geliştirilmesi olan ikinci amaç için, stego anahtar kullanımını önerilmiştir. Bu çalışmada kullanılan stego anahtarlar görevlerine göre iki sınıfa ayrılabilir. Birisi, önerilen metodun gömme aşamasında oluşturulan stego anahtarlar, öteki ise tüm işlemlerden önce sadece alıcı ve gönderici arasında paylaşılan önceden oluşturulmuş global stego anahtarlardır. Ayrıca, kombinatorik tabanlı kodlama (detaylı bilgi için bkz Jun ve ark., 2011) kullanılarak arzu edilen rastgelesellik sağlanmış ve güvenliğe katkıda bulunulmuştur. Kombinatorik tabanlı kodlama, alıcı için yorumlanabilir olmaktadır. Ancak, stego ortamı analiz etmeye çalışan bir gözlemci için oldukça rastgelesel görünmektedir. Bununla birlikte kombinatorik tabanlı kodlama steganografik ortamı daha dirençli kılmaktadır (Desoky, 2009). Tüm bu amaçlar doğrultusunda Latin karesi kullanılmıştır. Ayrıca Bailey ve Curran, tarafından 2006 yılında gerçekleştirilen çalışmaya dayanılarak LZW ve Huffman sıkıştırma algoritmalarının da güvenliğe katkı sağladığı söylenilebilir.

Bir Latin karesi, $n \times n$ bir matristir ve n adet bağımsız sembolden oluşmaktadır. Her bir sembol verilen bir satır ya da sütunda yalnızca bir defa görülmektedir. Şekil 1.3'te örnek bir gösterim görülmektedir. Satırların S_1'den itibaren başlamasının zorunlu olmadığına dikkat ediniz. Başka bir deyişle satırların başlangıç sembolü değiştirilebilmektedir (Desoky, 2009).



Şekil 1.3. n × n Latin karesinde, her bir satır ya da sütun n adet sembolün bağımsız permütasyonudur (Desoky, 2009).

Önerilen çalışmada Latin karesi, gömme aşamasında her bir sayıyı bir harfe eşleme, çıkarım aşamasında ise her bir harfi bir sayıya eşleme suretiyle kullanılmıştır. İngiliz alfabesine göre oluşturulan Latin karesi (bkz. Ek-1) 26 farklı dizilim içermektedir. Böylece ard arda aynı sayısal ya da harf örüntülerinin kullanılması durumunda bile farklı harf karşılıkları elde edilmektedir. Bu yolla gözlemci için arzu edilen rastgelesellik sağlanmaktadır.

Önerilen metotta gizli mesaj, seçilen örten metnin orijinalliği bozulmadan (örten metnin formatı ya da anlamı değiştirilmeden) saklanmaktadır. Bu saklama sonucu sayısal bir dizi elde edilmektedir. Bu sayısal dizinin elemanları Latin karesi ile harflere çevrilmekte ve bu harfler ile e-posta adresi seçimi gerçekleştirilmektedir. E-posta servis sağlayıcılarının dünya genelinde ISO (International Organization for Standardization) standartlarına göre temel Latin alfabesini desteklediği unutulmamalıdır. Yani dünya genelinde, bir e-posta adresinde bu alfabe dışındaki harfler kullanılamamaktadır. Bu nedenle Latin karesi 26×26'lık bir matris şeklinde oluşturulmuştur.

Gizlenecek mesajın diline bağlı kalınmaksızın gizleme işlemi sonucu sayısal bir dizi elde edilmektedir. Bu aşamadan sonraki tüm işlemler sayılar ile gerçekleştirildiğinden ötürü önerilen metotta dile bağımlılık asgari düzeydedir. Eğer saklanacak mesaj Türkçe ise, metin tabanı da Türkçe olmalıdır, İngilizce ise İngilizce olmalıdır, Çince ise Çince olmalıdır vb. Aksi halde örten metin ve gizlenecek mesaj arasında harf eşlemesi bulunamayacak ve başlangıçta gerekli olan sayısal dizi oluşturulamayacaktır.

Değerlendirme işlemi kapasite ve güvenlik ölçümleri gerçekleştirilerek yapılmıştır. Kapasite, stego ortama gömülen bit miktarının yüzde cinsinden hesaplanmasıyla ölçülmüştür. Güvenlik ise, algoritmanın herkesçe bilindiği varsayılarak, gizli mesajın çıkarılması için mümkün olan kombinasyon sayısı hesaplanarak ölçülmüştür. Son olarak, kapasite açısından önerilen metot, literatürdeki diğer metotlarla karşılaştırılarak, genel bir değerlendirme gerçekleştirilmiştir.

Bu tez çalışması şu şekilde organize edilmiştir: İkinci bölümde, literatürdeki mevcut metin steganografi metotları anlatılmıştır. Önerilen metot için kullanılan yöntem ve materyaller ile birlikte metodun kendisi üçüncü bölümde açıklanmıştır. Kapasite ve güvenlik analizleri gerçekleştirilerek elde edilen deneysel bulgular ve araştırma sonuçlarına dördüncü bölümde yer verilmiştir. Son olarak ulaşılan sonuçlar, çalışmada varılan nokta ve öneriler beşinci bölümde vurgulanmıştır.

2. KAYNAK ARAŞTIRMASI

Önceki bölümde anlatıldığı gibi steganografi, veriyi diğer bir verinin içine gizleyerek veya gömerek görünmez yapmaktadır. Veriyi gizleme amacıyla kullanılan bu diğer veri parçasına örten ortam ya da taşıyıcı denmektedir. Gizlenmiş veriyi içeren bu örten ortama ise stego nesne denilmektedir. Bu stego nesnesi, saklanabilmekte ya da iletilebilmektedir. Gizli veri değişik çeşitlerde örten ortamlara gömülebilmektedir. Veri bir metin dosyasına gömülmüş ise sonuçta oluşan nesne stego metin veya örten metin şeklinde adlandırılmaktadır. Bu sebeple örten resim, stego resim, örten ses, stego ses, örten video, stego video vb. isimlendirmeler mümkün olmaktadır (Salomon, 2005). Bu terminoloji, Birinci Uluslararası Bilgi Gizleme Semineri'nde kabul görmüştür (Pfitzmann, 1996) ve ilerleyen bölümlerde bu terminoloji kullanılacaktır.

İngilizce, Çince, Arapça gibi farklı dillerde metin steganografi metotları tasarlamak için birçok girişimde bulunulmuştur. Bu bölümde, metin steganografisi alanındaki bu çalışmalar açıklanmıştır.

Başlangıç olarak, ikinci dünya savaşında Alman bir casus tarafından gönderilen aşağıdaki mesajı inceleyelim:

"Apparently neutral's protest is thoroughly discounted and ignored. Isman hard hit. Blockade issue affects pretext for embargo on by-products, ejecting susets and vegetable oils."

Daha öncede vurgulandığı gibi, steganografi sonucunda üretilen stego metin, gözlemci takibine karşı şüphe uyandırmayacak biçimde gizli mesajı içerebilmelidir. Yukarıdaki stego metinde her bir kelimenin ikinci harfini çıkararak gizli mesaj çözülebilmektedir:

"Pershing sails from NY June 1" (Yeh ve Hwang, 2001).

Wayner (1992, 2002) mimic fonksiyon yaklaşımını önermiştir. Bu metotta, rastgele dağıtılmış bitlerden oluşan bir veri dizisi giriş olarak ele alınmakta ve Huffman kodunun tersi uygulanmaktadır. Bu işlemin amacı, normal bir metnin istatistiksel profiline uyan bir stego metin üretmektir. Bu nedenle, mimic fonksiyonları tarafından üretilen stego metin, istatistiksel saldırılara karşı dirençli olmaktadır. Mimic

fonksiyonları, çıkışı geliştirmek için bağlamdan bağımsız gramer kuralları ve van Wijnaarden gramer kurallarını kullanabilmektedir. Aslında sıradan bir mimic fonksiyonundan alınan çıkış anlaşılmazdır. Dolayısıyla, bu durum stego metni oldukça şüpheli göstermektedir. Ancak mimic fonksiyon ve bağlamdan bağımsız gramerin bir arada kullanılması metnin okunabilirliğini kısmen artırmaktadır. Ama halen, stego metin yanlış sözdizimi, yanlış retorik ve hatalı gramer gibi birçok eksiklik içermektedir. Ayrıca, bu metnin içeriği çoğunlukla anlam bakımından tutarsızdır ya da anlamsızdır. Bu noksanlıklar, gizli iletişim esnasında şüphe uyandırabilmektedir (Desoky, 2009).

1995 yılında Maher, Texto adı verilen bir veri gizleme programı önermiştir. Bu metot, ikili ve özellikle şifrelenmiş veri değiş tokuşu için oldukça uygun bir metottur. Burada gizli veri İngilizce kelimeler ile değiştirilmektedir. Yani Texto basit bit yerdeğiştirme ile (ortanım) kriptolama şeklinde çalışmaktadır. Sadece isim, fiil, sıfat ve zarflar önceki cümle yapılarını tamamlamak için kullanılmakta ve bu kelimeler nihai metinde oldukça belirgin olmaktadırlar. Ancak, kısaltma veya bağlaç gibi kelimeler ile sözlükte geçmeyen kelimeler ihmal edilmektedir (Wang ve ark., 2009a). Bu metottaki eksikliklerden ilki, oluşturulan örten metnin algısal olarak mantıklı görünmemesi, ikincisi ise metnin anlamsal bütünlüğünün olmamasıdır. Bu nedenle, taraflar arasındaki iletişimi analiz etmeye çalışan bir gözlemci için bu, şüphe uyandırıcı bir durum olmaktadır.

Chapman ve Davida, Nicetext ve Scramble adında iki fonksiyondan oluşan steganografik metot önermişlerdir. Nicetext, bir mesajı eşanlam yer değiştirmesi şeklinde gömmek için metnin bir kısmını kullanmaktadır (Desoky, 2009). Eşanlam tabanlı yaklaşım, son on yılda Winstein (Wintesin, 1999), Nakagawa (Nakagawa ve ark., 2001) ve Murphy (Murphy ve Vogel, 2007) gibi birçok araştırmacının dikkatini çekmiştir. Eşanlam tabanlı yer değiştirmede örten metnin anlamı korunmaktadır. Eşanlamlı kelimelerin seçimi uygun bir şekilde gerçekleştirildiği takdirde, stego metin dil bilgisi bakımından mantıklı olarak algılanabilir. Ancak Desoky tarafından 2009 yılında gerçekleştirilen çalışmada da vurgulandığı gibi bir mesajı gizlemek için aynı metni farklı eşanlamlı kelimeler içerecek şekilde tekrar tekrar kullanmak şüphe uyandırabilmektedir.

2004 yılında, Sun ve ark. Çince karakterlerin sağ ve sol bileşenlerini kullanan bir metot önermişlerdir. Önerilen bu metot L-R metodu adını almaktadır. L-R metodunda tüm Çince karakterlerin matematiksel ifadeleri metin gizleme stratejisine verilmektedir. Bilgi gizleme amacıyla sol ve sağ bileşeni mevcut karakterler seçilmektedir. Gömme

aşamasında, gizlenecek bilginin biti 0 ise karakterin orijinal görünümü korunmakta, 1 ise karakterin görünümü, sağ ve sol bileşenleri arasındaki boşluk ayarlanarak modifiye edilmektedir. Ancak bu metodun bazı eksiklikleri mevcuttur. Birincisi ters çevrilebilir olmamasıdır. Gizlenmiş bilgi çıkarılmış olsa bile alıcıya aynı örten metni tekrar kullanma imkânı verilmemektedir. İkincisi ise çıkarım işlemidir. Çıkarım aşamasında L-R metodunda, iki komşu karakterde bilgi gizlenip gizlenmediğini tespit etmek amacıyla bu iki karakter arasındaki genişlik ve boşluk hesaplamalıdır. Bu nedenle örten metin, çıkarım aşaması esnasında hesaplama işleminin gerçekleştirilebilmesi için metin dosyası yerine görüntü dosyası biçiminde tutulmalıdır. Ancak, bir görüntü dosyası bir metin dosyasından hem iletim amacıyla daha fazla bant genişliği gerektirmekte hem de alıcı tarafında ise saklama amacıyla daha fazla yer gerektirmektedir. Son olarak, L-R metodunda Çince karakterlerin yukarı ve aşağı bileşenleri dikkate alınmamaktadır. Bu nedenle L-R metodunun bilgi gizleme kapasitesi sınırlıdır. Bu metodun eksikliklerini gidermek ve kapasitesini artırmak amacıyla Wang ve ark. tarafından 2009 yılında metot yeniden düzenlenmiştir. Geliştirilen bu yeni metotta, Sun ve ark.'na ait olan L-R metodunun tüm matematiksel ifadeleri aynen devralınmakta ve ekstra bir aday küme olarak bu Çince karakterlerin yukarı ve aşağı bileşenleri de eklenmektedir. Ayrıca önerilen metotta, tersine çevrilebilir bir fonksiyon eklenmiş ve veri çıkarımı için basit bir strateji tasarlanmıştır. Bu geliştirmeler ile önerilen metot, alıcının stego metin dosyasından gizlenmiş veriyi kolayca çıkarabilmesine imkân sağlamakta ve orijinal metin dosyasının eşzamanlı olarak elde edebilmesine olanak vermektedir. Bu çıkarılan örten metin dosyası ise sonraki gizli iletişimlerde tekrar tekrar kullanılabilmektedir. Ayrıca, çıkarım aşamasındaki görüntü dosyası kullanımı ortadan kaldırılarak çıkarım stratejisi de basitleştirilmiştir. Ancak metin dosyalarının bütünlüğünü korumak amacıyla önerilen metodun uygulama alanının metin filigranlamasını da içermesi gerekmektedir (Wang ve ark., 2009a). Sınırlı kapasiteleri yanında bu iki metodun asıl ve ortak dezavantajı sadece Çince'ye uygulanabilir olmasıdır.

2007 yılında Shirali - Shahreza ve Shirali – Shahreza en basit haliyle, arada başka dönüşüm metotları uygulanmaksızın, ASCII (American Standard Code For Information Interchange- Bilgi Değişimi Amaçlı Amerikan Standart Kodlama Sistemi) kodundan ikilik tabana çevrilerek elde edilen gizli mesaj bitlerini saklamak için kelimeleri, kısaltmaları ile yer değiştirmeye dayalı bir metot önermişlerdir (218 – too late, C-see vb). Kısaltma ve kelime olarak farklı şekillerde yazılan bu ifadeler listelenmiş ve ayrı sütunlarda toplanmışlardır. Örneğin, kısaltmaların bulunduğu sütun 1

ile etiketlenirken, kelimelerin bulunduğu sütun 0 ile etiketlenmektedir. Önce, saklanacak mesaj, bitlerine ayrılmaktadır. Mesaj, önceden düzenlenen bu listeye uyan yazım farkına sahip kelimeleri bulmak amacıyla iteratif olarak aranır. Eşleşen bir kelime veya kısaltma bulunduğunda saklanacak bit kontrol edilerek kelime - kısaltma listesindeki 0 ve 1 sütunlarının hangisi altında olduğuna bakılmaktadır. Bu bitin değeri (0 veya 1) dikkate alınarak örten metinde, ilgili sütun etiketine göre kelime - kısaltma yer değiştirmesi yapılmaktadır. Aksi halde kelime - kısaltma değişimi gerçekleşmemektedir. Bu işlem gizlenecek mesajın sonuna kadar tekrarlanmaktadır. Esnek ve hızlı bir metot olmasına rağmen güvenliği zayıftır. Algoritma bilindiği takdirde gizli mesaj kolaylıkla çıkarılabilmektedir (Rafat ve Sher, 2010).

2008 yılında Shirali - Shahreza gizli mesaj bitlerini saklamak amacıyla, İngiliz ve Amerikan İngilizcesindeki kelimelerin farklı yazımından faydalanan bir metot önermiştir (favourite – favorite, criticize – criticise vb.). İngiliz ve Amerikan İngilizcesinde farklı şekillerde yazılan bu kelimeler listelenmiş ve ayrı sütunlarda toplanmışlardır. İngiliz İngilizcesine göre yazılan kelimelerin bulunduğu sütun 1 ile etiketlenirken, Amerikan İngilizcesine göre yazılan kelimelerin bulunduğu sütun 0 ile etiketlenmektedir. Önce, saklanacak mesaj, bitlerine ayrılmaktadır. Mesaj, önceden düzenlenen bu listeye uyan yazım farkına sahip kelimeleri bulmak amacıyla iteratif olarak aranır. Uyan bir kelime bulunduğunda saklanacak bit değerine göre 1 ya da 0 sütunlarının hangisi altında olduğuna bakılarak kontrol edilir. Sonrasında İngiliz veya Amerikan İngilizcesine göre yazılan kelimelerin bulunduğu ilgili sütundaki kelime örten metin içerisine yerleştirilmektir. Tabloda listelenmeyen kelimeler ise değiştirilmeden bırakılmaktadır. Bu işlem gizli mesajın sonuna kadar devam etmektedir. Hızlı bir metot olmasına rağmen dile özgü olması ve zayıf güvenliği metodun dezavantajlarıdır (Rafat ve Sher, 2010).

Bennet ve ark. tarafından 2004 yılında gizli bilginin HTML (Hyper Text Markup Language - Hareketli-Metin İşaretleme Dili) etiketleri içine saklanmasına dayalı bir metot önerilmiştir. HTML etiketlerinde büyük küçük harf duyarlılığı bulunmamaktadır. Örneğin <p align="center">, <p align="cenTER">, <p align="Center"> ve <p aLigN="center"> etiketlerinin hepsi geçerlidir ve aynı şekilde yorumlanmaktadır. HTML dokümanlarında steganografi, etiketlerdeki harfleri büyük ya da küçük harfler ile değiştirerek gerçekleştirilmektedir. Gizlenen bilgi ise dokümanın orijinal haliyle değiştirilmiş hali karşılaştırılarak çıkarılabilmektedir. HTML steganografisinde güvenlik, belli bir harf sırası fonksiyonu seçilerek artırılabilir. Örneğin çoğu etiketin

rastgele birkaç değiştirilmiş harfe sahip olduğu yerlerde etiketler içerisindeki üçüncü harf seçilerek gözlemci şaşırtılabilir (Gutub ve Fattani, 2007). Ancak metodun güvenliği zayıftır, algoritma bilindiği takdirde gizlenen bilgi çıkarılabilir.

2009 yılında Khairullah Microsoft Word dokümanlarında uygulanan yeni bir yaklaşım önermiştir. Ana fikir, boşluk ve satır başı gibi görünmeyen karakterler için herhangi bir ön plan rengi ayarlamaktır. Yani burada ilginç olan bulgulardan biri, boşluk, sekme, satır başı gibi karakterlerin yazı tipi renginin de ayarlanabilir olmasıdır. Çoğu kullanıcı veya gözlemci, bu görünmeyen karakterlerin renk değerleri ile ilgilenmez. Dolayısıyla her bir boşluk, sekme veya satır başı karakteri geçtiğinde, gizlenmiş bilginin ortaya çıkarılma riski olmadan, üç bayt saklanabilmektedir. Ayrıca bu yaklaşım istenen bitleri gizlemek için ekstra bilgi gerektirmemektedir. Yazı tipi renginin kullanıcı tarafından fark edilmesinin muhtemel olduğu yerde, rengin özelliği bunu engellemektedir. Bu nedenle gizli bilgi, görünmeyen karakterler içerisine RGB (Red Green Blue – Kırmızı Yeşil Mavi) değerleri şeklinde saklanabilmektedir. Örneğin aşağıdaki bit dizisi ele alınsın:

10101011010111010110001101010011001110100100111001010011100101010110001010

Bu bit dizisi sekizli gruplara ayrılsın. Gruplar normal ve kalın yazı tipinde gösterilmektedir:

1010110101110101**10001101**01001100**11101001**00111001**01001110**01010101**10001010**

Dokümanın ilk üç görünmeyen karakterinin RGB değerleri şu şekilde olmaktadır:

{173, 117, 141}, {76, 233, 57}, {78, 85, 138} (Khairullah, 2009).

İnsanların yaşamında sohbet odaları vasıtasıyla iletişim oldukça popüler bir hale geldiğinden Wang ve Chang 2009 yılında yeni bir metin steganografi metodu önermişlerdir. Önerilen metotta gizli bilgi, sohbet odalarında internet üzerinden iletişim esnasında yüz mimiklerini ifade eden küçük boyutlu resimler yani ikonlar (emoticon)

içerisine gömülmektedir. Bu metotta öncelikle, göndericinin ikon tablosunun alıcının ikon tablosuyla aynı olması gerekmektedir. Daha sonra, gönderici bu ikonları anlamlarına göre (gülümseme, gülme, ağlama vb.) farklı kümelere ayırmaktadır. Her bir ikon yalnızca bir kümeye ait olabilmektedir. Sıfırdan başlayarak bir ikonun kendi kümesindeki sıra numarası, gömülecek gizli bitleri göstermektedir. Bu nedenle önerilen steganografik metot, her bir kümedeki ikon sırasını kontrol etmek amacıyla gizli bir anahtar kullanmaktadır. Bu anahtar sadece alıcı ve gönderici tarafından tutulmaktadır. Birçok sohbet odasında kullanılan çok fazla sayıda ikon olduğundan bu metotla kapasite oldukça artırılsa da, bu artış büyük ölçüde önceden paylaşılan ikon tablosuna ve her bir kümedeki ikon sayısına bağlı olmaktadır (Wang ve ark., 2009b).

Grothoff ve ark. çeviri (tercüme) tabanlı bir steganografik metot geliştirmişlerdir. Bu metot bir mesajı saklamak için, makine çevirisinde doğal olarak görülen ve karşılaşılan hataları (gürültü) kullanmaktadır. Gizli mesaj, çoklu MT (Machine Translation – Makine Çevirisi) sistemlerinin çeviri çeşitliliğinden yararlanılarak çevirisi yapılan metin üzerinde yer değiştirme işlemi gerçekleştirilerek saklanmaktadır. Ayrıca burada kapasite (ya da bit oranı) artışı için, MT sistemlerinin yaygın hataları ve eşanlam yer değiştirmesi de kullanılmıştır. Eşanlam yer değiştirmesinin aksine gürültü tabanlı bu yaklaşımda dilsel hatalar çok fazla karşılaşılmadığı müddetçe rahatsız edici olmamaktadır. Ancak Grothoff ve ark. tarafından da vurgulandığı gibi MT sistemlerindeki süregelen gelişmeler bu alanda veri gizlemenin sınırlarını daraltmaktadır. Ayrıca çeviri tabanlı veri gizleme, temel yapısal farklılıklardan dolayı her dile de uygulanamamaktadır. Bu, ciddi anlamda mantıksız ve okunabilirlik açısından zayıf metinlerin üretimine sebep olmaktadır. Diğer bir metin tabanlı yaklaşım ise Topkara ve ark. tarafından 2007 yılında önerilmiştir. Burada, e-posta, forum vb. metinlerdeki yazım hataları ve dilbilgisine uygun olmayan kısaltmalar veri gizlemede kullanılmaktadır. Bu yaklaşımların eksikliği ise insan tarafından yazılan metinlerdeki hata ve gürültüye karşı çok duyarlı olmalarıdır. Bu, hem yaklaşımın saldırılara karşı savunmasızlığını artırırken hem de veri gizlemenin sınırlarını daraltmaktadır (Desoky, 2009).

2009 yılında Samphaiboon yeni bir steganografik metot geliştirmiştir. Bu metotta, gizli mesaj, televizyon ve web siteleri gibi medya ekranlarında kısa bir metin dizisi içerisinde çoklu alıcıya gönderilmektedir. Ancak burada, uygun OCR (Optical Character Recognition - Optik Karakter Tanıma) biriminin kod çözücüde hali hazırda bulunduğu varsayılmaktadır. Uygulama alanı olarak Tayland dili seçilmiştir ve gömme

aşamasında, etkili birkaç metinden bite dönüşüm metodu önerilmiştir. Öncelikle Tayland dilindeki kısa bir metni, çoklu gizli bitlere dönüştürmüşlerdir. Prensipte önerilen metot herhangi bir dildeki kısa bir metne uygulanabilmektedir. Deneysel bir değerlendirme sonucunda her bir kısa metne dört gizli metin bitinin gömülebildiğini göstermişlerdir. Ayrıca, kod çözücüde optik karakter tanıma birimi bulunmadan da gömülü bitlerin insan bir gözlemci tarafından doğru olarak kolayca çıkarılabildiği gösterilmiştir. Bu metodun ana avantajı, gizli mesajın aynı anda farklı yerlerdeki çoklu alıcılara yayınlanabilmesidir. Ancak yazar tarafından, göndericinin kısa metnin iletildiği kanal üzerinde kontrole sahip olduğu varsayılmaktadır. Ayrıca medya ekranında görüntülenen metni tanıyabilen ve bunu makine tarafından okunabilir formata doğru bir biçimde çeviren metinsel görüntü okuma biriminin var olduğu farz edilmektedir (Samphaiboon, 2009).

Desoky 2009 yılında Listega adında bir metot önermiştir. Bu metot, mesajları gizlemek için maddeler şeklinde oluşturulan metinsel listeleri kullanmanın avantajından faydalanmaktadır. Basitçe, mesaj kodlamakta ve daha sonra liste şeklinde örten bir metin üretmek amacıyla buna uygun ve mantıklı bir görünümde metinsel elemanlara atamaktadır. Listega metodu gizli mesajı ve iletimini, anlamlı elemanlardan oluşan bir liste ile kamufle etmeye dayandığı için akla yatkınlık ve mantıklılık elde etmektedir. Ayrıca bu yolla elde edilen liste şeklindeki bir steganografik ortam sözdizimsel ve mantıksal olarak da uygundur. Buna ilaveten iletişim esnasında arzu edilen rastgeleselliği sağlamak amacıyla kombinatorik tabanlı kodlama kullanılmıştır. Kombinatorik tabanlı kodlama alıcıya göre yorumlanabilir ancak gözlemciye göre oldukça rastgelesel olmaktadır (Desoky, 2009). Bu yaklaşım, metnin ne anlamını ne de formatını değiştirmeden metnin orijinalliğini korumasına ve bu yolla steganografik ortamı karşıtlık, çelişki, sözdizimsel ve istatistiksel vb. saldırılara karşı dirençli kılmasına rağmen, metodun çıkarım aşaması yeterince karmaşık değildir. Ayrıca veri gömme kapasitesi ise büyük ölçüde metinsel bir liste şeklinde stego ortam oluşturmak için seçilen elemanlara bağlıdır.

2010 yılında Lee ve Tsai, gizli mesajı PDF (Portable Document Format-Taşınabilir Belge Formatı) dosyalarına gizleyen yeni bir gizli iletişim metodu önermişlerdir. Mesajı gömmek amacıyla özel bir ASCII kodu olan A0 kullanılarak alternatif boşluk kodlama ve sıfır değer boşluk kodlama olarak iki veri kodlama çeşidi önerilmiştir. Bu metotta, bir mesaj karakter veya bit dizisi olarak ele alınmakta ve ikili ya da bütün şeklinde kodlama yapılarak özel bir ASCII kodu ile kodlanmaktadır. İki

teknikte de gizli mesaj bitleri kelimeler ya da karakterler arasına sırayla gömülmekte ve genel PDF okuyucu pencerelerinde görünmez hale gelmektedir. Karakter arası gömme için, A0 kodunun genişliği boşluk kodu olan 20 ile aynı olarak ayarlanmakta, karakter arası gömme için ise genişlik sıfır olarak ayarlanmaktadır. Böylelikle, steganografik bir etki oluşturmak sureti ile gizli iletişim gerçekleştirilmektedir (Lee ve Tsai, 2010). Önerilen metot bir çeşit boşluk kodlama tekniği olduğundan kapasitesi sınırlıdır ve çoğunlukla taşıyıcı olarak PDF dosyasındaki karakter sayısına bağlı olmaktadır. Bu metodun diğer bir dezavantajı ise güvenlik konusudur. Algoritmayı bilen bir gözlemci tarafından gömme işleminin tersi uygulanarak gizli mesaj tespit edilebilir.

2011 yılında Mir ve Hussain kriptografi ile birlikte bir steganografi metodu önermişlerdir ve XML (Extensible Markup Language - Genişleyebilir İşaretleme Dili) dosyalarında dokuz farklı gömme tekniği kullanmışlardır. Diğer bir güvenlik katmanı olarak eklenen AES (Advanced Encryption Standard - Gelişmiş Şifreleme Standardı) ile birlikte kullanılan dokuz metodun tümü için C# dilinden yararlanmışlardır. Uygulamada tüm gömme teknikleri farklı standartlara göre ölçülmüş ve alfabe dışı karakter yerleştirme metodu, renk yerleştirme metodu, satır sonu metodu, eş anlamlı metodu ve kısaltma metodunu savunulabilirlik açısından daha güçlü olarak analiz etmişlerdir. Sonrasında bu metotlar, metinsel bilgiye uygulanmıştır ve böylece XML dosyalarındaki diğer veri tiplerine de uyarlanabildiği görülmüştür. XML dosyaları sadece metinsel veri içermediğinden metodun uygulaması diğer kısımlara da genişletilebilmektedir (Ryabko ve Ryabko, 2011).

2011 yılında Shua ve ark. tarafından, taşıyıcı örten metinlerin bilgi gizleme şeklinin belirlediği alternatif bir çoklu metinsel bilgi gizleme algoritması önerilmiştir. Burada gizlenecek bilgi, saklanacak olan ve çoklu metin bölümleri halinde dağıtılan taşıyıcı türü ve miktarına bağlı olarak etkin bir biçimde gömülmektedir. Önerilen metotta, gizlenecek bilgi, taşıyıcı metin sayısına göre (2^n veya daha fazla) n defa XOR (Exclusive OR - Özel Veya*)* ayrıştırması ile 2^n parçaya ayrılmıştır. Sonrasında gizli bilginin her bir parçası taşıyıcı metinlerin kategorisi dikkate alınarak farklı metinlere gömülmektedir. Örneğin, taşıyıcı metin, matematik bilimi ve teknoloji hakkında olduğunda ve çok fazla matematiksel formül, noktalama işareti ve matematiksel kod içerdiğinde, kelime kategorileri arasına gömme işlemi uygulanmaktadır. Taşıyıcı metin literatür kategorisinde olduğunda, gizli bilgiyi gömmek için eş anlam yer değiştirmesi gibi taşıyıcı bilgisini ileten bir metot uygulanmaktadır. Taşıyıcı metin resimli bir metin olduğunda, metnin formatından yararlanan bir metot uygulanmaktadır. Diğer metin

kategorileri için ise, gizli bilgiyi gömmek için anahtar kullanımından yararlanılmaktadır. Burada, çoklu metin parçaları arasındaki sözü edilen ilişki, gizleme algoritmasının anahtarının bir kısmıdır. Genelde, bu ilişki yalnızca alıcı ve gönderici tarafından bilinmektedir ve bu nedenle yetkisi olmayan bir şahıs tarafından elde edilmesi oldukça güçtür (Shua ve ark., 2011). Gizli bilgiyi saklamak ve iletmek için farklı taşıyıcı metinlerin kullanımı ve taşıyıcı metnin kategorisine göre gömme işleminin belirlenmesi kuşkusuz stego ortamı gözlemcilere karşı daha dirençli kılmaktadır. Ancak anahtar kullanımı olmadığı takdirde algoritma bilindiğinde algoritmanın güvenliği sorgulanabilmektedir.

2012 yılında Por ve ark. boşluk karakterini kullanan UniSpaCh adında bir veri gizleme metodu önermişlerdir. Bu metotta bilgi Unicode (Evrensel kod) boşluk karakterleri kullanılarak Microsoft Word dokümanları içerisine gömülmektedir. Burada boşluklar, saklanacak veriyi kodlamak için ele alınmışlardır çünkü doküman içerisinde oldukça fazla yer almaktadırlar ve kullanımlarının doküman üzerinde algılanabilirlik açısından önemsiz bir etkisi vardır. UniSpaCh, veriyi gömmek için cümle arası, kelime arası, satır sonu ve paragraf arası boşlukların karışımını kullanmaktadır. Ancak, gömme kapasitesi artırılırken DASH (Dot and Arrow Attack – Nokta ve Ok İşareti Saldırısı) saldırısına karşı dirençli olmak için boşluk türüne bağlı olarak Unicode boşluk karakterlerinin farklı bir kümesi kullanılmıştır. Genellikle, Unicode Standart Versiyon 5.2'de 18 boşluk karakteri bulunmaktadır. Microsoft Word 2007 programında bulunan biçimlendirmeyi göster/gizle özelliği ile gerçekleştirilen basit bir tetkikten sonra, sadece 8 Unicode boşluk karakteri veri gizleme için uygun bulunmuştur. Bu bağlamda, DASH saldırısına göre algılanamaz (boşluğun varlığını gösteren herhangi bir işaret olmaksızın) olarak görünen bir boşluk uygun bulunmaktadır. Geri kalan boşluk karakterleri DASH saldırısı altında kaçınılmaz olarak kare ya da derece sembolü şeklinde açığa çıkmaktadır. Gömme işlemi için, gizli bilgiye göre sıradan boşluk ve Unicode boşluk karakterinin kombinasyonu kullanılmıştır. Önemli bir kapasite artışı sağlanmasına karşın bu metodun temel eksikliği çıkarım aşamasının yeterince karmaşık olmamasıdır. Yazarlarca da değinildiği gibi, sadece Unicode karakterlerini hedefleyen bir istatistiksel analiz gerçekleştirildiğinde gizli mesajın varlığı tespit edilebilmektedir (Por ve ark., 2012).

Bu tez çalışmasının anlatılanlardan farkı, kapasite artışı amacıyla veri sıkıştırma algoritmalarından yararlanılmasıdır. Ele alınan veri metinsel olduğu için kayıpsız veri sıkıştırma algoritmaları kullanılmıştır. Ayrıca Bailey ve Curran tarafından 2006 yılında

gerçekleştirilen çalışmaya dayanılarak sıkıştırma algoritmaları ile çıkarım aşaması da karmaşıklaştırılmış ve güvenliğin artırılması amaçlanmıştır. Başka bir deyişle bu çalışmanın amacı, kapasite bakımından önemli bir artış elde etmek iken, önerilen metodun güvenliği için çıkarım aşamasını karmaşıklaştırmaktır. Bu amaçlar doğrultusunda gizli veri, önceden oluşturulan metin tabanından seçilen bir metne gömülmektedir. Bu metin tabanı hatırlatma, makale özeti gibi toplu hitaplarda kullanılabilecek doğal olarak üretilen metinlerden oluşmaktadır. Böylece örten metin adayları olarak kullanılan bu metinler, anlamlı ve mantıklı olmalarının yanında doğru söz dizimi, gramer ve anlamsal bütünlüğe sahip olmaktadırlar. Metin tabanındaki her bir metnin uzunluğu kapasiteyi sınırlandırmamak için sabit tutulmuştur (en çok 3000 karakter). Gömme işlemi esnasında, seçilen metnin orijinalliği, gizli bilgi sadece kamufle edilmek suretiyle korunmuştur. Yani, anlatılan çalışmaların çoğunun aksine, burada gizli bilgiyi gömmek amacıyla seçilen metnin ne anlamı ne de formatı değiştirilmektedir. Bu, örten metni istatistiksel saldırılara karşı dirençli kılmaktadır. İletişim iki taraf arasında gerçekleştiğinden, iletişim kanalı olarak e-posta seçilmiştir. Bu nedenle stego ortam forward mail platformu şeklinde düzenlenmiştir. Böylece stego ortamın şüphe uyandırmaması ve algılanamaz olması amaçlanmaktadır. Kombinatorik tabanlı kodlama ile de, gözlemci için arzu edilen rastgelesellik sağlanarak güvenlik ve algılanamazlık desteklenmiştir. Latin karesi ve metin tabanından yararlanmak önerilen metodu herhangi bir dil için uyarlanabilir kılmaktadır. Çıkarım aşamasını karmaşıklaştırmak ve bu yolla güvenliği artırmak için yapılan işlemlerden bir diğeri ise iki çeşit stego anahtar kullanmaktır. Bu tez çalışmasında önerilen metin steganografi yaklaşımı, ilerleyen bölümlerde detaylı bir şekilde açıklanacaktır.

3. MATERYAL VE YÖNTEM

Son yıllarda internet çok hızlı bir büyüme göstermiştir. Birçok insanın dikkatini çeken alanlardan biri de internet üzerindeki bilgi güvenliğidir. Bu konuların arasında günümüzde, gizli bir iletişim kurma daha çok dikkat çeken sıcak bir konudur. (Shirali - Shahreza, 2008). Bu bölümde önerilen metoda ilişkin teorik bilgiler verilecektir. Bu kapsamda, steganografiye kısaca değinilerek metin steganografi teknikleri açıklanacak, kombinatorik tabanlı kodlama ve kayıpsız veri sıkıştırma tekniklerine de yer verilecektir. Ayrıca önerilen metin steganografi metodu da bu bölümde detaylı bir şekilde ele alınacaktır.

3.1. Steganografiye Bakış

Steganografi bilgi gizleme yöntemlerinin önemli bir alt dalıdır. Bu yaklaşım, bir nesnenin içerisine bir verinin gizlenmesi olarak tanımlanabilir. Bu yaklaşımla ses, sayısal resim, video görüntüleri içerisine veri saklanabilir. Görüntü dosyaları içerisine saklanacak veriler metin dosyası olabileceği gibi, herhangi bir görüntü içerisine gizlenmiş başka bir görüntü dosyası da olabilir. Bu yaklaşımda içine bilgi gizlenen ortama örtü verisi, oluşan ortama da stego-nesnesi denmektedir.

Steganografi kelimesi Yunanca steganos; gizli, saklı ve grafi; çizim ya da yazım kelimelerinden gelmektedir. Steganografi, Antik Yunan ve Heredot zamanına kadar uzanan oldukça eski bir veri gizleme yöntemidir. Heredot, İran Savaşları sırasında, kafasını kazıtıp kafa derisinin üzerine, gizli bir mesajın dövmesinin yapılmasına izin veren bir ulaktan bahsetmektedir. Mesaj yazıldıktan sonra ulak saçının uzamasını beklemekte, daha sonra ulak mesajı bekleyen kişiye ulaşmakta, kafasını tekrar tıraş etmekte, böylelikle mesaj ortaya çıkmaktadır. Bu yöntem bilinen ilk steganografi uygulamasıdır. Daha sonraki zamanlarda steganografi, harflere müzik notalarının atanması, I. ve II. Dünya Savaşlarında kullanılan mors kodları, II. Dünya savaşı esnasında başarıyla uygulanan görünmez mürekkeplerin kullanımı gibi uygulamalarla karşımıza çıkmaktadır.

Günümüzde ise sayısal nesneler üzerinde steganografi uygulamaları yapılmaktadır ve gelişen teknoloji nedeniyle, verilerimizi korumak amacıyla son yıllarda sıklıkla kullanılmaya başlanmıştır. Steganografi, Dilbilim Steganografi ve

Teknik Steganografi olmak üzere kendi içerisinde ikiye ayrılmaktadır. Dilbilim steganografi, taşıyıcı verinin metin olduğu steganografi koludur. Teknik Steganografi ise birçok konuyu içine almaktadır. Bunlar; görünmez mürekkep, gizli yerler, mikrodot, ve bilgisayar tabanlı yöntemler gibi başlıklar altında toplanabilmektedir. Bilgisayar tabanlı yöntemler ise metin, ses, görüntü, resim dosyalarını kullanarak veri gizleme yöntemleridir.

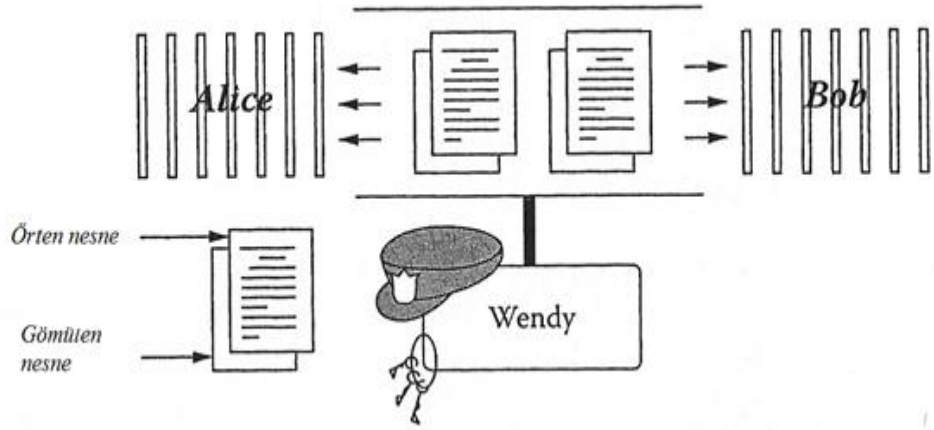
Steganografi, kullanım alanları açısından üçe ayrılmaktadır. Bunlar aşağıdaki gibidir:

- Metin steganografisi
- Görüntü steganografisi
- Ses steganografisi (Sahin ve ark., 2006)

3.1.1. Sayısal steganografi

Steganografi, veriyi başka bir veri içerisine saklama metodudur öyle ki, ilgili alıcıdan başka hiç kimse mesajın varlığını bilmez. Bu, steganografi ve diğer gizli bilgi değişim metotlarının temel farkıdır. Örneğin, kriptografide bilgi anlaşılmamasına rağmen, kodlanmış mesaj gözlenerek gizli bilginin varlığının farkına varılabilir (Shirali - Shahreza, 2008).

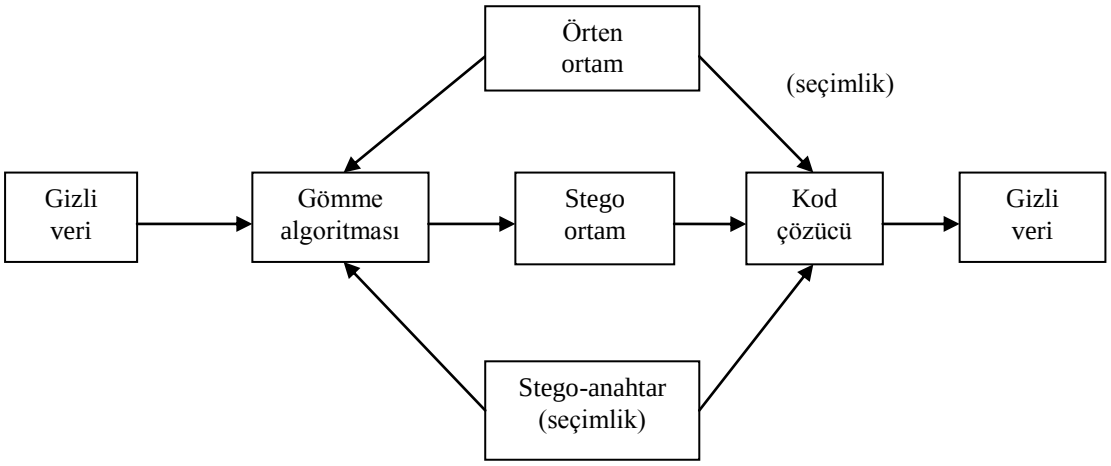
Sayısal steganografi, sayısal bilgiyi gizli kanallara saklamayı amaçlamaktadır öyle ki, bilgi saklanabilir ve gizli bilginin tespiti önlenebilir. Steganaliz ise; gizli bilginin varlığının keşfedilmesidir. Steganalitik sistemler; taşıyıcı ortamın gizlenmiş bir mesaj içerip içermediğinin tespiti için kullanılmaktadırlar.



Şekil 3.1. Klasik steganografik sistem

Şekil 3.1'de klasik bir steganografik model gösterilmektedir. Burada, Alice ve Bob hapishaneden kaçmayı planlamaktadırlar. Alice ve Bob arasındaki tüm iletişim bekçi olan Wendy tarafından gözlemlenmektedir. Bu nedenle her biri, birbirinin stego nesnesini elde etmek için mesajlarını zararsız görünen medya ortamlarına (örten nesnelere) saklamalıdırlar. Daha sonra stego nesneler herkese açık kanallar vasıtasıyla gönderilebilirler. Wendy, Alice ve Bob arasındaki mesajları aktif veya pasif olarak yalnızca bir şekilde denetleyebilmektedir. Pasif yaklaşım, mesajın gizli bir bilgi içerip içermediğinin tespiti amacıyla kontrol edilmesi ve bu doğrultuda ilgili işlemin yürütülmesinden ibarettir. Aktif yaklaşım ise; Wendy'nin, herhangi bir gizli örüntü izi bulamasa bile Alice ve Bob'un mesajlarını her durumda değiştirmesinden ibarettir (Shih, 2005).

Şekil 3.2'de tipik bir steganografik sistemin ana adımları gösterilmektedir. Buna göre, kodlama algoritması üç giriş almaktadır; gömülecek gizli veri, örten veri ve seçimlik olarak kullanılan bir stego anahtar. Sonrasında algoritma, saklanabilen yada iletilebilen bir stego ortam üretmektedir. Kod çözme algoritması ise, stego ortamı ve kullanılan stego anahtarı giriş olarak almakta ve gizli veriyi çıkarmaktadır. Bazı algoritmalarda kod çözücü, veriyi tam olarak çıkaramamakta ve sadece "İncelenen dosyada gerçekten gizli bir veri var mı?" sorusuna cevap verebilmektedir. Bu, gizli bilgi örten ortama saklanan bir filigran olduğunda fark yaratmaktadır. Ayrıca, bazı kod çözücülerin stego ortama gömülen veriyi çıkarmak için orijinal örten nesneye ihtiyaç duyduğu unutulmamalıdır (Salomon, 2005).



Şekil 3.2. Veri gizleme ve çıkarım işlemlerinin ana adımları (Salomon, 2005)

3.2. Metin Steganografisi

Faklı dillerin farklı karakteristik özellikleri vardır. Metin steganografisinde bilgi metnin içine saklandığından, örten ortam olarak kullanılan dile oldukça bağımlı olunmaktadır (Al-Nazer ve Gutub, 2009). Normal olarak, tüm diller için tek bir yöntem kullanmak mümkün değildir (Alla ve Prasad, 2009). Diğer medya türlerindeki yapı görünen kısımdan oldukça farklıdır ancak metinsel dokümanların yapısı görünen kısım ile oldukça benzerdir. Bu nedenle metinsel dokümanlarda, fark edilebilir bir değişlik yapmadan bilginin saklanması gerekmektedir (Shirali-Shahreza ve Shirali-Shahreza, 2006). Metin steganografisi en zor steganografi türüdür. Bunun nedenlerinden biri, diğer medya türleri ile karşılaştırıldığında metinsel verinin daha az fazlalık içermesi, bir diğeri ise insanların anormal görünümdeki bir metne karşı oldukça duyarlı olmalarıdır (Samphaiboon, 2009). Ancak metin steganografisini tercih etmenin avantajları da mevcuttur. Bunlar, daha az hafıza işgali ve basit iletişimdir (Alla ve Prasad, 2009).

3.2.1. Metin steganografi teknikleri

A. Kelimelerdeki belirli karakterler

Belirli kelimelerdeki karakterleri seçerek bilgi gizleme gerçekleştirilebilmektedir. Bu metot şartlara göre basitten oldukça karmaşığa kadar değişebilir. Bu metodun en basit halinde, her kelimenin ilk harfi alınır ve bu kelimelerdeki ilk harfler yan yana yerleştirilerek gizlenmiş bilgi çıkarılabilmektedir. Daha gelişmiş bir örnek ise ilk kelimenin ilk harfi, ikinci kelimenin ikinci harfi vb. şeklinde gizleyerek bilgiyi saklamaktır (Gutub ve Fattani, 2007). Örnek olarak:

PRESIDENT'S EMBARGO RULING SHOULD HAVE IMMEDIATE NOTICE. GRAVE SITUATION AFFECTING INTERNATIONAL LAW. STATEMENT FORESHADOWS RUIN OF MANY NEUTRALS. YELLOW JOURNALS UNIFYING NATIONAL EXCITEMENT IMMENSELY.

Verilen metinde her bir kelimenin ilk harfi alındığında çıkarılan gizli mesaj şu şekildedir:

PERSHING SAILS FROM NY JUNE I (Rafat ve Sher, 2010).

B. HTML dokümanları

Gizli bilgi, anlaşılmaz olduklarından HTML dokümanları içine saklanabilmektedir. HTML dokümanlarında steganografi, doküman etiketlerindeki büyük harfler küçük, küçük harfler ise büyük yaparak gerçekleştirilebilmektedir (Gutub ve Fattani, 2007). Bunun yanında HTML dokümanlarındaki etiketlerin değişik kombinasyonları yada yatay boşluk veya hizalamalardan faydalanılabilir. Bir alternatif olarak ise aşağıdaki örnek gösterilebilir:

Gizleme işlemine dair izlenecek yol şu şekilde olsun:
<img> </img> → 0
<img/> → 1

Gönderilen stego metin:

<img src=g1.jpg></img>

<img src=g5.jpg></img>

Buna göre stego metin içerisinden çıkarılan gizli bitler şu şekildedir: 01110 (Rafat, 2009).

Bilgi çıkarımı ise bu etiket kelimelerini normal halleriyle karşılaştırarak gerçekleştirilebilmektedir. HTML steganografisinin güvenliğini, harfleri belli bir şekilde sıralayan bir fonksiyon oluşturularak artırmak mümkündür. Bu şekilde mesajı gözlemlemekte olan gözlemci şaşırtılabilir (Gutub ve Fattani, 2007).

C. Satır ve kelime kaydırma

Metin satırlarını dikey olarak ve kelimeleri ise yatay olarak kaydırmak bilgi saklamaya yardımcı olabilir. Bu metodun güvenilirliği, kelimeler ve satırlar arasındaki uzaklığın değişebilirlik durumuna bağlı olmaktadır. Bu steganografi metodunda, satırlar sabit bir mesafede (0.003 inch gibi) biraz aşağı ya da yukarı kaydırılmakta ve saklanması düşünülen bilgiye göre bu mesafeler yeniden modifiye edilmektedir. Bu metin kaydırma steganografisi, boşluklara gömülecek bilgi için görsel şekiller oluşturmaya dayanmaktadır. Bu teknik, sağlamlık probleminden ötürü basılı metinler için uygundur. Metin elektronik olarak yeniden yazıldığında ya da düzenlendiğinde gizli bilginin yok edilme olasılığı artmaktadır. Ayrıca, karakter tanıma programları kullanıldığında bu görsel şekillere saklanmış bilgi kaybedilir veya doğru olarak izlenemez (Gutub ve Fattani, 2007).

D. Kısaltmalar ve boşluklar

Kısaltma ve boşluk steganografisinde metne çok az bilgi saklanabilir. Boşluk steganografisinde özellikle kelimeler arasına ekstra boşluk eklenerek ya da metindeki

satır sonlarına veya paragraflara ekstra boşluk eklenerek bilgi saklanmaktadır. Bu teknik herhangi bir metne uygulanabilmekte ve okuyucuda herhangi bir şüphe uyandırmamaktadır. Ancak burada kapasite ve sağlamlık düşüktür. Ayrıca bazı elektronik metin editörleri otomatik olarak boşlukları kaldırmaktadırlar (Gutub ve Fattani, 2007). Çizelge 3.1'de kısaltma metoduna ilişkin kullanılabilecek bir kısaltma - kelime tablosu verilmektedir. Şekil 3.3 ve 3.4'te ise anlatılan boşluk metoduna uygun örnekler gizleme işlemi öncesi ve sonrasında gösterilmektedir.

Çizelge 3.1. Kısaltma-kelime tablosu (Rafat, 2009)

Kısaltma	**Kelime**
218	Too late
ASAP	As Soon As Possible
C	See
CM	Call Me
F2F	Face to Face

T	h	e		q	u	i	c	k		b	r	o	w	n		f	o	x
j	u	m	p	s		o	v	e	r		t	h	e		l	a	z	y
d	o	g	.															

Şekil 3.3. Gizleme öncesi orjinal metin (Rafat, 2009)

T	h	e		q	u	i	c	k		b	r	o	w	n		f	o	x	
j	u	m	p	s		o	v	e	r		t	h	e		l	a	z	y	
d	o	g	.																

Şekil 3.4. Gizleme sonrası oluşan stego metin (Rafat, 2009)

E. Anlamsal ve karakter öznitelik metotları

Gizli bilgiyi, elektronik yazı veya bir önceki kaydırma yaklaşımındaki optik karakter tanıma sistemlerinin kullanımından korumak amacıyla, anlamsal ve karakter

öznitelik metotları önerilmiştir. Anlamsal metotlar, metin içine veri gizlemek için belli kelimelerin yerine eş anlamlıların kullanımını önermektedir. Ancak bu metot, bilgiyi saklama amacıyla kullanılacak olan metnin anlamını değiştirebilmektedir.

Karakter öznitelik steganografisi, metin karakterlerinin bazı özelliklerini değiştirmektedir. Örneğin bazı karakterlerin en anlamlı bitleri, gizli bilginin bitlerini tutmak üzere genişletilmişlerdir. Bu metot, okuyucuda gizli bir bilginin varlığı şüphesini uyandırmadan yüksek miktarda veri tutabilmektedir (Gutub ve Fattani, 2007)

Başka bir çalışmada ise geliştirilen metin steganografi sistemleri, üç kategoriye ayrılmıştır:

Metin tabanlı görüntü steganografisinde örten ortam metin tabanlı görüntülerdir. Bu kategoride, metinsel elemanlar bağlı ikili, gri ölçekli ya da renk pikselleri olarak görülmektedirler. Diğer yandan, metinsel olmayan elemanlar art alan renk pikselleri olarak görülmektedir. Görüntüdeki bazı piksellerin yoğunluklarını ayarlayarak gizli bilgi, metin tabanlı örten görüntü içerisine gömülebilir. Burada ayrıca gizli bilgiyi gömmek için kelime kaydırma, satır kaydırma ve kelimeler arasındaki boşlukları ayarlamak da mümkündür.

Zengin metin doküman steganografisinde örten ortam sayısal zengin metinlerdir. Bu kategoride, her bir karakter rastgele olarak farklı bir formatta (farklı boyut ve yazı tipinde) yazılabilmektedir. Dokümanda görülen tüm metinsel ve metinsel olmayan elemanlar, dokümanda kullanılan karakter kümesine göre kodlanmış bağımsız karakterlerdir. Metin tabanlı görüntülerin aksine, piksel değerlerini ayarlayarak gizli bilgiyi sayısal zengin metinsel görüntü içerisine gömmek için fazlalık yoktur. Ancak bağımsız karakterlerin boyut ya da fontunu biraz ayarlayarak gizli bilgi gömülebilmektedir.

Düz metin doküman steganografisinde, örten ortam sayısal düz metin dosyalarıdır. Bu kategoride karakterlerin boyut ya da font bilgisi bulunmamaktadır. Zengin metinsel dokümanlara benzer olarak bu kategoride, bir dokümanda metinsel ve metinsel olmayan elemanlar bağımsız karakterler olarak görülmektedirler. Ancak aksine, burada piksel değerleri veya karakter formatlarını ayarlayarak (font, boyut vb.) gizli bilgiyi gömme amacıyla fazlalık bulunmamaktadır. Bu nedenle bu kategoride örten metinler diğer kategoridekilere göre daha az fazlalık içermektedirler. Burada yalnızca eş anlamlı yer değiştirme, söz dizimsel dönüşüm, anlamsal dönüşüm, metin kısaltması veya karakter/sembol düzenlemesi metotları uygulanarak gizli bilgi gömülebilir. Metin steganografi düzenlemelerinin arasında düz metin dokümanlarında steganografi, metin

tabanlı görüntü steganografisi gibi diğer metin steganografisi türlerinden daha az fazlalık içerdiğinden en çok ilgi çekici tür olarak bilinmektedir (Samphaiboon, 2009).

3.3. Kombinatorik Tabanlı Kodlama

Kombinatorik tabanlı kodlama, bilgi tasarrufu, veri depolama ve iletiminde büyük öneme sahip olan bir veri kodlama türüdür. Kombinatorik kodlama, karakter serisinin yeri ve buna karşılık gelen serinin yeri arasındaki ilişkiyi kullanarak kodlama işlemini gerçekleştirir. Ancak kombinatorikte cevaplandırılması gereken bir soru vardır:

n eleman içeren ve elemanları $a_1, a_2, \ldots, a_n$ şeklinde birbirinden farklı olan bir seri olduğunu farz edelim. Bu seriye tam permütasyon uygulandığında farklı permütasyonların sayısı *n*! olmaktadır. w_1 defa tekrar eden bir a_1 elemanı mevcutsa, permütasyonların sonucu aynı olacaktır. Aslında, farklı permütasyon sayısı $n!/w_1!$olmalıdır. Yani, tekrar etme derecesi $w_1!$ olmaktadır. Eğer a_1, w_1 defa, a_2, w_2 defa ve a_m, w_m defa tekrar ediyorsa, $w_1 + w_2 + \ldots + w_m = n$ olmalıdır. Yani, tekrar eden elemanların toplam sayısı *n* ve farklı eleman sayısı *m* olmaktadır. Bu seriye tam permütasyon uygulandığında, farklı permütasyonların sayısı $n!/(w_1! \times w_2! \times \ldots \times w_m!)$ olmaktadır ve bu sayı *n!*'den daha küçüktür. Kombinatorik tabanlı kodlamaya bu tür bir soru ile açıklık getirilmiştir. Böylece, bir seri için gerçek farklı permütasyon sayısı aşağıdaki formül ile ifade edilebilmektedir (Jun ve ark., 2011):

$$P = \frac{n!}{\prod_{i=1}^{m}(w_i!)} \tag{3.1}$$

3.3.1. Latin karesi

Latin kareleri, çeşitli kombinatorik tasarım düzenlerinde ortaya çıktığından, yaygın olarak çalışılmıştır. *n.* derecen bir Latin karesi, $n \times n$ boyutunda bir dizidir. Her bir hücre ise $\psi = \{1, 2, \ldots, n\}$ kümesinden bir eleman içermektedir. Dizinin her bir satır ve sütunu her bir elemanı yalnızca bir defa içermektedir. Şekil 3.5'te tamamlanmış bir Latin karesi gösterilmektedir.

1	2	3	4	5
5	1	2	3	4
4	5	1	2	3
3	4	5	1	2
2	3	4	5	1

Şekil 3.5. 5×5 bir Latin karesine uygun bir tamamlama

Şekil 3.5'te gösterildiği gibi, Latin karesi oluşturmanın kolay bir yolu, ilk sıraya *1, 2,…, n* tamsayılarının herhangi bir permütasyonunu yerleştirmektir ve sonrasında *n* tamsayılarının kalan *n-1* döngüsel permütasyonunu *2*'den *n*'ye kadarki ardışık sıralara yerleştirmektir. Bu her zaman, çok büyük olasılıkları çıkan tek bir geçerli Latin karesine yer vermektedir. Örneğin her bir *n* için, en az $n!(n-1)!\ldots(2!)1$ uygun konfigürasyon vardır (Easton ve Gary Parker, 2001).

Önerilen çalışmada Latin karesi, her bir sembolü (yani her bir harfi), yer bilgisini içeren sayısal dizinin her bir değerine eşleme yoluyla kullanılmıştır. Bu eşlemenin sonucu, yer bilgisinin her biri için değişmektedir. Böylece, istenilen rastgelesellik ve karmaşıklık artırılmaktadır.

3.4. Veri Sıkıştırma

Sıkıştırma, bilginin fiziksel boyutunu azaltmak için kullanılan bir işlemdir. Aynı ortam üzerinde daha fazla bilgi depolayabilmek, bir ağ üzerindeki disk boyutunu veya gönderim zamanını ve bant genişliğini azaltabilmek ve veriyi daha sonra tekrar kullanabilmek sıkıştırma işleminin ana amaçlarıdır (Liang ve ark., 2008).

Veri sıkıştırma işleminde amaç, verilen türdeki verinin fazlalıklarını azaltmaktır (Galambos ve Bekesi, 2002). Veri sıkıştırma algoritmaları genellikle kayıplı ya da kayıpsız olarak sınıflandırılırlar. Kayıpsız veri sıkıştırma işlemi, orijinal veri setinin dönüşümü sonrasında gerçekleştirilen çözme işlemi sonucu aynı verinin birebir üretiminin mümkün olmasını gerektirmektedir. Kayıpsız sıkıştırma, orijinal verinin önemli olduğu ve orijinal veri ve çözülen veri dosyalarının özdeş (metin dosyaları, çalıştırılabilen kod dosyaları, kelime işlemci dosyalarının sıkıştırılması) olması gerektiği

durumlarda kullanılmaktadır. Kayıplı veri sıkıştırma, orijinal veri setinden verinin tekrar üretiminin mümkün olmadığı bir dönüşümdür ve gerçekleştirilen bir çözme işlemi ile ancak orijinal veriye yakın bir temsil oluşturulabilir. Bu türdeki sıkıştırma internet ve özellikle duraklamasız medya ortamları ve telefonculuk uygulamalarında kullanılmaktadır (Al-Bahadili, 2008). Bu noktada, şu iki tanımlama gerçekleştirilebilir:

TANIM 1.Sıkıştırma, verilen bir *D* kaynak bilgisinden, daha kısa bir $\Delta(D)$ bilgisini üreten bir süreçtir.

TANIM 2. Kayıpsız sıkıştırma, $\Delta(D)$ bilgisinden *D* kaynak bilgisinin birebir çıkarılabildiği bir işlemdir. Kayıplı sıkıştırma ise, orijinal verinin yaklaşık olarak kodlanabildiği bir metottur (Galambos ve Bekesi, 2002).

Kayıplı ve kayıpsız sıkıştırma teknikleri arasındaki ayırım önemlidir çünkü kayıplı sıkıştırma metotlarında sıkıştırma oranı kayıpsız sıkıştırma metotlarına göre daha fazladır. Kayıpsız sıkıştırma metotları genellikle 2:1 oranından 8:1 oranına kadar sıkıştırma oranı elde edebilirler. Kayıplı sıkıştırma metotları ise, 100:1 oranından 200:1 oranına kadar sıkıştırma sağlayabilirler. Buna ilaveten, orijinal veride hatalara ne kadar fazla tolerans gösterilirse, o kadar fazla sıkıştırma oranı elde edilir. Ayrıca özellikle kayıplı sıkıştırma metotlarında, sıkıştırma verimi, kaynak bilginin karakteristiğinden önemli derecede etkilenmektedir (Al-Bahadili, 2008).

Veri sıkıştırma tekniklerinin üç ana modeli mevcuttur: Yer değiştirme, İstatistiksel ve Sözlük Tabanlı sıkıştırma. Yer değiştirme sıkıştırması, tekrar eden karakterlerin tümü için daha kısa bir ifade kullanılmasını içermektedir (Null Suppression, Run Length Encoding, Bit Mapping, Half - Byte Packing). İstatistiksel sıkıştırma teknikleri, karakterlerin hesaplanan olasılıklarına göre en kısa ortalama kod uzunluğunun üretimini içermektedir (Shannon - Fano Coding, Static - Dynamic Huffman Coding, Arithmetic Coding). Bu tür sıkıştırma tekniklerinde, kaynak dosyadaki karakterler ikili koda dönüştürülür. Dosyadaki en genel karakterler en kısa ikili kodu alırken, en az genel olan karakterler en uzun ikili kodu almaktadır. Son olarak sözlük tabanlı sıkıştırma metotları mevcuttur. Bu tür sıkıştırma tekniklerinde, metindeki karakterler, oluşturulan bir sözlüğe göre indis veya işaretçi kodu ile gösterilirler (Lempel Ziv Welch - LZW). Çoğu sıkıştırma algoritması, sıkıştırma oranını artırmak için farklı veri sıkıştırma tekniklerinin kombinasyonlarını kullanmaktadır. Sıkıştırma

miktarı, sıkıştırma oranı olarak bilinen bir *C* faktörü ile ölçülmektedir (Al-Bahadili, 2008):

$$C = \frac{So}{Sc} \tag{3.2}$$

Burada *So*; orijinal dosya boyutu, *Sc* ise sıkıştırılan dosya boyutudur. Sıkıştırma oranı *C:1* şeklinde ifade edilmektedir. Sıkıştırma miktarı aynı zamanda orijinal veri miktarındaki azalma ile de ölçülebilmektedir:

$$R = \frac{So - Sc}{So} \tag{3.3}$$

R genellikle yüzde ile ifade edilmektedir (Al-Bahadili, 2008).

3.4.1. LZW algoritması

LZW algoritmasını açıklamadan önce dizi eşleşme problemine (string matching problem) bakalım:

TANIM 3. Verilen bir $P = p_1,\ldots, p_m$ örüntüsü ve $T = t_1,\ldots, t_u$ metni için *T* içinde mevcut olan tüm *P* örüntülerini bul ve $\{|x|, T = xPy\}$ kümesini döndür.

Dizi eşleşme probleminin ilgi çekici bir özelliği de metin sıkıştırması ile ilgili olmasıdır (Kärkkäinen ve ark., 2003). Doğal olarak bilgi metin, görüntü, ses gibi değişik formatlarda olabilir. Veri sıkıştırma açısından, bunların arasında önemli bir fark vardır. Metinsel bilgide sıkıştırma/çözme işlemi gerçekleştirme durumunda orijinal verinin birebir olarak geri getirilmesi gerekmektedir. Ancak görüntü ve ya ses dosyalarında orijinal bilginin yakın bir tahminini yeterli olmaktadır (Galambos ve Bekesi, 2002). Metin sıkıştırmasında daha az yer kullanılarak verinin temsili amacıyla fazlalıklardan yararlanılmaktadır. Ziv–Lempel ailesi başarılı sıkıştırma oranı ile verimli sıkıştırma ve çözme zamanı sebebiyle uygulamada en yaygın olan sıkıştırma tekniklerindendir (Kärkkäinen ve ark., 2003).

Sıkıştırılmış eşleşme problemi (compressed matching problem), sıkıştırılmış bir metinde, çözme işlemi uygulamadan dizi eşleşme probleminin gerçekleştirilmesidir (Kärkkäinen ve ark., 2003):

TANIM 4. Verilen bir $T = t_1,\ldots,t_u$ metni, buna tekabül eden sıkıştırılmış $Z = z_1,\ldots, z_n$ dizisi ve $P = p_1,\ldots, p_m$ örüntüsü için sıkıştırılmış eşleşme problemi, sadece P ve Z'yi kullanarak T içindeki mevcut tüm P'lerin bulunmasından ibarettir.

Uygulamada sıkıştırılmış dizi eşleşme problemi önemlidir. Günümüz metin tabanları, şu her iki problemin de kritik derecede önemli olduğu iyi bir örnektir; (1) metinler, yer kazancı, giriş – çıkış ve ağ zamanı için sıkıştırılmalı ve (2) metinler etkin olarak aranabilmelidir. 90'lı yıllardan önceki tek çözüm, metni sıkıştırmadan sorguları gerçekleştirmek ve metinlerin içerisinde arama yapmak olduğundan, bu iki ortak gereksinimin de birlikte karşılanması kolay değildir. LZ (Lempel Ziv) ailesi, uygulanabilirliği, iyi sıkıştırma oranları ve hızlı sıkıştırma/çözme zamanı sebebiyle oldukça popülerdir (Kärkkäinen ve ark., 2003).

LZW kodlama tekniğinde hali hazırda bir sözlük yoktur. LZW algoritması öncelikle veriyi okur ve sözlükte kodlanan bir diziden yararlanarak mümkün olduğunca büyük veri biti serisi ile eşleşme yapmaya çalışır. Eşleşen veri sırası ve bunu izleyen karakter sonraki veri serilerinin kodlanması amacıyla birlikte gruplandırılarak sözlüğe eklenir. Daha küçük, sıkıştırılmış bir kod daha yüksek sıkıştırma oranıyla sonuçlanırken, sözlük boyutunu da sınırlandırmaktadır. Aşağıda, algoritmanın işleme şekli verilmiştir:

1. C veri dizisindeki bir sonraki karakter olsun.
2. $P + C$ dizisi sözlükte var mı?
    2.1. Eğer var ise, $P \leftarrow P + C$ (P'yi C ile genişlet)
    2.2. Yok ise
        2.2.1. P kod kelimesini, kod dizisine çıkış olarak ver
        2.2.2. $P + C$ dizisini sözlüğe ekle
        2.2.3. $P \leftarrow C$ (P bu durumda sadece C karakterini içermektedir.)
3. Veri dizisinin sonuna gelindi mi?
    3.1. Hayır ise 2. Adıma git
    3.2. Evet ise P kod kelimesini, kod dizisine çıkış olarak ver (Liang ve ark., 2008).

3.4.2. Huffman algoritması

Huffman kodu, olasılıkların kümesinden üretilen bir ek koddur. Öncelikle veri kümesi, olasılık modelini çıkarmak için taranır ve elde edilen bu olasılıklarla bir kod ağacı oluşturulur (Liang ve ark., 2008). Semboller olasılıklarına göre artan sırada sıralanır. Sonrasında sembol olasılıklarının toplamı her bir kümeye atanarak öz yineli olarak bazı kümelerde toplanır. Her bir adımda, kümedeki sembollerin koduma bir bit eklenir. Algoritma aşağıdaki gibi işlemektedir:

1. Sembollerin olasılıklarını çıkar
2. Sembolleri olasılıklarına göre artan sırada sırala
3. En düşük olasılıklı iki sembolü al
4. Bu sembolleri, olasılıkları bu iki olasılığın toplamı olan bir küme yer değiştir. İlk sembolün koduna sıfır değerli bir bit, diğerine ise 1 değerinde bir bit ata.
5. Bu sınırlandırılan yeni kümeler için yalnızca bir eleman içeren bir liste elde edilinceye kadar önceki 3 adımı (2, 3, 4) tekrarla.

TANIM 5. Bir Huffman ağacının harici yol uzunluğu, kökten yapraklara tüm yol uzunluklarının toplamıdır (Galambos ve Bekesi, 2002).

Ön-kod ağacı oluştururken en uygun değişken uzunluktaki ikili bir kod için bazı koşullara dikkat edilmesi gerekmektedir:

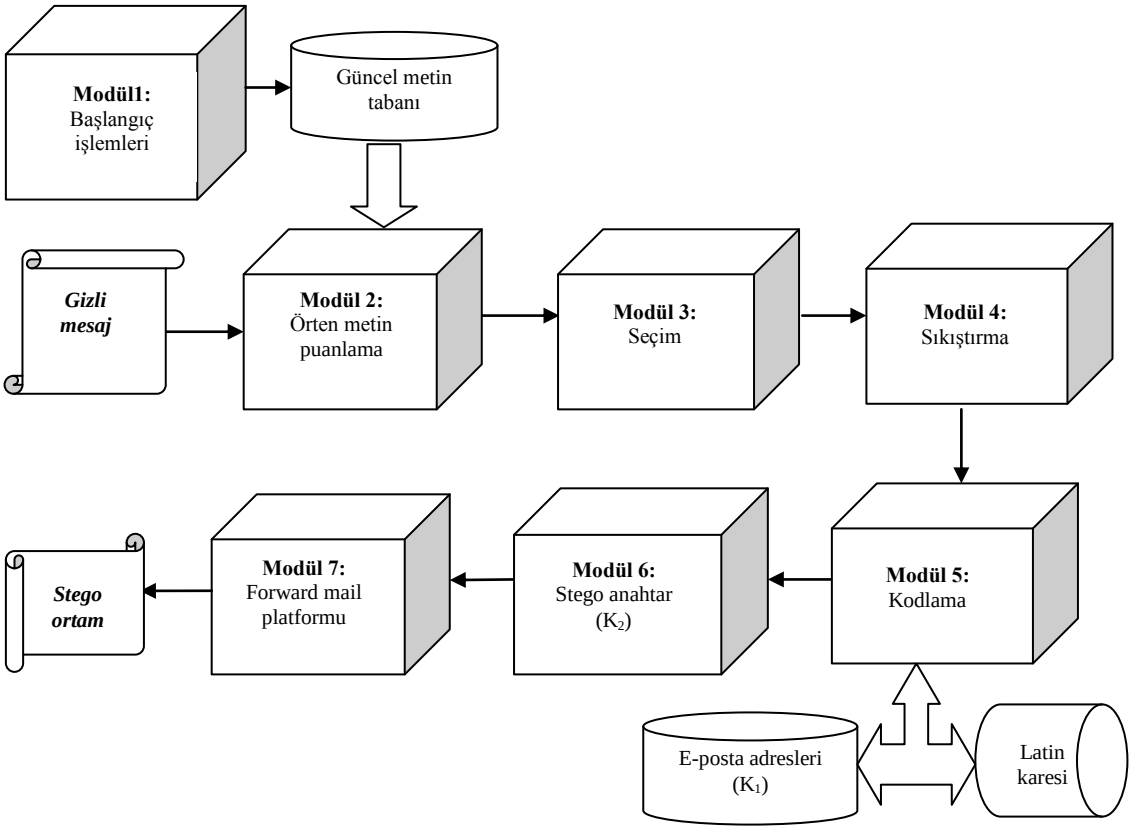
1. a_i ve a_j şeklinde verilen iki harf için $P[a_i] \geq P[a_j]$ ise, $l_i \leq l_j$ olmaktadır (l_i; a_i için üretilen kod kelimesi içindeki bit sayısıdır).
2. Olası en az iki harf, aynı maksimum l_m uzunluğundaki kod kelimelerine sahiptir.
3. Ağaçta, optimum koda karşılık gelen, her bir orta seviye düğümden çıkan iki dal olmalıdır.
4. Bir ara düğüm, kendisinden çıkan tüm yaprakları, indirgenmiş bir alfabenin bileşik bir kelimesi içinde, bir araya getirerek bir yaprak düğüme dönüştürülebilir. Orijinal ağaç, orijinal alfabe için en uygun seçenekse, indirgenmiş ağaçta indirgenmiş alfabe için en uygun seçenektir (Liang ve ark., 2008).

3.5. Önerilen Yöntem

Bu bölümde önerilen metot LZW ve Huffman kodlaması için hem gönderici tarafı hem de alıcı tarafı ele alınarak açıklanmıştır. Metodun gömme ve çıkarım aşaması detaylandırılarak anlatılmış ve ayrıca kullanılan stego anahtarlar da bu bölümde açıklanmıştır.

3.5.1. Gönderici tarafı: Gömme aşaması

Şekil 3.6'da önerilen metodun gömme aşamasının mimarisi gösterilmektedir:



Şekil 3.6. Gömme aşamasının mimarisi

Gömme aşamasını anlatmadan önce aşağıdaki değişkenleri açıklayalım:

S: Gizli mesaj (Küme)

T: Metin Tabanı

Text: Metin Tabanındaki bir metin

$\overrightarrow{\Delta D}$: Göreli uzaklık vektörü

A: E - posta adres uzantı kümesi

$\vec{F}$:Huffman kodlama frekansları

NT: Metin tabanındaki metin sayısı

D: Göreli uzaklık matrisi

E: Taşma matrisi

R: Yeniden yapılandırılan göreli uzaklık matrisi

K_1: Global Stego Anahtar (Küme)

K_2:Seçilen ve stego anahtar olarak düzenlenen e - posta adres kümesi

MaxC: Maksimum karakter sayısı

$$S = \{a_1, a_2, \dots, a_m\}$$

$$D = D_{30,m} = \begin{bmatrix} d_{1,1} & \dots & d_{1,m} \\ \vdots & \dots & \vdots \\ d_{30,1} & \dots & d_{30,m} \end{bmatrix}$$

$$T = T_{30,300} = \begin{bmatrix} t_{1,1} & \dots & t_{1,300} \\ \vdots & \dots & \vdots \\ t_{30,1} & \dots & t_{30,300} \end{bmatrix}$$

$$E = E_{30,m} = \begin{bmatrix} e_{1,1} & \dots & e_{1,m} \\ \vdots & \dots & \vdots \\ e_{30,1} & \dots & e_{30,m} \end{bmatrix}$$

$$Text = \{b_1, b_2, \dots, b_n\}$$

$$R = R_{30,m} = \begin{bmatrix} r_{1,1} & \dots & r_{1,m} \\ \vdots & \dots & \vdots \\ r_{30,1} & \dots & r_{30,m} \end{bmatrix}$$

$$\overrightarrow{\Delta D} = (c_1, c_2, \dots, c_m)$$

$$K_1 = \{j_1, j_2, \dots, j_{676}\}$$

K_1 temsili olarak şu şekilde gösterilebilir:

K₁={ aa...@hotmail.com,
ab...@hotmail.com,
ac...@hotmail.com,
ad...@hotmail.com,
...,
zv...@hotmail.com,
zy...@hotmail.com,
zz...@hotmail.com}

A = {hotmail.com, gmail.com, yahoo.com, msn.com, windowslive.com, mail.com, myspace.com, mynet.com}
(İkili İndeks = 000, 001, 010, 011, 100,101, 110,111)

$\vec{F} = (f_1, f_2, \dots, f_{26})$

S, gizli mesajın karakterlerini (*a*) içeren bir kümedir. *Text* ise metin tabanındaki (*T*) herhangi metni ifade etmektedir ve her bir karakteri *b* olarak gösterilmektedir. Burada metin tabanı 30 adet örnek metin (*Text*) içermektedir. 300 ise metin tabanındaki maksimum karakter sayısıdır. 300 karakterden az olan *Text* kümesinin ilgili elemanları 0 olarak atanmaktadır.

Modül 1- Başlangıç işlemleri:

Metin tabanı, steganografik örten ortamı daha şüphesiz ve uygun göstermek amacıyla güncellenmektedir. Bu yolla benzer örüntülerin saklanması durumumda aynı metnin tekrar tekrar kullanımı önlenmektedir. Güncelleme işlemi; metin tabanı toplu hitapta kullanılacak hatırlatma, bildirim, zorunluluk, haber vb. metinleri içerecek şekilde gerçekleştirilmektedir. Metinler, alıcı ve göndericinin hobi ve meslek vb. durumlarını dikkate alarak da seçilebilmektedir. Son olarak, hatalı bir işlemi önlemek amacıyla her bir metnin karakter sayısı ve metin tabanındaki metin sayısı hesaplanır (Metin tabanındaki maksimum karakter sayısı ve metin tabanındaki metin sayısı, metodun işletilebilmesi için ilgili değişkenlere atanmaktadır).

$$D = D_{NT,m} = \begin{bmatrix} d_{1,1} & \dots & d_{1,m} \\ \vdots & \dots & \vdots \\ d_{NT,1} & \dots & d_{NT,m} \end{bmatrix} \quad T = T_{NT,MaxC} = \begin{bmatrix} t_{1,1} & \dots & t_{1,MaxC} \\ \vdots & \dots & \vdots \\ t_{NT,1} & \dots & t_{NT,MaxC} \end{bmatrix}$$

$$E = E_{NT,m} = \begin{bmatrix} e_{1,1} & \dots & e_{1,m} \\ \vdots & \dots & \vdots \\ e_{NT,1} & \dots & e_{NT,m} \end{bmatrix} \quad R = R_{NT,m} = \begin{bmatrix} r_{1,1} & \dots & r_{1,m} \\ \vdots & \dots & \vdots \\ r_{NT,1} & \dots & r_{NT,m} \end{bmatrix}$$

Burada *NT = 30* ve *MaxC = 300* olmaktadır. Eğer herhangi bir metnin karakter sayısı *MaxC*'den küçük ise ilgili elemanları *0* olarak atanmaktadır. Metin tabanında çok fazla metin olması yerine, metin tabanının kullanıcıya bağlı olarak güncellenmesi hesaplama zamanı açısından daha avantajlı olmaktadır.

Modül 2- Örten metin puanlama:

a) *S* ve *Text* ele alınarak, $a = b$ durumu aranır. Dolayısıyla, $\overrightarrow{\Delta D}$; elemanları (*c*), bu karakter eşlemesinin gerçekleştiği yerdeki *b* elemanlarının indekslerinin farkı olan bir vektördür. Bu durum şu şekilde açıklanabilir:

Karakterler ASCII kodları olduğundan;

$$\begin{aligned} a_1 = b_1 &\rightarrow a_1 - b_1 = 0 \\ a_1 = b_2 &\rightarrow a_1 - b_2 = 0 \\ &\vdots \\ a_1 = b_n &\rightarrow a_1 - b_n = 0 \end{aligned}$$

$a_1 = b_2$ olduğunu farz edelim. Bu durumda $\overrightarrow{\Delta D}$ elemanı olan *c, b'* nin indeksi olan 2 değeridir. Daha sonra kalınan *S* ve *Text* elemanlardan devam edilerek:

$$\begin{aligned} a_2 = b_3 &\rightarrow a_2 - b_3 = 0 \\ a_2 = b_4 &\rightarrow a_2 - b_4 = 0 \\ &\vdots \\ a_2 = b_n &\rightarrow a_2 - b_n = 0 \end{aligned}$$

$a_2 = b_4$ olduğunu farz edelim Bu durumda $\overrightarrow{\Delta D}$ elemanı olan *c*; *b*'nin şimdiki indeksi ve bir önceki indeksi arasındaki fark olacaktır: *4 - 2 = 2*. Bu işlem *S* küme elemanları adedince her bir *Text* için iteratif olarak devam eder ve böylece $\overrightarrow{\Delta D}$ oluşturulur.

b) *T* içindeki her bir *Text* için $\overrightarrow{\Delta D}$ oluşturulur. Böylece her bir $\overrightarrow{\Delta D}$ vektörü tutularak *D* matrisi oluşturulur:

$$D = D_{30,m} = \begin{bmatrix} d_{1,1} & \dots & d_{1,m} \\ \vdots & \dots & \vdots \\ d_{30,1} & \dots & d_{30,m} \end{bmatrix}$$

D matrisi, *T* içindeki en uygun *Text* seçimi için kullanılmaktadır. Bu işlemin amacı LZW kodlaması için en uygun metni bulmaktır. *D* elemanlarının 26 değerini aşıp aşmadığına bakılarak *E* ve *R* oluşturulur:

$$E = E_{30,m} = D \backslash 26 \tag{3.4}$$

$$R = R_{30,m} = D mod\ 26 \tag{3.5}$$

Burada amaç Latin karesinin sınırları dışına taşmamaktır. Eğer taşma yoksa ilgili *e*; 0 ilgili *r* ise *d* değerine eşit olacaktır.

c) *R* içindeki her bir satır için ikili örüntü tekrarı sayılır ve böylece *P* vektörü elde edilir:

$$P = \begin{bmatrix} p_1 \\ \vdots \\ p_{30} \end{bmatrix}$$

Modül 3 - Seçim:

P vektörü içindeki en fazla *p* değeri seçilir ve bu değerin indeksine karşılık gelen *E* ve *R* matrisleri içindeki satırlar $\vec{E}$ ($\overrightarrow{\Delta D}$ için yeniden yapılandırılan taşma vektörü) ve $\vec{R}$ (yeniden yapılandırılan $\overrightarrow{\Delta D}$) olarak alınır. Aynı zamanda, *T* matrisi içinde bu indekse karşılık gelen *Text* ise *T** yani örten metin olarak seçilmektedir. Bütün bu işlemlerin amacı LZW ve Huffman kodlamalarının performanslarını artırmaktır.

Modül 4- Sıkıştırma:

Seçilen $\vec{R}$, LZW ve Huffman kodlaması ile sıkıştırılır:

a) LZW kodlaması için; 1 - 26 arası sayılar başlangıçtaki LZW sözlüğünü oluşturmak için kullanılır. Bu kodlar hiç tekrar olmaması durumunda kullanılacaktır. Karşılaşılan her bir sembol ya da sembol dizisi için tekrar durumları da ele alınarak LZW sözlüğü güncellenir. Bu sembol, LZW sözlüğü içindeki ilgili indekse bakılarak kodlanır. LZW kodlaması sonucunda $\overrightarrow{R'}$ vektörü elde edilir ve $\left\|\overrightarrow{R'}\right\| < \|\vec{R}\|$ olmaktadır. $\overrightarrow{R'}$ vektörünün her bir elemanı iki tabanında ifade edilir; $(\overrightarrow{R'})_2$. İki tabanında ifade edilen her bir eleman birleştirilerek bir bit dizisi elde edilir.

b) Huffman kodlaması için; $\vec{R}$ içindeki 1-26 arası tüm sembollerin frekansları hesaplanarak $\vec{F}$ elde edilir. Bu frekanslara göre Huffman ağacı oluşturulur. Her bir sembol için Huffman ağacı kullanılarak kod kelimeleri çıkarılır. $\vec{R}$ Huffman ağacı üzerinden kodlanır. Huffman kodlaması sonucu, $(\overrightarrow{R'})_2$ şeklinde bir bit dizisi elde edilmektedir (Ayrıca $\left\|(\overrightarrow{R'})_{10}\right\| < \|\vec{R}\|$ olmaktadır).

Modül 5 – Kodlama:

a) Huffman kodlaması için; Modül 4 (b) hesaplanan frekans değerleri Latin karesi kullanılarak harflere çevrilmektedir. Bu harfler K_1 içinden e-posta adreslerinin seçimi için kullanılmaktadır. Bir e-posta adresinin seçimi için iki harf kullanıldığından, hesaplanan bu 26 frekans değerini saklamak için 13 e-posta adresi alınacaktır. Bu modülde sonraki işlemler LZW ve Huffman kodlamaları için aynı olmaktadır:

b) Elde edilen bit dizisi 12'lik gruplara ayrılmaktadır. Her bir gruptaki ilk 9 bit, e-posta adresinin kullanıcı adı kısmını belirlemede, son 3 bit ise e-posta adresinin uzantısını belirlemede kullanılmaktadır. G_1 ilk 9 bit olsun. Aşağıdaki işlemleri gerçekleştirerek iki tamsayı elde edilmektedir:

$$x = [(G_1)_{10}]\backslash 26 \qquad (3.6)$$

$$y = (G_1)_{10} \bmod 26 \tag{3.7}$$

Bu tamsayılar K_1 içinden e-posta adreslerinin seçimi için kullanılacaktır. K_1 önceden oluşturulmuş e-posta adreslerini içeren bir kümedir ve global stego anahtar olarak kullanılmaktadır. *x* ve *y*, Latin karesi üzerinden harflere dönüştürülür ve bu harfler ile K_1 içinden ilgili e-posta adresi seçilmektedir.

c) G_2 son 3 bit olsun:

$$z = (G_2)_{10} \tag{3.8}$$

Bu işlem ile belirlenen *z* sayısına göre *A* içinden ikili indeksler kullanılarak ilgili e-posta adres uzantısı seçilmektedir (Bu modifikasyon, K_2 kümesinin elemanlarının stego anahtar olarak düzenlenmesini sağlar.).

Modül 6 – Stego anahtar:

K_2 oluşumunun tamamlanması için e-posta adresleri, $\vec{E}$ vektörü kullanılarak modifiye edilmektedir. Bu modifikasyon; $\vec{E}$ elemanlarını, seçilen e-posta adresi içine "@" karakterinden önce ekleyerek gerçekleştirilmektedir. E-posta adreslerinin oluşturulmasında herhangi bir kural ya da kısıt olmadığı için eklenen bu rakamlar, adresin doğal bir parçası olarak görünmektedir. Böylece K_2, seçilen ve stego anahtar olarak düzenlenen e-posta adreslerinden oluşan bir küme olmaktadır ve $s(K_2) < s(K_1)$ olmaktadır.

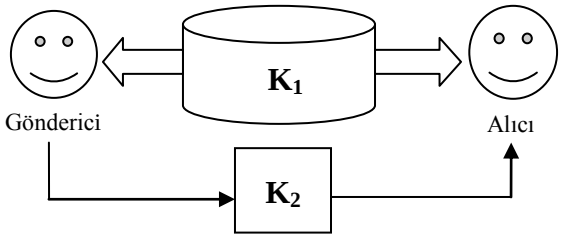
Modül 7 - Forward mail platformu:

T^* ve K_2 kullanılarak stego ortam, forward mail platformu şeklinde düzenlenir. Stego ortam içindeki belli e-posta adresleri (Modül 1(b)'de hesaplanan) taşma durumlarına göre modifiye edilerek stego anahtar olarak düzenlenmiştir ve global anahtar K_1 olmadan hangi K_2 elemanının stego anahtar olduğunu bulmak imkansızdır.

K_1 tüm işlemlerden önce yalnızca alıcı ve gönderici arasında paylaşılan bir global stego anahtardır.

3.5.2. Stego anahtar oluşumu ve kullanımı

Önerilen metotta, güvenliği artırmak amacıyla stego anahtar kullanılmaktadır. Stego anahtarların kullanımı ve iletimi Şekil 3.7' de gösterilmektedir. Görevlerine göre stego anahtarlar iki kategoride sınıflandırılabilir:



Şekil 3.7. Stego anahtarların kullanımı ve iletimi

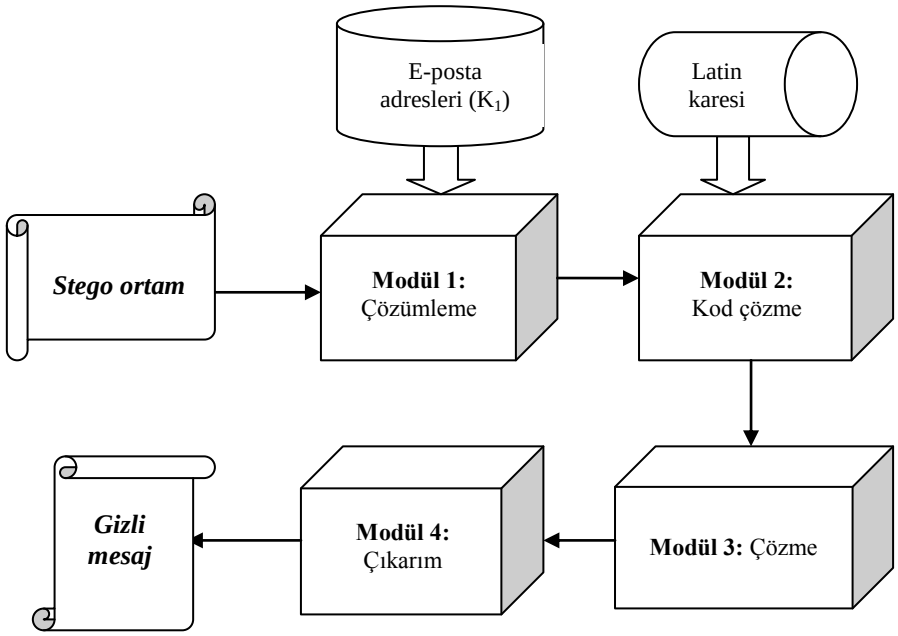
a) K_2; seçilen ve modifiye edilen e-posta adresleri kümesidir. E-posta adresindeki "@" karakterinden önce taşma bilgisi gömülerek ve A kümesi üzerinden z'ye göre e-posta adres uzantıları değiştirilerek oluşturulmuştur. K_2'nin oluşumu gömme aşamasında (Modül 4 ve Modül 5) gerçekleştirilmektedir. Saklı bilginin tam olarak doğru yerini muhafaza etmek amacıyla kullanılmaktadır. K_2'nin iletimi stego ortam içinde gerçekleştirilmektedir. K_2'nin elemanları seçilip modifiye edilmiş e-posta adresleridir ve bunlar forward mail olarak düzenlenen stego ortamın doğal bir parçasıdır. Bu e-posta simülasyon amaçlıdır. Yani alıcı dışında herhangi bir e-posta adresine gönderim gerçekleşmemektedir.

b) K_1; Lou ve ark. tarafından gerçekleştirilen "A novel adaptive steganography based on local complexity and human vision sensitivity" (Lou ve ark., 2010) ve Wang ve ark. tarafından gerçekleştirilen "Emoticon-based Text Steganography in Chat" (Wang ve ark., 2009b) çalışmalarında kullanılanlara benzer global bir stego anahtardır. Alıcı ve gönderici tarafından tüm işlemlerden önce paylaşılan ve e-posta adreslerinden oluşan

bir kümedir. Bu yolla, simetrik bir şifreleme tekniği kullanılarak saklı bilginin çıkarımı karmaşıklaştırılmıştır. Ayrıca, global anahtar kullanmanın bir diğer amacı, K_2 içine gömülen doğru konum bilgilerinin tespitidir. Bu nedenle, gizli mesajın doğru yer bilgilerinin tespiti için K_2'yi çözmek amacıyla global anahtara ihtiyaç duyulmaktadır.

3.5.3. Alıcı tarafı: Çıkarım aşaması

Şekil 3.8'de önerilen metodun çıkarım aşaması gösterilmektedir:



Şekil 3.8. Çıkarım aşamasının mimarisi

Modül 1 - Çözümleme:

Stego ortam alınarak her bir K_2 elemanı, her bir K_1 elemanı ile karşılaştırılır. Karşılaştırılan elemanların (e-posta adresleri) "@" karakterinden önceki kısımları farklı ise farklı rakamlar çıkarılır ve böylece her bir karşılaştırılan eleman için $\vec{E}$ vektörü adım adım oluşturulur. Fark yoksa $\vec{E}$ vektörünün ilgili eleman ya da elemanları 0 olarak atanır.

Modül 2 - Kod çözme:

a) LZW kodlaması için K_2 daha ayrıntılı olarak incelenir. Yani K_2 içindeki her bir elemanın (e-posta adresi) ilk iki karakterine bakılır ve bu karakterler Latin karesi ile rakamlara dönüştürülür. Böylece *x* ve *y* elde edilir. *A* üzerinden, e-posta adres uzantılarının indekslerine bakılarak *z* elde edilir. Bulunan *x* ve *z* değerleri ile her bir 12'lik bit grubu için:

$$G_1 = (x \cdot 26 + y)_2 \quad (3.9)$$

$$G_2 = (z)_2 \quad (3.10)$$

Bulunan G_1 ve G_2 değerleri birleştirilir ve böylece $(\vec{R}')_2$, yani LZW kodlaması sonucu sıkıştırılan bit dizisi elde edilir.

b) Huffman kodlaması için; K_2 daha ayrıntılı olarak incelenir. 14. elemandan itibaren (ilk 13 eleman $\vec{F}$ vektörünün elemanlarını içerdiğinden) K_2 içindeki her bir elemanın (e-posta adresi) ilk iki karakterine bakılır ve bu karakterler Latin karesi ile rakamlara dönüştürülür. Böylece *x* ve *y* elde edilir. *z* ise *A* üzerinden indeksler kullanılarak elde edilir. Bulunan *x* ve *z* değerleri ile her bir 12'lik bit grubu için:

$$G_1 = (x \cdot 26 + y)_2$$

$$G_2 = (z)_2$$

Bulunan G_1 ve G_2 değerleri birleştirilir ve böylece $(\vec{R}')_2$, yani Huffman kodlaması sonucu sıkıştırılan bit dizisi elde edilir.

Modül 3 – Çözme:

Bu modülde, $(\vec{R}')_2$, $\vec{R}$'nin çıkarımı için çözülmektedir:

a) LZW kodlaması için; 1 - 26 arası sayılar başlangıçtaki LZW sözlüğünü oluşturmak için kullanılır. Bu kodlar hiç tekrar olmaması durumunda kullanılacaktır. Karşılaşılan her bir sembol ya da sembol dizisi için tekrar

durumları da ele alınarak LZW sözlüğü güncellenir. Bu sembol LZW sözlüğü içindeki ilgili indekse bakılarak kodlanır.

b) Huffman kodlaması için; K_2'nin ilk 13 elemanı $\vec{F}$ vektörünün elemanlarını içermektedir. Her bir K_2 elemanının (e-posta adresinin) ilk iki karakteri Latin karesi üzerinden rakamlara dönüştürülür. Bu işlem K_2'nin tüm elemanları için yapılarak $\vec{F}$ elde edilir. Huffman ağacı oluşturulur. Böylece $(\vec{R}')_2$ içindeki her bir sembol için ilgili kod kelimesi oluşturulabilir.

Bu çözme işlemi sonucunda $\vec{R}$ elde edilmektedir.

Modül 4 – Çıkarım:

$\vec{R}$ ve $\vec{E}$ kullanılarak başlangıçtaki $\overrightarrow{\Delta D}$ hesaplanır. $\vec{R}$ ve $\vec{E}$ elemanları sırasıyla *r* ve *e* olsun. $\overrightarrow{\Delta D}$ elemanı olan *c*:

$$c = r + (26 \cdot e) \tag{3.11}$$

$\overrightarrow{\Delta D}$ elemanları kullanılarak stego ortamdaki *T** içinden *S* elemanları çıkarılır. *T** içinde her bir $\overrightarrow{\Delta D}$ elemanı olan *c* kadar ilerlenerek *a* çıkarılır. Çıkarılan bu *a* elemanları birleştirilerek gizli mesaj; *S* elde edilir.

4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA

Bu bölümde önerilen metot için performans ölçümleri gerçekleştirilmiştir. Önerilen metodun performansını belirleyen ölçütler kapasite ve güvenlik olarak belirlenmiştir. Tasarlanan steganografik bir metodun en önemli gereksinimleri algılanamazlık yanında kapasite, güvenlik ve sağlamlıktır. Bu bölümde önerilen metot kapasite güvenlik ölçümleri gerçekleştirilerek analiz edilmiştir.

4.1. Kapasite Analizi

Kapasite, gizli mesajın boyutunun, stego ortamın boyutuna oranı olarak ifade edilmektedir (Desoky, 2009):

$$C = \frac{Bits\ of\ Secret\ Message}{Bits\ of\ Stego\ Cover} \tag{4.1}$$

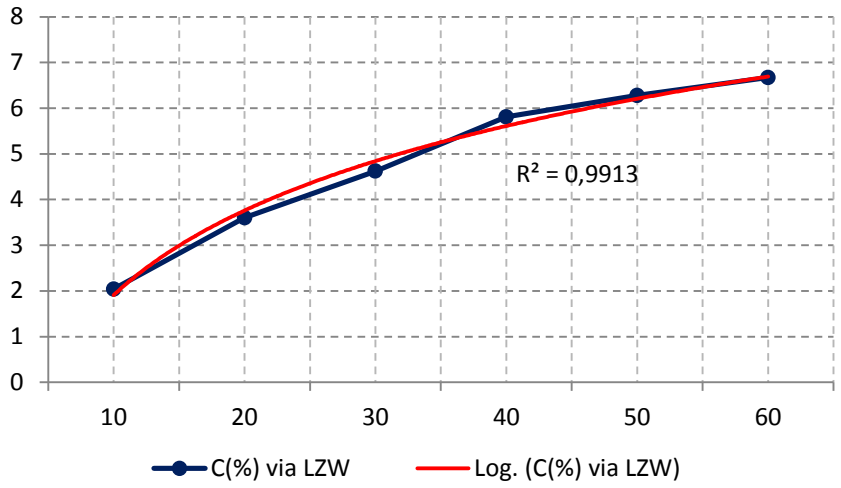
Çizelge 4.1'de örnek olması bakımından aşağıdaki gizli metin gizlenerek elde edilen bulgular verilmektedir:

"Steganography is the art and science of communicating in suc"

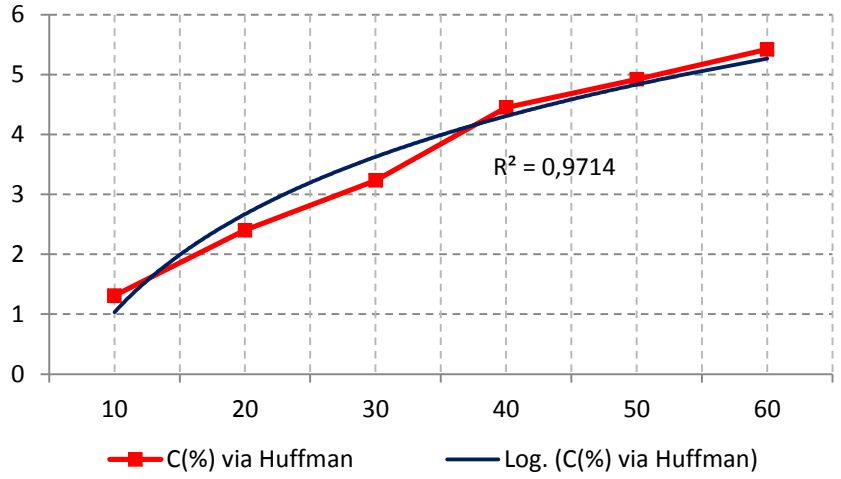
Çizelge 4.1. 6 örnek gizli mesaja ilişkin elde edilen $\vec{R}$,*n* ve *C* bulguları

	***S* (Gizli Mesaj)**	**$\vec{R}$(Yeniden yapılandırılan $\overrightarrow{\Delta D}$)**	***n***	***C(%)***	
				LZW	**Huffman**
S_1	Steganogra	(13, 8, 10, 14, 1, 1, 1, 1, 1,1)	10	2.04	1.31
S_2	Steganography is the	(13, 8, 10, 14, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 5, 7, 1, 1, 1, 1)	20	3.60	2.40
S_3	Steganography is the art and s	(13, 8, 10, 14, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 5, 7, 1, 1, 1, 1, 1, 3, 4, 25, 10, 6, 9, 22, 1, 9, 13)	30	4.62	3.23
S_4	Steganography is the art and science of	(12, 3, 2, 15, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 25, 19, 20, 6, 1, 1, 3, 1, 1, 22, 4, 1, 11, 7, 14, 5, 5, 15, 1, 1, 1, 3, 15, 2, 21, 8)	40	5.81	4.45
S_5	Steganography is the art and science of communicat	(12, 3, 2, 15, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 25, 19, 20, 6, 1, 1, 3, 1, 1, 22, 4, 1, 11, 7, 14, 5, 5, 15, 1, 1, 1, 3, 15, 2, 21, 8, 23, 1, 25, 10, 11, 6, 16, 3, 10, 7)	50	6.28	4.92
S_6	Steganography is the art and science of communicating in suc	(12, 3, 2, 15, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 25, 19, 20, 6, 1, 1, 3, 1, 1, 22, 4, 1, 11, 7, 14, 5, 5, 15, 1, 1, 1, 3, 15, 2, 21, 8, 23, 1, 25, 10, 11, 6, 16, 3, 10, 7, 11, 7, 6, 11, 25, 1, 12, 5, 16, 7, 18)	60	6.67	5.42

Burada, verilen metin karakter uzunluğu 10'ar artırılarak saklanmıştır. Böylelikle 6 parça elde edilmiştir. Her bir parça için yeniden yapılandırılan uzaklık dizisi ($\overrightarrow{R}$), karakter uzunluğu (*n*) ve kapasite (*C*) değerleri tabloda verilmektedir. Karakter sayısı arttıkça tekrar sayısı artacağından sıkıştırma performansı da artmaktadır. Sıkıştırma performansı arttıkça $\vec{R}$ den daha küçük bir sayı dizisi $\vec{R}'$ elde edilmekte ve buda Latin karesi kullanılarak elde edilen ve stego ortama eklenen e-posta adresi sayısını düşürmektedir. Denklem 4.1'e bakılacak olunursa bu durumun payda kısmında bulunan stego ortam boyutunu küçülttüğü ve bu yolla kapasite artışı sağladığı görülebilmektedir.



Şekil 4.1. LZW kodlaması ile kapasite-karakter uzunluğu ilişkisi

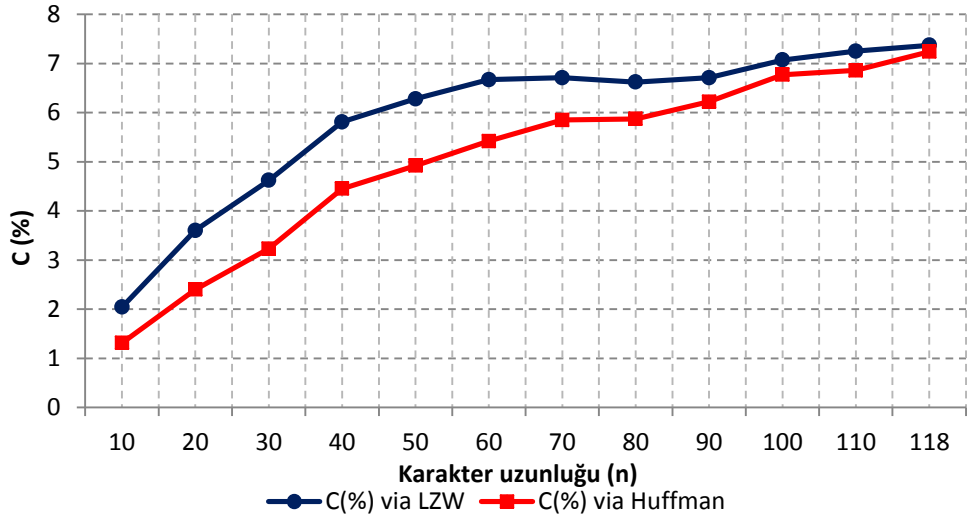


Şekil 4.2. Huffman kodlaması ile kapasite-karakter uzunluğu ilişkisi

Şekil 4.1 ve Şekil 4.2'de, kapasite ve karakter uzunluğu arasındaki ilişki her iki kodlama tekniği için ayrı grafiklerde sunulmuştur. Dikey eksen yüzde cinsinden kapasite değerlerini ($\%C$), yatay eksen ise karakter uzunluğunu (n) göstermektedir. Her iki kodlama tekniği içinde karakter uzunluğu arttıkça kapasitenin de arttığı görülmektedir. Ayrıca grafiklere logaritmik eğilim eğrisi de eklenerek kapasitenin beklenen artışı gösterilmiştir. R^2 değerinin 1'e çok yakın olmasından ötürü logaritmik eğri tercih edilmiştir.

Şekil 4.3'de LZW ve Huffman kodlama algoritmaları ile elde edilen kapasite değerleri karşılaştırma kolaylığı açısından birlikte verilmiştir. Başlangıçta Huffman kodlaması ile elde edilen kapasite değerlerinin LZW kodlaması ile elde edilen kapasite

değerlerinden daha düşük olduğu görülmektedir. Bunun nedeni, 26 harfe ait frekans bilgisinin stego ortama gömülmesi için 13 e-posta adresinin kullanımıdır. Ancak karakter sayısı artıkça bu durumun sebep olduğu dezavantajın azaldığı görülmüştür. Özellikle gizli mesaj 100 karaktere ulaştığında Huffman kodlaması ile elde edilen kapasite değerleri LZW kodlaması ile elde edilen değerlere yaklaşmaktadır. Sonunda gizli mesaj 118 karaktere sahip olduğunda kapasite değerlerinin birbirine çok yakın olduğu gözlemlenmektedir.



Şekil 4.3. LZW ve Huffman kodlamalarına göre kapasite-karakter uzunluğu grafiği

4.2. Güvenlik Analizi

Güvenlik, bir gözlemcinin saklı bilgiyi kolayca çıkarıp çıkaramaması ile ilgilidir. Güvenliği desteklemek amacıyla Kombinatorik tabanlı kodlama (Latin Karesi) ve sıkıştırma algoritmaları (LZW ve Huffman) kullanılmıştır. Kombinatorik tabanlı kodlama ile dışarıya arzu edilen rastgelesellik sağlanırken sıkıştırma ile bilginin çıkarım aşaması daha da karmaşıklaştırılmıştır. Güvenlik analizi şu şekilde gerçekleştirilmektedir:

Önerilen metodun herkesçe bilindiği yani dışarıya açık olduğu varsayılmaktadır ancak bölüm 3.5.2'de de değinildiği gibi global anahtar sadece gönderici ve alıcıda mevcuttur. Bu durumda gizli mesajı çıkarmak için oluşturulması gereken kombinasyon sayısı hesaplanmaktadır.

LZW kodlaması için formülasyon şu şekildedir:

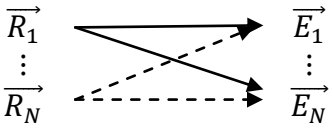
$$LZW_C = 8^N \times \prod_1^N 2^m \quad (4.2)$$

İspat: Global anahtar yok iken gözlemci tarafından gizli mesajı çıkarmak için yapılması gereken işlemler üç adımda açıklanabilir:

1. Öncelikle gözlemci sıkıştırılan $(\overrightarrow{R'})_2$ bit dizisini elde etmelidir. Bunun için gözlemcinin *x*, *y* ile G_1'i ve z ile de G_2'yi elde etmesi gerekmektedir. *x* ve *y*, Latin karesi ile her bir e-posta adresinin ilk iki harfinin sayı karşılıkları bulunarak elde edilebilir. *z* ise e-posta adresi uzantısına bakılarak çıkarılabilmektedir. Ancak gözlemcinin *z* için e-posta adresi başına 8 kombinasyon yapması gerekmektedir. Bu durumda *N* tane e-posta adresi varsa gözlemci 8^N kombinasyon oluşturmalıdır. Daha sonra her bir *z*, 2 tabanında ifade edilerek G_1 ve G_2'nin çıkarılması için *x* ve *y* ile birleştirilir. Kısaca doğru bit dizisi $(\overrightarrow{R'})_2$ vektörünü bulabilmek için 8^N kombinasyon oluşturulmalıdır.
2. $(\overrightarrow{R'})_2$ çözülerek $\vec{R}$ elde edilir. (Bölüm 3.5.1 modül 2 ve bölüm 3.5.3, modül 4):

   $$\overrightarrow{\Delta D} = (\vec{R} \times 26) + \vec{E} \quad (4.3)$$

   Her bir e-posta adresinin rakam içerip içermediğine bakılır. E-posta adresinin rakam içermesi durumunda, bu rakamlar $\vec{E}$ elemanlarının bulunması için ele anılır. Yani bu rakamlar ilgili e-posta adresine ait olabileceği gibi gömme aşamasında $\vec{E}$'nin elemanlarını gizlemek için ilgili e-posta adresine eklenmiş de olabilir. Eğer bir e-posta adresindeki rakam sayısını *m* ile ifade edecek olursak, alt küme teorisinden de hatırlanacağı üzere, gözlemcinin her bir e-posta adresi için 2^m kombinasyon gerçekleştirmesi gerekmektedir.
3. Elde edilen her bir $\vec{R}$ ve $\vec{E}$ doğru gizli mesajı bulmak amacıyla kendi aralarında denenmelidir:



Böylece LZW kodlaması için önerilen metodun karmaşıklığı şu şekilde formüle edilebilir:

$$LZW_C = 8^N \times \prod_1^N 2^m$$

Huffman kodlaması için formülasyon şu şekildedir:

$$H_C = 8^{N-13} \times \prod_1^N 2^m \quad (4.4)$$

İspat: Global anahtar yok iken gözlemci tarafından gizli mesajı çıkarmak için yapılması gereken işlemler üç adımda açıklanabilir:

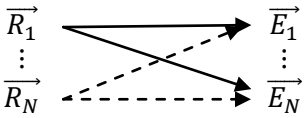
1. Öncelikle gözlemci, sıkıştırılan $(\overrightarrow{R'})_2$ bit dizisini elde etmelidir. Bunun için gözlemcinin *x, y* ile G_1'i ve z ile de G_2'yi elde etmesi gerekmektedir. *x* ve *y*, Latin karesi ile her bir e-posta adresinin ilk iki harfinin sayı karşılıkları bulunarak elde edilebilir. *z* ise e-posta adresi uzantısına bakılarak çıkarılabilmektedir. Hatırlanacağı üzere Huffman kodlamasında ilk 13 e-posta adresi yalnızca frekans bilgilerini ($\vec{F}$) içermekte ve bu nedenle uzantıları değişmemekteydi (Bölüm 3.5.1, Modül 4). Bu nedenle gözlemcinin *N* tane e-posta adresi için 8^{N-13} kombinasyon gerçekleştirmesi gerekmektedir. Daha sonra her bir *z*, 2 tabanında ifade edilerek G_1 ve G_2'nin çıkarılması için *x* ve *y* ile birleştirilir. Kısaca doğru bit dizisi $(\overrightarrow{R'})_2$ vektörünü bulabilmek için 8^{N-13} kombinasyon oluşturulmalıdır.
2. $(\overrightarrow{R'})_2$ çözülerek $\vec{R}$ elde edilir (Bölüm 3.5.1, modül 2 ve bölüm 3.5.3, modül 4):

$$\overrightarrow{\Delta D} = (\vec{R} \times 26) + \vec{E}$$

Her bir e-posta adresinin rakam içerip içermediğine bakılır. E-posta adresinin rakam içermesi durumunda, bu rakamlar $\vec{E}$ elemanlarının bulunması için ele anılır. Yani bu rakamlar ilgili e-posta adresine ait olabileceği gibi gömme aşamasında $\vec{E}$'nin elemanlarını gizlemek için ilgili

e-posta adresine eklenmiş olabilir. Eğer bir e-posta adresindeki rakam sayısını *m* ile ifade edecek olursak, alt küme teorisinden de hatırlanacağı üzere, gözlemcinin her bir e-posta adresi için 2^m kombinasyon gerçekleştirmesi gerekmektedir.

3. Elde edilen her bir $\vec{R}$ ve $\vec{E}$ doğru gizli mesajı bulmak amacıyla kendi aralarında denenmelidir:



Böylece Huffman kodlaması için önerilen metodun karmaşıklığı şu şekilde formüle edilebilir:

$$H_C = 8^{N-13} \times \prod_1^N 2^m$$

Şekil 4.4 ve Şekil 4.5'te LZW ve Huffman kodlama teknikleri ile aşağıda verilen mesajın gizlenmesi sonucu oluşturulan stego ortamlar gösterilmektedir:

"visual degradation a"

----Original Message----
From: sender@hotmail.com
Date: Monday, September 27, 2010 8: 38 am
Subject: Notification
To:recipient@hotmail.com
CC:gkorkut_ozan010@yahoo.com, snermin_karma200@myspace.com, ckorkmaz@yahoo.com, hlale_ergun@yahoo.com, gzemm_derman@msn.com, kgizemcaneli@msn.com, gquarters@yahoo.com, hpeter_urban@gmail.com, ihsantopkara@gmail.com

> Dear participants;
>We would like to inform you about our recently added and organized new web sites. They contain very useful
>information about text steganography. We hope for your join. Please do not hesitate for extra questions.
>Best regards

Şekil 4.4. LZW kodlaması kullanılarak oluşturulan stego ortam

----Original Message----

From: sender@hotmail.com

Date: Monday, September 27, 2010 8: 38 am

Subject: Notification

To: recipient@hotmail.com

CC: ece_namli400@hotmail.com, dbetul_arici103@hotmail.com, ccansel_ugur010@hotmail.com, ddundar_cetin010@hotmail.com, ggencer200@hotmail.com, gizem_gursel@hotmail.com, ghost99@hotmail.com, hilmi_ongun@hotmail.com, ijar99@hotmail.com, jjanuary@hotmail.com, kkuyumcu@hotmail.com, mlorridharma@hotmail.com, mmoriss@hotmail.com, qleslie@mail.com, hvedat_oguz@msn.com, lburhanmevki@mynet.com, wlenny@mail.com, oayhan@yahoo.com, hvedat_oguz@hotmail.com

> Dear participants;

>We would like to inform you about our recently added and organized new web sites. They contain very useful

>information about text steganography. We hope for your join. Please do not hesitate for extra questions.

>Best regards

Şekil 4.5. Huffman kodlaması kullanılarak oluşturulan stego ortam

Şekil 4.4 ve Şekil 4.5'te verilen stego ortamlardaki gizli mesajı çıkarmak için Denklem 4.2 ve Denklem 4.4 kullanılarak oluşturulması gereken kombinasyon sayıları şu şekilde hesaplanmaktadır:

$$LZW_C = 8^N \times \prod_1^N 2^m = 8^9 \times 2^3 \times 2^3 \times 2^0 \times 2^0 \times 2^0 \times 2^0 \times 2^0 \times 2^0$$
$$= 8^9 \times 2^6 = 85 \times 10^8$$

$$H_C = 8^{19-13} \times \prod_1^N 2^m = 8^6 \times 2^3 \times 2^3 \times 2^3 \times 2^3 \times 2^3 \times 2^0 \times 2^2 \times 2^0 \times 2^2 \times 2^0 \times 2^0 \times 2^0 \times 2^0 \times 2^0 \times 2^0 \times 2^0 \times 2^0 \times 2^0$$
$$= 8^6 \times 2^{19}$$
$$= 137 \times 10^9$$

Karmaşıklık sayısı, Latin karesinin başladığı harf bilgisi gizli tutularak, LZW ve Huffman kodlamalarının her ikisi içinde 26 kat artırılabilmektedir:

$$MaxLZW_C = 26 \times 8^N \times \prod_1^N 2^m \quad (4.5)$$

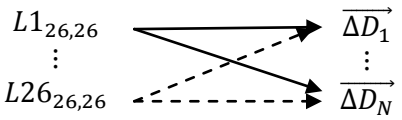
$$MaxH_C = 26 \times 8^{N-13} \times \prod_1^N 2^m \quad (4.6)$$

İspat: Latin karesini gömme aşamasında harf karşılıklarını bulmak amacıyla (bölüm 3.5.1, modül 5) ve çıkarım aşamasında sayı karşılıklarını bulma amacıyla (bölüm 2.3,

modül 2) kullanmaktayız. Latin karesi 26×26'lık bir matristir (bkz. EK-1). Latin karesini sırasıyla "A, B, C,..., Z" harfleri ile başlatmamız durumunda:

A:					B:					C:					...					Z:				
A	B	C	…	Z	B	C	D	…	A	C	D	E	…	B	…	Z	A	B	…	Y
B	C	D	…	A	C	D	E	…	B	D	E	F	…	C	...	A	B	C	…	Z
.	.	.		.	.	.	.		.	.	.	.		.		.	.	.		.
.	.	.		.	.	.	.		.	.	.	.		.		.	.	.		.
.	.	.		.	.	.	.		.	.	.	.		.		.	.	.		.
Z	A	B	…	Y	A	B	C	…	Z	B	C	D	…	A	…	Y	Z	A	…	X

Buna göre elimizde 26 tane farklı Latin karesi mevcut olacaktır. Gizli mesajın çıkarımı için her bir Latin karesinin, e-posta adreslerinin kullanıcı ve uzantı kısımları sınanarak elde edilen her bir sayı dizisi kombinasyonu için değerlendirilmesi gerekmektedir (Her bir sayı dizisi kombinasyonunu bir $\overrightarrow{\Delta D}$ adayı olduğunu fark ediniz):



Böylece, LZW ve Huffman kodlamaları için maksimum karmaşıklık şu şekilde formüle edilir:

$$MaxLZW_C = 26 \times LZW_C = 26 \times 8^N \times \prod_1^N 2^m$$
$$MaxH_C = 26 \times H_C = 26 \times 8^{N-13} \times \prod_1^N 2^m$$

Şekil 4.4 ve Şekil 4.5'te verilen stego ortamlardaki gizli mesajı çıkarmak için denklem 4.5 ve denklem 4.6 kullanılarak oluşturulması gereken maksimum kombinasyon sayıları şu şekilde hesaplanmaktadır:

$$LZW_C = 8^N \times \prod_1^N 2^m = 8^9 \times 2^{3\times}2^3 \times 2^0 \times 2^0 \times 2^0 \times 2^0 \times 2^0 \times 2^0 \times 2^0$$
$$= 8^9 \times 2^6 = 85 \times 10^8$$

$$MaxLZW_C = 26 \times 8^N \times \prod_1^N 2^m = 26 \times 8^9 \times 2^6 = 223 \times 10^9$$

$$H_C = 8^{19-13} \times \prod_1^N 2^m = 8^6 \times 2^3 \times 2^3 \times 2^3 \times 2^3 \times 2^3 \times 2^0 \times 2^2 \times 2^0 \times 2^2 \times 2^0 \times 2^0 \times 2^0 \times 2^0 \times 2^0 \times 2^0 \times 2^0 \times 2^0 \times 2^0$$

$$= 8^6 \times 2^{19} = 137 \times 10^9$$

$$MaxH_C = 26 \times 8^{N-13} \times \prod_1^{\mathrm{N}} 2^{\mathrm{m}} = 26 \times 8^6 \times 2^{19} = 357 \times 10^{10}$$

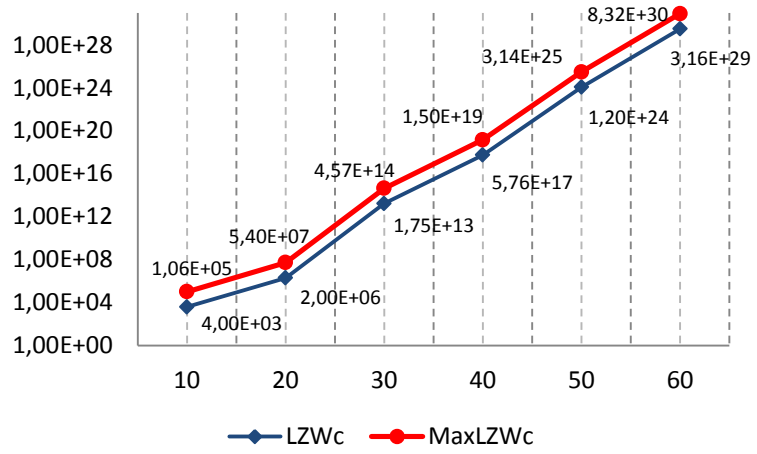
Çizelge 4.2'de örnek olması bakımından aşağıdaki gizli metin gizlenerek elde edilen karmaşıklık ve maksimum karmaşıklık bulguları verilmektedir:

" Steganography is the art and science of communicating in suc"

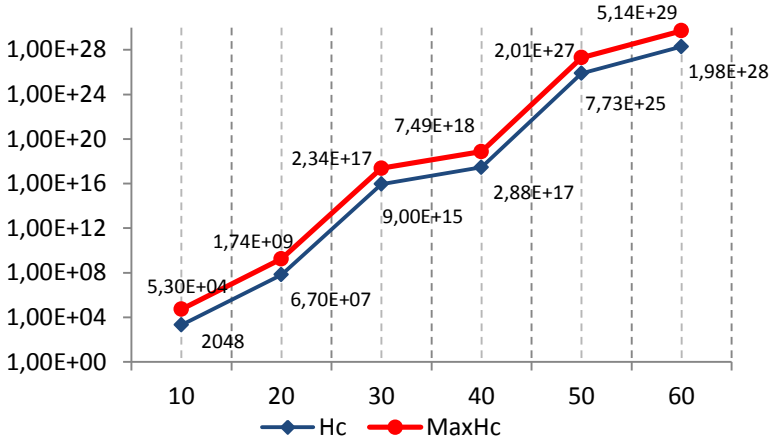
Çizelge 4.2. 6 örnek gizli mesaja ilişkin edilen karmaşıklık ve maksimum karmaşıklık bulguları

	S (Gizli Mesaj)	***n***	***Karmaşıklık (Complexity)***			
			LZW_C	$MaxLZW_C$	H_C	$MaxH_C$
S_1	Steganogra	10	40×10^2	106×10^3	2048	53×10^3
S_2	Steganography is the	20	20×10^5	54×10^6	67×10^6	174×10^7
S_3	Steganography is the art and s	30	175×10^{11}	457×10^{12}	900×10^{13}	2341×10^{14}
S_4	Steganography is the art and science of	40	576×10^{15}	1498×10^{16}	288×10^{15}	749×10^{16}
S_5	Steganography is the art and science of communicat	50	120×10^{22}	314×10^{23}	773×10^{23}	2011×10^{24}
S_6	Steganography is the art and science of communicating in suc	60	316×10^{27}	832×10^{28}	198×10^{26}	514×10^{27}

Gizlenecek metnin uzunluğu 10'ar artırılmıştır. Böylece elimizde 6 parça olmaktadır. Çizelge 4.2'den yola çıkılarak karakter sayısı arttıkça hem LZW hem de Huffman kodlaması ile gerçekleştirilen metotta karmaşıklık ve maksimum karmaşıklığın arttığı sonucuna varılabilmektedir. Şekil 4.6 ve Şekil 4.7'de bu bulgular grafiksel olarak sunulmaktadır. Grafiklerde dikey eksen karmaşıklık değerini, yatay eksen ise karakter uzunluğunu (*n*) göstermektedir. Gizlenen mesajın karakter uzunluğu arttıkça her iki kodlama tekniği içinde karmaşıklık artmaktadır.



Şekil 4.6. LZW kodlaması ile karmaşıklık-karakter uzunluğu ilişkisi



Şekil 4.7. Huffman kodlaması ile karmaşıklık-karakter uzunluğu ilişkisi

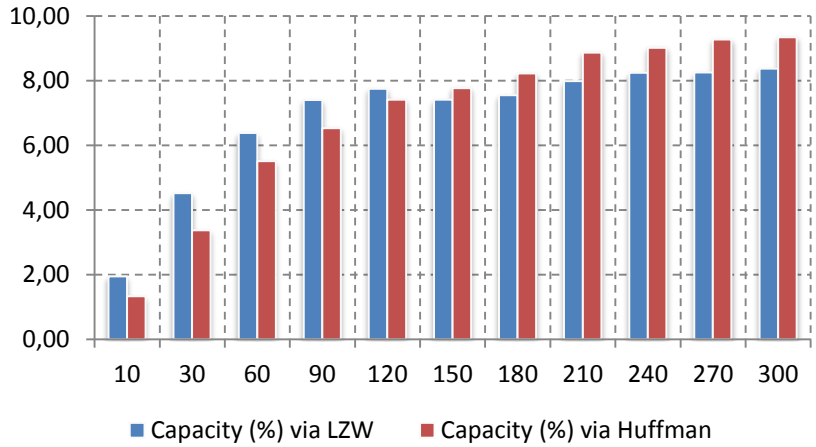
4.3. Deneysel Sonuçlar

Bu bölümde LZW ve Huffman kodlama teknikleri ile elde edilen kapasite ve karmaşıklık değerleri açıklanacak ve karşılaştırılacaktır. Winstein veri tabanı kullanılarak elde edilen bulgular Çizelge 4.3'te sunulmuştur (Winstein K, Lexical steganography, http://alumni.imsa.edu/~keithw/tlex, 24 Temmuz 2012).

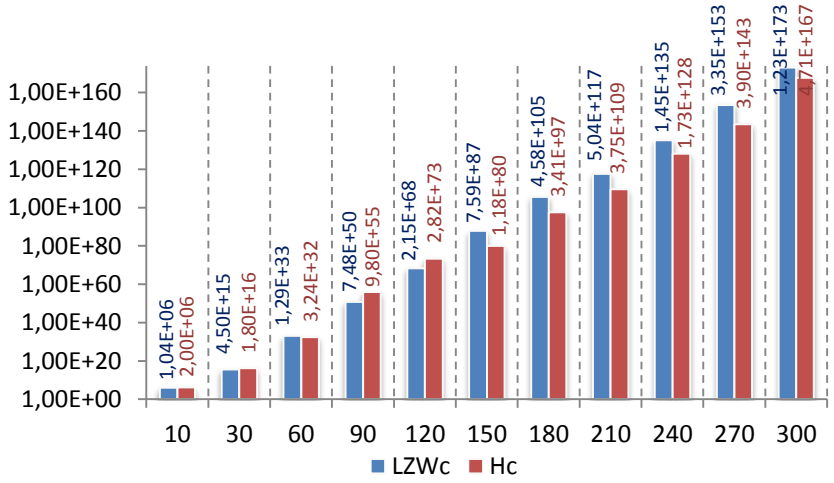
Çizelge 4.3. Winstein veri tabanı kullanılarak elde edilen deneysel bulgular

	n	**Kapasite *(%C)***		**Karmaşıklık**		**Maksimum Karmaşıklık (×26)**	
		LZW	***Huffman***	LZW_C	H_C	$MaxLZW_C$	$MaxH_C$
1	10	1.95	1.34	104×10^{4}	20×10^{5}	272×10^{5}	54×10^{6}
2	30	4.52	3.37	450×10^{13}	180×10^{14}	117×10^{15}	468×10^{15}
3	60	6.38	5.51	129×10^{31}	324×10^{30}	337×10^{32}	843×10^{31}
4	90	7.40	6.53	748×10^{48}	980×10^{53}	1945×10^{49}	2550×10^{54}
5	120	7.75	7.41	215×10^{66}	282×10^{71}	560×10^{67}	735×10^{72}
6	150	7.41	7.77	759×10^{85}	118×10^{78}	2068×10^{86}	308×10^{79}
7	180	7.55	8.22	458×10^{103}	341×10^{95}	1192×10^{104}	888×10^{96}
8	210	7.99	8.87	504×10^{115}	375×10^{107}	1311×10^{116}	976×10^{108}
9	240	8.24	9.02	145×10^{133}	173×10^{126}	377×10^{134}	450×10^{127}
10	270	8.25	9.27	335×10^{151}	390×10^{141}	871×10^{152}	1014×10^{142}
11	300	8.37	9.34	123×10^{171}	471×10^{165}	321×10^{172}	1226×10^{166}

Şekil 4.8 ve Şekil 4.9'da ise LZW ve Huffman kodlama teknikleri ile elde edilen kapasite ve karmaşıklık değerleri karşılaştırma kolaylığı açısından grafiksel olarak gösterilmektedir. Şekil 4.8'de yatay eksen karakter uzunluğunu, dikey eksen ise yüzde cinsinden kapasite değerlerini göstermektedir. Şekil 4.9'da ise dikey eksen karmaşıklık değerlerini, yatay eksen karakter uzunluğunu göstermektedir. LZW ve Huffman kodlaması ile elde edilen kapasite değerlerine bakıldığında başlangıçta Huffman kodlaması ile elde edilen değerlerin daha düşük olduğu görülmektedir. Bunun nedeni Huffman kodlaması ile karşı tarafa göndermek zorunda olduğumuz frekans bilgileri için 13 e-posta adresinin kullanılması ve bunlarında stego ortama ek kapasite yükü getirmesidir. Ancak özellikle 120 karakter ve sonrasında bu durum ortadan kalkmış ve Huffman kodlaması ile elde edilen kapasite değerlerinin LZW kodlaması ile elde edilen kapasite değerlerinden daha fazla olduğu görülmüştür. Bunun nedeni ise, Huffman kodlamasının sembol frekanslarına göre ikili ağaç yapısında sıkıştırmayı gerçekleştirmesidir. Bu durum sonucu en çok tekrara sahip sembol için en kısa, en az tekrara sahip sembol için ise en uzun kod kelimesi kullanılmaktadır. Bu durum Huffman kodlamasının performansını artırmaktadır. LZW kodlamasında ise karşı tarafa herhangi bir bilgi gönderilmediği için başlangıçta kapasite değerleri yüksektir ancak burada sembollerin frekans değerleri yerine sözlük oluşturularak ve karşılaşılan her bir sembol için bu sözlük güncellenip kodlama yapıldığından Huffman sıkıştırmasındaki kadar kapasite artışı söz konusu olmamaktadır.



Şekil 4.8. Winstein veri tabanı ile elde edilen kapasite-karakter uzunluğu ilişkisi



Şekil 4.9. Winstein veri tabanı ile elde edilen karmaşıklık-karakter uzunluğu ilişkisi

5. SONUÇLAR VE ÖNERİLER

Bu bölümde, önerilen metot literatürdeki diğer metotlar ile karşılaştırılmıştır. Karşılaştırma ölçütü olarak literatür esas alınarak kapasite seçilmiştir. Sonrasında ise önerilen metoda ilişkin avantaj ve dezavantajlara değinilmiş ve genel bir yargıya ulaşılmıştır.

5.1. Değerlendirme Sonuçları

Bu bölümde, önerilen metodun canlandırılması amacıyla örnek bir gizli mesaj ve bu mesajı saklamak için oluşturulan stego ortam verilerek, önerilen metot, literatürdeki diğer metotlar ile karşılaştırılmıştır. Karşılaştırma amacıyla deneyler Microsoft Windows Vista işletim sistemine sahip, Intel(R) Core (TM)2 Duo, 1.66 GHz işlemcili ve 2 GB RAM bulunan bir bilgisayarda gerçekleştirilmiştir.

Önceki bölümde de anlatıldığı gibi önerilen metot kapasite ve güvenlik açısından analiz edilmiştir. Ancak literatürdeki en yaygın ölçüm parametresi olduğundan ötürü karşılaştırma amacıyla kapasite esas alınmıştır. Çizelge 5.1'de önerilen metotlar, literatürdeki diğer güncel metotlar ile kapasite açısından karşılaştırılmıştır. Karşılaştırma, tırnak işaretleri hariç, boşluklar dahil aşağıdaki 200 karakterlik mesaj kullanılarak gerçekleştirilmiştir:

"behind using a cover text is to hide the presence of secret messages the presence of embedded messages in the resulting stego text cannot be easily discovered by anyone except the intended recipient. "

Çizelge 5.1. Karşılaştırma sonuçları

Metot	C(%) (Kapasite)	Açıklama
Mimic fonksiyonları (Wayner 1992, Wayner 2002)	1.27	Verilen örnek mesaj kullanılarak hesaplanmıştır. (http://www.spamimc.com)
NICETEXT (Chapman ve Davida, 1997; Chapman ve Davida, 2001; Chapman ve Davida, 2002)	0.29	İlgili makalelerdeki örneklerden sağlanmıştır.
Winstein (Winstein, 1999; Winstein 2002)	0.5	İlgili makalelerdeki örneklerden sağlanmıştır.
Murphy ve ark. (Murphy ve Vogel, 2007)	0.30	İlgili makalede rapor edilmiştir.
Nakagava ve ark. (Nakagawa ve ark., 2001)	0.12	İlgili makalede rapor edilmiştir.
Çeviri tabanlı (Stutsman ve ark., 2006)	0.33	İlgili makalede yazarlarca rapor edilmiştir.
Confusing (Topkara ve ark., 2007)	0.35	İlgili makalelerdeki örneklerden sağlanmıştır.
L-R metodu (Sun ve ark., 2004)	2.17	Wang et al., 2009 çalışmasındaki örneğe dayanarak UNICODE formatında hesaplanmıştır.
Wang ve ark. (Wang ve ark., 2009a)	3.53	Wang et al., 2009 çalışmasındaki örneğe dayanarak UNICODE formatında hesaplanmıştır.
Listega (Desoky, 2009)	3.87	İlgili makalelerdeki örnekten sağlanmıştır
TEXTO (Maher, 1995)	6.91	Verilen örnek mesaj kullanılarak hesaplanmıştır. (http://www.eberl.net/cgi-bin/stego.pl)
Satir ve Isik; güncelleme modülü olmadan uzun metinler ile LZW kodlama (Satir ve Isik 2012a)	6.92	Verilen örnek mesaj kullanılarak hesaplanmıştır.
Satir ve Isik; güncelleme modülü olmadan uzun metinler ile Huffman kodlama (Satir ve Isik 2012b)	7.017	Verilen örnek mesaj kullanılarak hesaplanmıştır.
Kısa metinler ile LZW tabanlı metot	8.15	Verilen örnek mesaj kullanılarak hesaplanmıştır.
Kısa metinler ile Huffman tabanlı metot	8.90	Verilen örnek mesaj kullanılarak hesaplanmıştır.

Ortanımla kriptolama şeklinde çalışan TEXTO ve dilbilgisi açısından doğru ancak anlamsal bakımdan oldukça eksik metinler üreten mimic fonksiyon gibi ulaşılabilen metotların kapasiteleri yukarıda verilen örnek metin kullanılarak hesaplanmıştır.

Eşanlam tabanlı yaklaşımlar olan Nicetext ve Winstein metotlarına ilişkin kapasite değerleri ilgili makalelerdeki örneklere dayanılarak verilmiştir. Diğer eşanlam tabanlı yaklaşımlar olan Murphy ve Nakagava'nın metotlarının kapasite değerleri de, ilgili makalelerden yararlanılarak verilmektedir. Makine çevirisinde doğal olarak görülen ve karşılaşılan hatalara veri gizleyen Stutsman'ın metodunun kapasite değeri de ilgili makaleye dayanılarak verilmiştir. Diğer bir çeviri tabanlı metot olan Topkara'nın metodunun kapasite değeri ise ilgili makaledeki örneklere dayanılarak verilmektedir.

L-R metodu ve Wang ve ark.'nın metodunun kapasite değerleri Wang ve ark. (Wang ve ark., 2009) tarafından gerçekleştirilen çalışmadaki örneklerden yararlanılarak, Çince'de uygulandıklarından ötürü, Unicode formatında hesaplanmıştır. Veriyi metinsel listelerden yararlanarak kamufle eden Listega metodunun kapasite değeri ise, ilgili makaledeki örneğe dayanılarak verilmiştir.

Daha uzun metinler kullanılarak ve metin tabanının güncellenmesi yapılmadan LZW ve Huffman kodlaması ile gerçekleştirilen metotların kapasite değerleri de verilen gizli mesaj kullanılarak hesaplanmıştır.

Son olarak daha kısa metinler kullanılarak gerçekleştirilen LZW ve Huffman kodlama tabanlı metin steganografisi metotlarının kapasite değerleri de verilen örnek gizli mesaj kullanılarak hesaplanmıştır.

Uzun ve kısa metinler kullanılarak önerilen LZW ve Huffman kodlama tabanlı metin steganografi metotlarında stego ortam doğal olarak üretilen örten metin ve e-posta adreslerinden oluşmaktadır. Şekil 5.1 ve Şekil 5.2'de, verilen örnek gizli mesaj kullanılarak oluşturulan stego ortamlar gösterilmektedir. Çizelge 5.1'de verilen kapasite değerleri Denklem 4.1 kullanılarak hesaplanmıştır. Çizelge 5.1'e göre kısa metinler kullanılarak önerilen LZW ve Huffman kodlama tabanlı metin steganografi metotlarının kapasiteleri sırasıyla %8.15 ve %8.90 değerlerine ulaşmıştır. Böylelikle gizlenecek karakter sayısının artmasının kapasite üzerinde sebep olduğu dezavantaj, avantaja çevrilmiştir.

----Original Message----

From: sender@hotmail.com

Date: Monday, September 27, 2010 8: 38 am

Subject: Notification

To:recipient@hotmail.com

CC:oerkan@myspace.com, qdaren@gmail.com, mxedos010@windowslive.com, duru_bozdemir@windowslive.com, emrah_tutkun010@gmail.com, gfergie65001@yahoo.com, nserhan001@mail.com, fqashqai@hotmail.com, iquitos200@mail.com, rjeff_arden@mail.com, ukora@mail.com, lvalley@windowslive.com, murat_karahan004@windowslive.com, pzuhre_kocatas002@windowslive.com, vgiresun002@windowslive.com, vquinton@mynet.com, guldenguzel88300@windowslive.com, wwwfranky@yahoo.com, jbieb113@windowslive.com, djake_glen@yahoo.com, gnelson_88010@hotmail.com, gxeribi401@windowslive.com, wayne_robin010@gmail.com, lwyona92410@windowslive.com, jgleeyn005@windowslive.com, zile_tuna@windowslive.com, bcengiz004@hotmail.com, dstevelarsson200@hotmail.com, fqashqai001@yahoo.com, dharmanci100@myspace.com, jennifercane43@hotmail.com, yilmazoztas001@windowslive.com, greg_colt@mynet.com, hserhan_kunduraci001@hotmail.com, irem_ilay001@myspace.com, kxylander@windowslive.com, uerdem100@yahoo.com, bwolf66@hotmail.com, xfizalya@mail.com, qxavier@myspace.com, uparla@yahoo.com, qerkin88@yahoo.com, czeynep_onal78@gmail.com, rvahide_sorgun@hotmail.com, brown_stan72@mail.com, bleonard@msn.com, cveliaydogdu@msn.com, ffred_dawson@hotmail.com, nhilal_iscan@windowslive.com, iayhan@yahoo.com, ggencer@hotmail.com, nurdan_akcan@hotmail.com, qcady@yahoo.com, farukbilgic87@msn.com, oquente@hotmail.com, msophie@hotmail.com, farukbilgic87@gmail.com, fnermin_ozsoy@hotmail.com, hmahmut_alan@gmail.com, iwona1988@hotmail.com, igulmez82@hotmail.com, uhdeaksan@hotmail.com, karla_sonic@windowslive.com, zj_wattson@hotmail.com, mkemal_sari@yahoo.com, nrakel@hotmail.com, utkuardic@hotmail.com, eerbil_72@mail.com, skyblue@gmail.com, fmermerci1980@windowslive.com, zuhrekalecik@mynet.com, eerbil_72@gmail.com, vquinton@gmail.com, bdundar@yahoo.com, auslu_erkin@yahoo.com, lwyona92@hotmail.com, banu_agca@windowslive.com, zzuleyha@mail.com, gxeribi@yahoo.com, pwilson@windowslive.com, ewan_mcgregor@msn.com, koraycelik@windowslive.com, rberry34@mail.com, fquinet@hotmail.com, kgizemcaneli@windowslive.com, lguldenozhan@windowslive.com, qisk_roll@gmail.com, jmax_77@windowslive.com, lguldenozhan@windowslive.com, uhdeaksan@hotmail.com

>Dear collegues;

>You can find extra information for figes in the given link: http://www.figes.com.tr/argegunleri/kprogram.php

>Best regards.

Şekil 5.1. Örnek gizli mesaj kullanılarak LZW kodlaması ile oluşturulan stego ortam (Forward mail platformu)

----Original Message----

From: sender@hotmail.com

Date: Monday, September 27, 2010 8: 38 am

Subject: Notification

To:recipient@hotmail.com

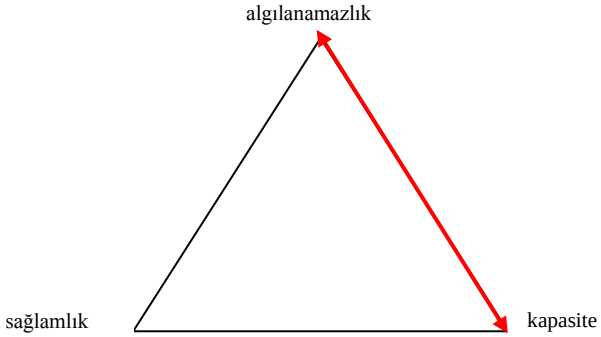
CC:mmoriss402@gmail.com, rzinnur_bademci002@hotmail.com, qerkin88@hotmail.com, llindsay001@hotmail.com, llindsay010@hotmail.com, imrankahraman002@hotmail.com, jillian_jones010@hotmail.com, johnthompson001@hotmail.com, jnanee@hotmail.com, pmaden@hotmail.com, normasurixx112@hotmail.com, pniyazigil020@hotmail.com, srabia_soyder011@hotmail.com, tdoganay@mail.com, mpersy@mynet.com, fyelda_güldere012@windowslive.com, thelvaci@hotmail.com, tnazanaslim001@myspace.com, ymeteboztas010@hotmail.com, yukselerten@hotmail.com, yesimcebel100@windowslive.com, tnazanaslim100@hotmail.com, pzuhre_kocatas@yahoo.com, cberkay110@msn.com, orhan_tekin@windowslive.com, txbox1990@hotmail.com, rx_media77120@windowslive.com, ygorgulu@yahoo.com, vcezayir@gmail.com, gbirsu300@msn.com, vj_albert@gmail.com, erdem_sengul@hotmail.com, ddundar_cetin010@gmail.com, ingrid_michael@mynet.com, williammoore010@myspace.com, jmax_77001@myspace.com, ferman_toro001@windowslive.com, ibrahim_erkin@gmail.com, eorhan_akan200@msn.com, cwilson@windowslive.com, nderman@mail.com, px4storm@mail.com, expedia004@mynet.com, odurukan002@yahoo.com, hjersey002@windowslive.com, uyelkenci@gmail.com, lcandan88300@msn.com, jaleata33@myspace.com, agnes76113@mail.com, uwonder@msn.com, ybirsen_ilter010@myspace.com, fulya_yapici67401@mail.com, asli_87010@gmail.com, ghostwisperer99410@mynet.com, hreha_samanci@mail.com, imrankahraman004@mail.com, ukora200@msn.com, zsedacetinkaya001@mynet.com, mxedos100@hotmail.com, gdidemguzel@gmail.com, greg_colt001@mail.com, htopal_sinan@windowslive.com, qwendy001@mynet.com, nnina_dorhan001@mail.com, planeta@mynet.com, cjake_carmen100@gmail.com, pjak_ad@mynet.com, orhan_tekin@msn.com, evindar_aydin@hotmail.com, vnazar@mail.com, obaykan@msn.com, tdoganay@msn.com, zquit@mynet.com, ulgerdoganay@msn.com, riza_alan@mynet.com, uparla@gmail.com, rramazan91@hotmail.com, unaleksi@myspace.co, greg_colt@mynet.com, znida_68@mail.com, ddundar_cetin@myspace.com, jdharma@hotmail.com

>Dear collegues;

>You can find extra information for figes in the given link: http://www.figes.com.tr/argegunleri/kprogram.php

>Best regards.

Şekil 5.2. Örnek gizli mesaj kullanılarak Huffman kodlaması ile oluşturulan stego ortam (Forward mail platformu)



Şekil 5.3. Önerilen metodun sihirli üçgendeki yeri

Şekil 5.3'de önerilen metodun sihirli üçgendeki yeri gösterilmektedir. Hatırlanacağı üzere sihirli üçgende gösterilen bilgi gömme sistemlerindeki üç temel faktör arasında bir ödünleşim mevcuttur ancak hepsinin aynı anda sağlanması mümkün olmamaktadır. Bu çalışmada, yeni bir yaklaşım önerilerek algılanamazlık ve kapasite konuları ele alınmış, güvenlik konusu desteklenmiştir. Başka bir deyişle bu çalışmanın yeniliği ve katkısı, algılanamazlığı korurken, örten ortama saklanabilen veri miktarını artırmak ve güvenliği sağlamaktır. Algılanamazlık, stego ortamı forward mail platformu olarak düzenleyerek ve iki taraf arasındaki iletişim için bu ortamı kullanarak sağlanmıştır. Bilindiği üzere, bir forward e-postanın görünen kısmı, e-posta adresleri ve ilgili metinden oluşmaktadır. Gizli bilgi, grup hitabında kullanılabilecek doğal olarak (dilbilgisi kurallarına uyularak) oluşturulan metinleri içeren metin tabanından seçilen bir metin içerisine saklanmıştır. Saklama işlemi, ne anlam ne de biçim değiştirilmeden, örten metnin orijinalliği korunarak gerçekleştirilmiştir. Örten metin içerisindeki gizli mesajın konum bilgisi ise, stego ortamda kullanılan e-posta adreslerine gömülmüştür. Böylece, stego ortam iki taraf arasındaki iletişim için mantıklı ve akla yatkın görünmektedir.

Güvenlik konusu, çıkarım aşamasının karmaşıklaştırılması için veri sıkıştırma teknikleri ve steganografik ortamı analiz etmeye çalışan bir gözlemci için istenen rastgeleselliği sağlamak amacıyla kombinatorik tabanlı kodlama kullanılarak sağlanmıştır. Burada en önemli nokta, metinsel veri ile çalışıldığı için bilgi kaybını önlemektir. Dolayısıyla, kullanılan veri sıkıştırma algoritmalarının kayıpsız olması gerekmektedir. Bu doğrultuda, literatürdeki yaygın kullanımları ve dikkate değer sıkıştırma oranları sebebiyle LZW ve Huffman kodlama algoritmaları kullanılmıştır. Simetrik şifreleme usulü paylaşılan stego anahtar kullanımı ile de güvenlik artırılmıştır.

Tablo 4.3.' te gösterildiği gibi Winstein veri tabanı kullanılarak gerçekleştirilen deneyler neticesinde ise 300 karakter (ya da 300·8 bit) içeren bir gizli mesaj için, LZW kodlaması kullanıldığında elde edilen kapasite değeri %8.37, Huffman kodlaması kullanıldığında elde edilen kapasite değeri %9.34 olmaktadır. Diğer taraftan, güvenlik analizi, önerilen metodun tüm adımlarının herkesçe bilindiği farz edilerek, gizli mesajın çıkarılması amacıyla gerçekleştirilmesi gereken kombinasyon sayısı formüle edilip hesaplanarak gerçekleştirilmiştir. Önerilen metot, literatürdeki diğer güncel metotlar ile en yaygın ölçüt olan kapasite, açısından karşılaştırılmıştır. Önerilen metot ile elde edilen kapasite değerleri literatürde tepelerde yer almaktadır. Ayrıca gizli mesajın uzunluğu arttıkça elde edilen kapasite değerlerinin de arttığı görülmüştür. Bu yolla uzunluk artışının kapasite üzerine olan dezavantajı da avantaja çevrilmiştir.

5.2. Öneriler

Bu bölümde önerilen metodun avantaj ve dezavantajları açıklanmıştır. Önerilen metodun bir avantajı, dile özgü olmamasıdır. Önerilen metotta işlemler sayılar ile gerçekleştirilmektedir. Gizlenecek mesajın diline bağlı kalınmaksızın gizleme işlemi sonucu sayısal bir dizi elde edilmektedir. Bu aşamadan sonraki tüm işlemler sayılar ile gerçekleştirildiğinden ötürü önerilen metotta dile bağımlılık asgari düzeydedir. Eğer saklanacak mesaj Türkçe ise, metin tabanı da Türkçe olmalıdır, İngilizce ise İngilizce olmalıdır, Çince ise Çince olmalıdır vb. Aksi halde örten metin ve gizlenecek mesaj arasında harf eşlemesi bulunamayacak ve başlangıçta gerekli olan sayısal dizi oluşturulamayacaktır.

Diğer bir avantaj ise, iletişim esnasında örten ortamın orijinalliğinin korunmasıdır. Önerilen metot, gizli bilginin korunması amacıyla gürültü üretmemektedir. Ayrıca yine bu amaçla metnin anlamı ve formatı değiştirilmemektedir. Önerilen metotta, stego ortam; iki örten ortamdan oluşan bir forward mail platformudur. Bunlardan biri dilbilgisi kurallarına uyularak önceden oluşturulmuş bir metindir. Bu durum, metni mantıklı ve anlamlı yapmaktadır. İkinci örten ortam ise mail platformunu forward mail olarak göstermek amacıyla kullanılan e-posta adresleridir. Burada başlıca amaç şüphe uyandırmamaktır çünkü e-posta adreslerinin oluşturulması için belli bir format ya da kural yoktur. Sayılar, farklı ya da aynı rakamlar veya karakterler ardı ardına kullanılabilir ve e-posta adreslerinin anlamlı olmaları gerekmez. Buradaki tek

kısıt, e-posta servis sağlayıcılarının dünya genelinde ISO (International Organization for Standardization) standartlarına göre temel Latin alfabesini desteklemesidir. Yani dünya genelinde, bir e-posta adresinde bu alfabe dışındaki harfler kullanılamamaktadır.

Bu özelliklerinden ötürü önerilen metot OCR programları ve yeniden yazım karşısında dayanıklıdır. Ayrıca önerilen metodun güvenliği kullanılan stego anahtarlar, simetrik şifreleme ve LZW ve Huffman kodlaması ile güçlendirilmektedir. Ancak, çıkarım aşamasını daha da karmaşıklaştırarak önerilen metodun güvenliğini daha da artırmak amacıyla, kriptolojik teknikler kullanılabilir. Bu durumun kapasiteye ek yük getirebileceği göz önünde bulundurulmalıdır. Bunu önlemek için ise diğer kayıpsız sıkıştırma algoritmalarının etkisi incelenebilir. Böylelikle hem önerilen metot ile saklanabilen veri miktarı artırılabilirken hem de karmaşıklık, sayılması ve hesaplanması güç bir miktara ulaşabilecektir.

KAYNAKLAR

Aabed, M.A., Awaideh, S.M., Abdul-Rahman, M.E., Gutub, A., 2007, Arabic diacritics based Steganography, *IEEE International Conference on Signal Processing andCommunications (ICSPC 2007)*, Dubai, UAE, 756-759.

Alla, K., Prasad, R. S. R., 2009, An Evolution of Hindi Text Steganography, S*ixth International Conference on Information Technology: New Generations, ITNG 2009*, Las Vegas, Nevada,1577 – 1578.

Al-Bahadili, H., 2008, A novel lossless data compression scheme based on the error correcting Hamming codes, *Computers & Mathematics with Applications*, 56(1), 143-150.

Al-Nazer, A., Gutub, A., 2009, Exploit Kashida Adding to Arabic e-Text for High Capacity Steganography, *International Workshop on Frontiers of Information Assurance & Security (FIAS 2009) - IEEE 3rd International Conference on Network & System Security (NSS 2009)*, Gold Coast, Queensland, Australia, 447-451.

Bailey, K., Curran, K., 2006, An evaluation of image based steganography methods using visual inspection and automated detection techniques, *Multimed Tools Appl.*, 30(1), 55-58.

Begum, M. B., Venkataramani, Y., 2012, LSB Based Audio Steganography Based On Text Compression, *Procedia Engineering*, 30, 703-710.

Chang, C., Kieu, T.D., 2010, A reversible data hiding scheme using complementary embedding strategy, *Information Sciences*, 180 (16), 3045–3058.

Chapman, M., Davida, G.I., 1997, Hiding the hidden: a software system for concealing cipher text as innocuous text, *International Conference on Information and Communications Security*, Lecture Notes in Computer Science, Springer, Beijing, 1334, 335–345.

Chapman, M., Davida, G.I., 2001, A practical and effective approach to largescale automated linguistic steganography, *Information Security Conference (ISC '01)*, Lecture Notes in Computer Science, Springer, Malaga, 2200, 156 - 165.

Chapman, M., Davida, G.I., 2002, Plausible deniability using automated linguistic steganography, *Davida, G., Frankel, Y. (eds.) International Conference on Infrastructure Security (InfraSec '02)*, Lecture Notes in Computer Science, Springer, Berlin, 2437, 276–287.

Desoky, A., 2009, Listega: list-based steganography methodology, *International Journal of Information Security*, 8(4), 247-261.

Easton, T., Gary Parker, R., 2001, On completing latin squares, *Discrete Applied Mathematics*, 113 (2-3), 167-181

Elci, B., Ors, S. B., Dalmisli V., 2008, Bir Steganografi Sisteminin FPGA Üzerinde Gerceklenmesi, *3. Uluslararasi Katilimli Bilgi Guvenligi ve Kriptoloji Konferansi*, Ankara

Galambos, G., Bekesi, J., 2002, Data Compression: Theory and Techniques, Department of Informatics, Teacher's Training College, Database and Data Communication Network Systems, Vol. 1, *Elsevier Science*, USA, Copyright 2002.

Gutub, A., Fattani, M., 2007, A novel Arabic text steganography method using letter points and extensions, *WASET International Conference on Computer, Information and Systems Science and Engineering (ICCISSE)*, Vienna, Austria, 28–31.

Jun, L., Tong, W., Daxin, L., 2011, Research on Ordinal Properties in Combinatorics Coding Method, *Journal of Computers*, 6(1), 51-58.

Khairullah, M., 2009.A Novel Text Steganography System Using Font Color of the Invisible Characters in Microsoft Word Documents.*Proceedings of the 2009 Second International Conference on Computer and Electrical Engineering*, Dubai, 482-484.

Kärkkäinen, J., Navarro, G., Ukkonen, E., 2003, Approximate string matching on Ziv-Lempel compressed text, *J. Discrete Algorithms*, 1(3-4), 313-338.

Lee, I.S., Tsai, W.H., 2010, A new approach to covert communication via PDF files, *Signal Process*, 90(2), 557–565

Liang, J. Y., Chen, C.S., Huang, C.H., Liu, L., 2008, Lossless compression of medical images using Hilbert space-filling curves, *Comp. Med. Imag. and Graph.*, 32(3), 174-182.

Lou, D.,Wu, N., Wang, C., Lin, Z., Tsai, C.S., 2010, A novel adaptive steganography based on local complexity and human vision sensitivity, *J Syst Softw*, 83(7) , 1236-1248.

Maher, K., TEXTO, 1995. ftp://ftp.funet.fi/pub/crypt/steganography/texto.tar.gz, [Ziyaret Tarihi: 23 Temmuz 2012]

Murphy, B., Vogel, C., 2007, The syntax of concealment: reliable methods for plain text information hiding., *Proceedings of the SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents*

Nakagawa, H., Sampei, K., Matsumoto, T., Kawaguchi, S., Makino, K., Murase, I., 2001, Text information hiding with preserved meaning—a case for Japanese documents, *IPSJ Trans*. 42 (9), 2339–2350. (İngilizcesi için: http://www.r.dl.itc.u-tokyo.ac.jp/nakagawa/academic-res/finpri02.pdf. Ziyaret Tarihi: 4 Hairan 2008)

Park, J., Lee, S., 2009, Forensic investigation of Microsoft PowerPoint files, *Digital Investigation*, 6 (1–2), 16–24.

Pfitzmann, B., 1996, Information Hiding Terminology, in Information Hiding, Springer Lecture Notes in Computer Science, *New York*,1174, 347-350.

Por, L.Y., Wong, K., Chee, K.O., 2012, UniSpaCh: a textbased data hiding method using unicode space characters, *J Syst Softw*, 85(5), 1075-1082.

Rafat, K. F., 2009, Enhanced text steganography in SMS, *Proc. of the 2nd Int. Conf. Computer, Control and Communication*, Karachi, 1-6.

Rafat, K. F., Sher, M., 2010, Survey Report – State Of The Art In Digital Steganography Focusing ASCII Text Documents, *International Journal of Computer Science and Information Security (IJCSIS)*, 7 (2), 63-72.

Ryabko, B., Ryabko, D., 2011, Constructing perfect steganographic systems, *Inf Comput*, 209(9),1223–1230

Sahin, A.,Bulus, E., Sakallı, M.T., 2006, 24-Bit Renkli Resimler Üzerında En Önemsiz Bite Ekleme Yöntemini Kullanarak Bilgi Gizleme, *Trakya Universitesi Fen Bilimleri Dergisi*, Edirne-TURKIYE

Salomon, D., 2005, Coding for Data and Computer Communications, Springer Science + Business Media Inc., *USA*.

Samphaiboon, N., 2009, Steganography via running short text messages, *Multimed Tool Appl*, 52(2–3), 569–596

Satir, E, Isik, H. A., 2012, Compression - Based Text Steganography Method, *J Syst Softw*, 85 (10), 2385-2394

Satir, E, Isik, H. A, 2012, Huffman Compression - Based Text Steganography Method,*Multimed Tools Appl.*, DOI 10.1007/s11042-012-1223-9

Shih, F. Y., 2005, Digital Watermarking and Steganography Fundamentals and Techniques, *CRC Press*, London

Shirali-Shahreza, M.H., Shirali-Shahreza, M., 2006.A New Approach to Persian/Arabic Text Steganography.Proceedings of the 5th IEEE/ACIS International Conferenceon Computer and Information Science (ICIS 2006), Honolulu, HI, USA, 10-12 July, pp. 310-315.

Shirali-Shahreza, M., 2008, Text Steganography by Changing Words Spelling, *Proceedings of the 10th International Conference on Advanced Communication Technology (ICACT 2008)*, Phoenix Park, Korea, 1912-1913.

Shu, Y., Liu, L., Tian, W., Miao, X., 2011, Algorithm for Information Hiding in Optional multi-Text, *Procedia Engineering*,15, 3936-3941.

Stutsman, R., Atallah, M., Grothoff, C., Grothoff, K., 2006. Lost in just the translation, *2006 ACM symposium on applied computing*, Dijon, France, 338–345

Sun, X.M., Luo, G., Huang, H.J., 2004. Component-based digital watermarking of Chinese texts, *3rd International Conference on Information Security*, Shanghai, China, 76–81

Topkara, M., Topkara, U., Atallah, M.J., 2007. Information hiding through errors: a confusing approach, *SPIE International Conference on Security, Steganography, and Watermarking of MultimediaContents*, San Jose, CA, USA, 29 Ocak -1 Şubat

Wang, Z., Chang, C., Lin, C., Li, M., 2009a, A reversible information hiding scheme using left-right and updown Chinese character representation, *J Syst Softw* 82:1362–1369

Wang, Z.H., Kieu, T.D., Chang, C.C., Li, M.C., 2009b. Emoticon-based text steganogra-phy in chat. *Proceedings of 2009 Asia-Pacific Conference on Computational Intelligence and Industrial Applications (PACIIA 2009),* Wuhan, China, 2, 457–460.

Wayner, P., 1992, Mimic functions, *Cryptologia*, XVI(3), 193–214, doi:10.1080/0161-119291866883

Wayner, P., 2002, Disappearing Cryptography, 2nd ed. Morgan Kaufmann, *Menlo Park*, 81–128.

Winstein, K., 1999, Lexical steganography through adaptive modulation of the word choice hash, Secondary education at the IllinoisMathematics and Science Academy, January, http://alumni.imsa.edu/~keithw/tlex/lsteg.ps. [Ziyaret Tarihi: 15 Nisan 2008]

Winstein. K., Lexical steganography, http://alumni.imsa.edu/~keithw/tlex, [Ziyaret Tarihi: Accessed 24 Mayıs 2012]

Yeh, W. H.,Hwang, J.J., 2001, Hiding Digital Information Using a Novel System Scheme. *Computers & Security*, 20(6), 533-538.

Zaker, N., Hamzeh, A., 2011, A novel steganalysis for TPVD steganographic method based on differences of pixel difference histogram, *Multimed Tool Appl*. doi:10.1007/s11042-010-0714-9

EKLER

EK-1 Düzenlenen Latin karesi

Row	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
1	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
2	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
3	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
4	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
5	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
6	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
7	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
8	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
9	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
10	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
11	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
12	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
13	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
14	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
15	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
16	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
17	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
18	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
19	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
20	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
21	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
22	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
23	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
24	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
25	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
26	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

EK-2 Gömme aşaması için sözde kodlar-LZW kodlaması

Get S
Get T
For each $Text$ in T
    Calculate $\overrightarrow{\Delta D}$
    Generate a line of D by getting $\overrightarrow{\Delta D}$
End for
For each $\overrightarrow{\Delta D}$ in D
    For each c in $\overrightarrow{\Delta D}$
        If $c > 26$ then
            $e = c \backslash 26$
            $r = c \bmod 26$
        else
            $e = 0$
            $r = c$
        End if
        Generate $\vec{E}$ by getting e
        Generate $\vec{R}$ by getting r
    End for
    Generate a line of E by getting $\vec{E}$
    Generate a line of R by getting $\vec{R}$
End for
For each $\vec{R}$ in R
    Calculate p
    Generate $\vec{P}$ by getting p
End for
Find maximum p and its index in $\vec{P}$
Get the line of R as $\vec{R}$ which corresponds to the index of maximum p
Get the line of E as $\vec{E}$ which corresponds to the index of maximum p
Get the line of T as T^* which corresponds to the index of maximum p
Generate $\overrightarrow{R'}$ via *LZW coding*
Bit Stream $= (\overrightarrow{R'})_2$
Generate *12 bit groups* by seperating *Bit Stream*
For each *12 bit groups* in *Bit Stream*
    Get G_1
    $x = [(G_1)_{10}] \backslash 26$
    $y = (G_1)_{10} mod\ 26$
    Generate letter equivalents for x and y via *Latin Square*
    Generate an element of K_2 by getting the corresponding e-mail address from K_1
    Get G_2
    $z = (G_2)_{10}$
    Generate a stego key in K_2 by getting email address extension from A according to the value of z
End for
$i = 0$
For each 3 elements of $\vec{E}$
    Generate a stego key by adding 3 elements to $K_2[i]$ before "@"
    $i = i+1$
End for
Generate *Stego Cover* by getting T^* and K_2

EK-3 Gömme aşaması için sözde kodlar-Huffman kodlaması

Get S
Get T
For each $Text$ in T
    Calculate $\overrightarrow{\Delta D}$
    Generate a line of D by getting $\overrightarrow{\Delta D}$
End for
For each $\overrightarrow{\Delta D}$ in D
    For each c in $\overrightarrow{\Delta D}$
        If $c > 26$ then
            $e = c \backslash 26$
            $r = c \bmod 26$
        else
            $e = 0$
            $r = c$
        Endi f
        Generate $\vec{E}$ by getting e
        Generate $\vec{R}$ by getting r
    End for
    Generate a line of E by getting $\vec{E}$
    Generate a line of R by getting $\vec{R}$
End for
For each $\vec{R}$ in R
    Calculate p
    Generate $\vec{P}$ by getting p
End for
Find maximum p and its index in $\vec{P}$
Get the line of R as $\vec{R}$ which corresponds to the index of maximum p
Get the line of E as $\vec{E}$ which corresponds to the index of maximum p
Get the line of T as T^* which corresponds to the index of maximum p
Calculate frequency *(f)* of the symbols of Latin Square according to $\vec{R}$
Generate $\vec{F}$ by getting f values
For each f in $\vec{F}$ step 2
    Generate letter equivalents of $\vec{F}$ via *Latin Square*
    Generate an element of K_2 by getting the corresponding e-mail address from K_1
End for
Generate $\overrightarrow{R'}$ via *Huffman coding*
Generate *12 bit groups* by seperating $\overrightarrow{R'}$
For each *12 bit groups* in $\overrightarrow{R'}$
    Get G_1
    $x = [(G_1)_{10}] \backslash 26$
    $y = (G_1)_{10} mod\ 26$
    Generate letter equivalents for x and y via *Latin Square*
    Generate an element of K_2 by getting the corresponding e-mail address from K_1
    Get G_2
    $z = (G_2)_{10}$
    Generate a stego key in K_2 by getting e-mail address extension from A according to the value of z
End for
$i = 0$
For each element of $\vec{E}$ step 3
    Generate a stego key by adding 3 elements to $K_2[i]$ before "@"
    $i = i+1$
End for
Generate *Stego Cover* by getting T^* and K_2

EK-4 Çıkarım aşaması için sözde kodlar-LZW kodlaması

Get *Stego Cover*
For each k in K_2
    If $k =$ *j (element of* K_1) then
        e= 0
        Generate $\vec{E}$ by getting e
    Else
        $e = k - j$
        $e' = e \backslash 100$
        $e'' = [e - (e' \cdot 100)] \backslash 10$
        $e''' = e \; mod \; 10$
        Generate 3 elements of $\vec{E}$ by getting e', e'', and e'''
    End if
End for
For each k in K_2
    Find x and y via *Latin Square*
    Find z via A
    $G_1 = (x \cdot 26 + y)_2$
    $G_2 = (z)_2$
    Generate $(\vec{R}')_2$ by getting G_1 and G_2
End for
Generate $\vec{R}$ via *LZW coding*
i =0
For each r in $\vec{R}$
    $c = r + \left(26 \cdot \overrightarrow{E[\dot{1}]}\right)$
    i =i+1
    Generate $\overrightarrow{\Delta D}$ by getting c
End for
For each c in $\overrightarrow{\Delta D}$
    $a = T^*[c]$
    Generate S by getting a
End for

EK-5 Çıkarım aşaması için sözde kodlar-Huffman kodlaması

Get *Stego Cover*
For each k in K_2
    If $k = j$ *(element of* K_1) then
        $e = 0$
        Generate $\vec{E}$ by getting e
    Else
        $e = k - j$
        $e' = e \backslash 100$
        $e'' = [e - (e' \cdot 100)] \backslash 10$
        $e''' = e \; mod \; 10$
        Generate 3 elements of $\vec{E}$ by getting e', e'', and e'''
    End if
End for
$i = 0$
For i= 1 to 13
    Find f values via *Latin Square* by getting $K_2[i]$
    Generate $\vec{F}$ by getting f
End for
$i = 0$
For $i = 14$ to $s(K_2)$
    Find x and y via *Latin Square* by getting $K_2[i]$
    Find z via A
    $G_1 = (x \cdot 26 + y)_2$
    $G_2 = (z)_2$
    Generate $\overrightarrow{R'}$ by getting G_1 and G_2
End for
Generate $\vec{R}$ via *Huffman coding*
$i = 0$
For each r in $\vec{R}$
    $c = r + \left(26 \cdot \overrightarrow{E[\dot{1}]}\right)$
    $i = i + 1$
    Generate $\overrightarrow{\Delta D}$ by getting c
End for
For each c in $\overrightarrow{\Delta D}$
    $a = T^*[c]$
    Generate S by getting a
End for

EK-6 Örnek: LZW kodlaması kullanılarak stego ortamın oluşturulması

Gizlenecek mesaj: inan

S=(i, n, a, n)

Modül 1: Başlangıç işlemleri yapılır.

Modül 2 ve 3' e göre gizli mesaj dikkate alınarak seçilen örten metin ve bu örten metne göre oluşturulan ilgili diziler:

T:*

Dear trainers;

I want to inform you about the next session. We are going to study on information security. It is very important for you to join and follow. You have to organize about the hour. It is planning as two o'clock. Please contact me for extra questions.

Best regards

$\overrightarrow{\Delta D} = (9,1,9,1)$

$\vec{R} = (9,1,9,1)$

$\vec{E} = (0,0,0,0)$

Modül 4: LZW sıkıştırması gerçekleştirilmektedir. Oluşturulan LZW sözlüğü ve sıkıştırılmış dizi aşağıda verilmiştir.

LZW Sözlüğü							
Indeks	**İçerik**	**Indeks**	**İçerik**	**Indeks**	**İçerik**	**Indeks**	**İçerik**
1	1	8	8	15	15	22	22
2	2	9	9	16	16	23	23
3	3	10	10	17	17	24	24
4	4	11	11	18	18	25	25
5	5	12	12	19	19	26	26
6	6	13	13	20	20	**27**	**9,1**
7	7	14	14	21	21	**28**	**1,9**

$\overrightarrow{R'} = (9,1,27)$

Modül 5: Kodlama ile sayılardan e-posta adreslerine geçilmektedir.

Adım 1: 12'lik gruplar

R'= (9, 1, 27)

1001 1 11011

01001 00001 11011

010010000111011**000000000**

Adım 2: İkilik Tabandan Onluk Tabana

$\underbrace{010010000}_{G_1}\ \underbrace{111}_{G_2}$ $\underbrace{011000000}_{G_1}\ \underbrace{000}_{G_2}$

010010000	111	011000000	000
X_1=144\26=5	mynet	X_2= 192\26=7	hotmail
Y_1=144mod26=14		Y_2= 192mod26=10	
5+1=6, 14+1=15		7+1=8, 10+1=11	

Adım 3: Onluk Tabandan Harflere:

Latin karesinden alınan harfler: 6→F, 15→O, 8→i, 11→L

Listeden (K_1) seçilen e-posta adresleri:

folgun_ersoy@mynet.com

ilhankirmizi@hotmail.com

Row	0	1
1	A	B
2	B	C
3	C	D
4	D	E
5	E	F
6	F	G
7	G	H
8	H	I
9	I	J
10	J	K
11	K	L
12	L	M
13	M	N
14	N	O
15	O	P

Modül 6: Stego anahtar:

$\vec{E} = (0,0,0,0)$ olduğundan bu adımda global stego anahtar kullanımı için yalnızca e-posta adres uzantısı değiştirilmektedir.

Modül 7: Forward Mail Platformu:

Seçilen e-posta adresleri ve örten metin birleştirilerek forward mail platformu oluşturulur:

----Original Message----

From: sender@hotmail.com

Date:Wednesday, February 27, 2012 8: 38 am

Subject: Notification

To: recipient@hotmail.com

CC:folgun_ersoy@mynet.com, ilhankirmizi@hotmail.com

>Dear trainers;

>I want to inform you about the next session. We are going to study on information security. It is very >important for you to join and follow. You have to organize about the hour. It is planning as two o'clock.

>Please contact me for extra questions.

>Best regards

EK-7 Örnek: Huffman kodlaması kullanılarak stego ortamın oluşturulması

Gizlenecek mesaj: inan

S=(i, n, a, n)

Modül 1: Başlangıç işlemleri yapılır.

Modül 2 ve 3' e göre gizli mesaj dikkate alınarak seçilen örten metin ve bu örten metne göre oluşturulan ilgili diziler:

T:*

Dear trainers;

I want to inform you about the next session. We are going to study on information security. It is very important for you to join and follow. You have to organize about the hour. It is planning as two o'clock. Please contact me for extra questions.

Best regards

$\overrightarrow{\Delta D} = (9,1,9,1)$

$\vec{R} = (9,1,9,1)$

$\vec{E} = (0,0,0,0)$

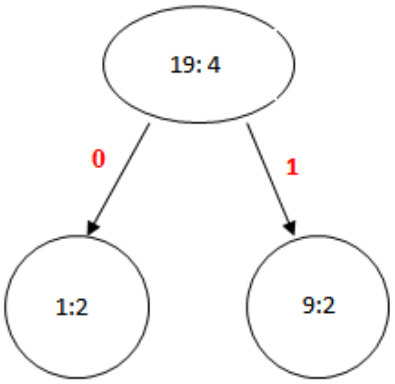
Modül 4: Huffman sıkıştırması gerçekleştirilmektedir.

Huffman Frekans Tablosu

Frekans Tablosu	
Sembol	**Frekans**
1	2
2	0
⋮	⋮
8	0
9	2
10	0
⋮	⋮
26	0

$\overrightarrow{R'} = (1,0,1,0)$

Huffman Ağacı



Modül 5: Kodlama ile sayılardan e-posta adreslerine geçilmektedir.

Adım 1: Öncelikle sıkıştırılmış dizinin çözülmesi için 26 sembolün frekans bilgilerinin göndericiye iletilmesi gerekmektedir. Örneğe göre:

Row	0	1	2
1	A	B	C
2	B	C	D
3	C	D	E
4	D	E	F
5	E	F	G
6	F	G	H

1: 2 defa, 2: 0 defa geçmektedir:

2+1=3, 0+1=1→ ca…@hotmail.com

3: 0 defa, 4: 0 defa germektedir:

0+1=1, 0+1=1→ bb…@hotmail.com

⋮

1-26 arası tüm sembollerin frekans bilgilerini alıcıya iletilmesi için aynı işlemler yapılarak toplamda 13 adet e-posta adresi oluşturulmaktadır.

Adım 2: 12’ lik bit grupları:

1010**00000000**

Adım 2: İkilik Tabandan Onluk Tabana

$\underbrace{101000000}_{G_1}\ \underbrace{000}_{G_2}$

101000000 000

X_1=320\26=12 hotmail

Y_1=320mod26=8

12+1=13, 8+1=9

Adım 3: Onluk Tabandan Harflere:

Latin karesinden alınan harfler: 13→M, 9→i

Listeden seçilen Email adresi:

miray_kara@hotmail.com

Row	0
1	A
2	B
3	C
4	D
5	E
6	F
7	G
8	H
9	I
10	J
11	K
12	L
13	M

Modül 6: Stego anahtar:

$\vec{E} = (0,0,0,0)$ olduğundan bu adımda global stego anahtar kullanımı için yalnızca e-posta adres uzantısı değiştirilmektedir.

Modül 7: Forward Mail Platformu:

Seçilen e-posta adresleri ve örten metin birleştirilerek

forward mail platformu oluşturulur:

----Original Message----

From: sender@hotmail.com

Date: Wednesday, February 27, 2012 8: 38 am

Subject: Notification

To: recipient@hotmail.com

CC: canan_kuru@hotmail.com, bbaykal@hotmail.com, ccansel_ugur@hotmail.com, ddundar_cetin@hotmail.com, georgina_milano@hotmail.com, ffred_dawson@hotmail.com, ggencer@hotmail.com, hharmanci_85@hotmail.com, iilknurkara@hotmail.com, jjanuary@hotmail.com, kkuyumcu@hotmail.com, llindsay@hotmail.com, mmoriss@hotmail.com, miray_kara@hotmail.com

>Dear trainers;

>I want to inform you about the next session. We are going to study on information security. It is very >important for you to join and follow. You have to organize about the hour. It is planning as two o'clock.

>Please contact me for extra questions.

>Best regards

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı	**:**	Esra ŞATIR
Uyruğu	**:**	T.C.
Doğum Yeri ve Tarihi	**:**	KONYA - 26.04.1983
Telefon	**:**	0332 223 33 31
Faks	**:**	0332 241 21 79
e-mail	**:**	esatir@selcuk.edu.tr

EĞİTİM

Derece		**Adı, İlçe, İl**	**Bitirme Yılı**
Lise	:	Selçuklu Anadolu Lisesi, Selçuklu, KONYA	2001
Üniversite	:	Gazi Üniversitesi, ANKARA	2005
Yüksek Lisans	:	Selçuk Üniversitesi, Selçuklu, KONYA	2009
Doktora	:	Selçuk Üniversitesi, Selçuklu, KONYA	2013

İŞ DENEYİMLERİ

Yıl	**Kurum**	**Görevi**
2005-2006	M.E.B. Boyabat Ticaret Meslek Lisesi	Teknik Öğretmen
2006-…	T.C. Selçuk Üniversitesi	Araştırma Görevlisi

UZMANLIK ALANI

Yapay Zeka ve Optimizasyon Teknikleri

Bulanık Mantık
Yapay Sinir Ağları
Adaptive Neuro Fuzzy Inference System (ANFIS)
Genetik Algoritmalar
Yapay Bağışıklık

Bilgi Güvenliği

Steganografi

Kayıpsız Veri Sıkıştırma

LZW Kodlaması
Huffman Kodlaması

YABANCI DİLLER

İngilizce

BELİRTMEK İSTEĞİNİZ DİĞER ÖZELLİKLER

İNOVASYON SÜRECİ VE AR-GE DESTEKLERİ EĞİTİM SERTİFİKASI
GENEL GİRİŞİMCİLİK EĞİTİM SERTİFİKASI
İLETİŞİM VE BEDEN DİLİ EĞİTİM SERTİFİKASI
FİKRİ VE SINAİ MÜLKİYET HAKLARI KAPSAMINDA MARKA VE TASARIM EĞİTİM SERTİFİKASI
FİKRİ VE SINAİ MÜLKİYET HAKLARI KAPSAMINDA PATENT VE FAYDALI MODEL EĞİTİM SERTİFİKASI

YAYINLAR

SCI İndeksli Yayınlar:

1. Isık H., **Saracoglu E.**, 2007. The Design of Thermoelectric Footwear Heating System via Fuzzy Logic, *Journal of Medical Systems*, 31(6): 521 - 527

2. Isık H., **Saracoglu E.**, Guler I., 2008. Design of Fuzzy Logic Controlled Thermoelectric Renal Hypothermia System. *Instrumentation Science & Technology*, 36(03): 310 – 322.

3. Isık H., **Saracoglu E.**, Harmanci H., Guler I., 2010. Design of a Cervical Collar to Facilitate and Implementation of First Aid. *Journal of medical Systems*, 34: 573-578

4. Isık H., Sezer E., **Saracoglu E.**, 2012. Employment and Comparison of Different Artificial Neural Networks for Epilepsy Diagnosis from EEG Signals. *Journal of Medical Systems*, 36: 347–362

5. **Satir E.**, Isik H. A., 2012. A Compression - Based Text Steganography Method. *Journal Of systems and Software*, 85(10): 2385-2394

6. **Satir E**., Isik H. A, 2012. A Huffman Compression - Based Text Steganography Method. *Multimed Tools Appl*, DOI 10.1007/s11042-012-1223-9

Uluslararası Sempozyum - Konferans

1. Isik H., **Saraçoglu E.**, Sezgin E., Caglayan N., A case of study on the calculate of optimum window ventilation opening in greenhouses by using of decision tree method. *International Conference for Academic Disciplines*, June 24, 2011, Prague

Diğer Uluslar Arası Dergiler

1. Isık H., **Saracoglu E.**, 2010, Comparison of Proportional Control and Fuzzy Logic Control to Develop an Ideal Thermoelectric Renal Hypothermia System. *World Academy of Science, Engineering and Technology*, 68(44): 1059-1066

2. **Satir E**., Isik H., 2012, TENS Modelling via ANFIS. *International Journal of Future Computer and Communication*, 1(2): 173-175

Ulusal Sempozyum - Konferans

1. Isık H., **Saracoglu E.**, İnsandaki Psikolojik ve Fizyolojik Parametreler doğrultusunda ANFIS ile TENS Modellemesi, *Bilimde Modern Yöntemler Sempozyumu (BMYS) -2008*, Eskişehir