

**STEGANOĞRAFİK YAKLAŞIMLARIN İNCELENMESİ, TASARIMI
VE GELİŞTİRİLMESİ**

Mehmet Ali ATICI

**YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ**

**GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**NİSAN 2007
ANKARA**

Mehmet Ali ATICI tarafından hazırlanan STEGANOĞRAFİK YAKLAŞIMLARIN İNCELENMESİ, TASARIMI VE GELİŞTİRİLMESİ adlı bu tezin yüksek lisans tezi olarak uygun olduğunu onaylarım.

Doç. Dr. Şeref SAĞIROĞLU

Tez Yöneticisi

Bu çalışma, Jürimiz tarafından oy birliğiyle Bilgisayar Mühendisliği Anabilim Dalında Yüksek Lisans tezi olarak kabul edilmiştir.

Başkan : Prof. Dr. Sezai DİNÇER

Üye : Doç. Dr. Şeref SAĞIROĞLU

Üye : Doç. Dr. Mehmet Ali AKCAYOL

Tarih : 17/04/2007

Bu tez, Gazi Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına uygundur.

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada orijinal olmayan her türlü kaynağa eksiksiz atıf yapıldığını bildiririm.

Mehmet Ali ATICI

STEGANOĞRAFİK YAKLAŞIMLARIN İNCELENMESİ, TASARIMI VE GELİŞTİRİLMESİ

(Yüksek Lisans Tezi)

Mehmet Ali ATICI

**GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

Nisan 2007

ÖZET

Bu çalışmada steganografik yaklaşımlar incelenmiş, ses dosyaları içerisine ses, resim, düz metin saklamak için bir yazılım geliştirilmiştir. Steganografik yazılımların günlük hayatımızda kişisel bilgi güvenliğinin sağlanmasında kullanılması için ise ayrıca bir klasör gizleme yazılımı da geliştirilmiştir. Steganografik yazılımların steganalize karşı sağlamlığını arttırmak için yazılımlar üzerinde analiz çalışmaları yapılmıştır. Haberleşme güvenliğinin sağlanmasının yanında bilgi güvenliğinin de sağlanması için AES, DES, 3DES gibi şifreleme seçenekleri de geliştirilen yazılım içerisinde sunulmuştur. Geliştirilen yazılımda bir anahtar yardımıyla belirlenen WAV dosyası seti içerisine LSB modifikasyonu yöntemiyle rastgele saklama gerçekleştirmektedir. Belirli bir anahtara göre saklama işlemi ile güvenliği algoritmanın bilinmesinden bağımsız hale getirmiş, rastgele saklama yapmasıyla ise istatistiksel steganaliz ataklarına karşı dayanıklılık sağlamıştır. Birden fazla WAV dosyası içerisine saklama yapabilmesiyle de rastgele saklamanın getirdiği saklama kapasitesi sorununa dolaylı bir çözüm sunmuştur. Bu tez çalışmasında geliştirilen yazılım ile hem iletişim hem de veri gizliliği sağlanabilmiştir.

Bu alışmanın, lkemizde bilgi gvenliĐinin yksek seviyede saĐlanmasına, bu konuda yapılacak yeni alışmalara ve gvenlik bilincinin yaygınlaştırılmasına byk katkılar saĐlayacaĐı deĐerlendirilmektedir.

Bilim Kodu : 902.1.011
Anahtar Kelimeler : Ses, Steganografi, Veri Saklama, Veri GvenliĐi, LSB, AES, DES, 3DES.
Sayfa Adedi : 109
Tez Yneticisi : Do.Dr. řeref SAĐIROĐLU

INVESTIGATION, DESIGN AND DEVELOPMENT OF STEGANOGRAPHIC TOOLS

(M.Sc. Thesis)

Mehmet Ali ATICI

**GAZİ UNIVERSITY
INSTITUTE OF SCIENCE AND TECHNOLOGY**

April 2007

ABSTRACT

In this work, steganografik tools were investigated and a software was developed for hiding audio, image and text files into audio files. In addition, a folder lock software was also developed for using steganographic tools to provide personel information security in our daily life. Analysis was made on the softwares to increase the robustness of steganographic tools against steganalysis. Encryption options like AES, DES and 3DES were also introduced in the software for providing information security as well as providing communication security. In the developed software, the process of random data hiding into the wav files set by LSB modification is occured with the help of a key. The tool makes the security independent from the algorithm by hiding according to a key and provides robustness against steganalysis attacks by random hiding. By hiding data into the wav files set instead of only one file, it provides indirect solution to the hiding capacity that results from random data hiding. Both communication and data security wav provided by the software that was developed in this thesis work.

It is examined that, this work will introduce important contributions to provision of high information security in our country, to the future works related to this topic and to increasing the security consciousness.

Science Code : 902.1.011

Key Words : Audio, Steganography, Data Hiding, Security, LSB, DES, AES, 3DES.

Page Number: 109

Supervisor : Assoc. Prof. Dr. Şeref SAĞIROĞLU.

TEŞEKKÜR

Bu tez çalışması boyunca her türlü konuda yardımlarını esirgemeyen değerli Hocam ve Danışmanım Doç. Dr. Şeref SAĞIROĞLU'na teşekkürlerimi sunarım.

İÇİNDEKİLER

Sayfa

ÖZET.....	iv
ABSTRACT.....	vi
TEŞEKKÜR.....	viii
İÇİNDEKİLER.....	ix
ÇİZELGELERİN LİSTESİ.....	xiii
ŞEKİLLERİN LİSTESİ.....	xiv
SİMGELER VE KISALTMALAR.....	xvi
1. GİRİŞ.....	1
2. STEGANOGRAFI.....	5
2.1. Steganografi.....	5
2.2. Resim İçerisine Veri Saklama.....	11
2.2.1. LSB modifikasyonu ve bitmap içerisine veri saklama.....	11
2.2.2. Dönüştürme tekniği ve jpeg resimlerine veri saklama.....	14
2.2.3. Maskeleye ve filtreleme.....	15
2.3. Ses İçine Veri Saklama.....	15
2.3.1. WAV dosyaları içerisine veri saklama.....	16
2.3.2. Mp3 içerisine veri saklama.....	17
2.3.3. MIDI dosyaları içerisine veri saklama.....	17
2.4. Metin İçine Veri Saklama.....	17
2.5. HTML İçerisine Veri Saklama.....	19
2.6. XML İçine Veri Saklama.....	22
2.7. Çalıştırılabilir Dosyalar İçerisine Veri Saklama.....	23

Sayfa

2.8. Steganografik Uygulamalar Hakkında Genel Değerlendirmeler.....	23
2.9. Steganografik Yazılımlar.....	24
2.9.1. OutGuess 0.2.....	24
2.9.2. Snow.....	25
2.9.3. Puff.....	25
2.9.4. GifShuffle.....	25
2.9.5. Deogol.....	25
2.9.6. Publimark 0.1.2.....	25
2.9.7. JPHS.....	26
2.9.8. MP3Stego.....	26
2.9.9. Steghide 0.5.1.....	26
2.9.10. Hydan.....	26
2.9.11. wbStego4open.....	26
2.9.12. SecureEngine professional 1.0.....	27
2.9.13. Hermetic stego.....	28
2.10. Türkiye’de Yapılan Steganografik Çalışmalar.....	29
2.11. Steganografi Temelli Kişisel Güvenlik Uygulamaları.....	31
3. STEGANALİZ.....	33
3.1. Resim Steganalizi.....	34
3.2. Ses Steganalizi.....	36
3.3. Ses Yazılımları.....	37
3.3.1. StegSpy.....	37

	Sayfa
3.3.2. Stego suite.....	37
3.3.3. Stegdetect.....	38
3.4. Steganaliz Konusunda Değerlendirmeler.....	38
4. ŞİFRELEME VE ÖZETLEME ALGORİTMALARI.....	39
4.1. Simetrik Şifreleme Algoritmaları.....	39
4.1.1. DES.....	39
4.1.2. 3DES.....	40
4.1.3. AES.....	41
4.2. Özetleme Algoritmaları.....	41
4.3. Steganografi & Kriptografi.....	42
5. SES İÇİNE VERİ SAKLAMA UYGULAMASI – DIS.....	43
5.1. PCM.....	43
5.2. WAV Dosya Yapısı.....	44
5.3. Ses Uygulaması için LSB Yöntemi.....	47
5.4. Programın Modülleri.....	47
5.5. Saklama İşlemi ve Detayları.....	49
5.5.1. Saklama kapasitesi ve saklama aralığı.....	53
5.5.2. XOR işlemi.....	55
5.5.3. Hata oranı.....	56
5.6. Değişik Seviyelerde Saklama İşleminin Analiz Edilmesi.....	56
5.7. Histogram Analizleri.....	57
5.8. Çıkartma İşlemi ve Detayları.....	61
5.9 Sonuç ve Değerlendirmeler.....	61

Sayfa

6. STEGO TABANLI KLASÖR KİLİTLEME YAZILIMI.....	63
6.1. Saklama İşlemi.....	64
6.1.1. Saklama kapasitesi.....	67
6.1.2. Oluşan hata.....	69
6.2. Çıkartma İşlemi.....	70
6.3. Yazılımın Özellikleri.....	73
6.4. STKK ve Diğer FL Yazılımları.....	75
6.5. STKK Yazılımının Modülleri.....	78
6.6. STKK Yazılımının Mevcut Steganografik Yazılımlarla Karşılaştırılması.....	83
6.7. Örnek Saklama İşlemi.....	85
7. SONUÇ VE ÖNERİLER.....	90
7.1. Sonuçlar.....	90
7.2. Öneriler.....	91
KAYNAKLAR.....	93
EKLER.....	98
EK-1 Stego ses dosyalarına ait histogram ve spectrogram grafikleri.....	99
ÖZGEÇMİŞ.....	109

ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 2.1. Örnek HTML etiket kodlaması.....	20
Çizelge 2.2. Çeşitli taşıyıcı dosya türleri.....	24
Çizelge 5.1. XOR doğruluk tablosu.....	55
Çizelge 6.1. STKK yazılımının mevcut yazılımlarla karşılaştırılması.....	84

ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 2.1. Resim dosyası yapısı.....	11
Şekil 2.2. Renk küpü.....	12
Şekil 2.3. Bir rengi oluşturan kırmızı, yeşil ve mavi tonların ait değerler.....	13
Şekil 2.4. Taşıyıcı HTML dosyası.....	21
Şekil 2.5. Stego HTML dosyası.....	22
Şekil 2.6. wbStego4 yazılımının ekran görüntüsü.....	27
Şekil 2.7. SecureEngine Professional 1.0 yazılımının ara yüzü.....	28
Şekil 2.8. Hermetic Stego demo versiyonu ara yüzü.....	28
Şekil 2.9. TürkSteg yazılımı ara yüz ekranı.....	29
Şekil 2.10. verigizle.com sitesinin işlem ekranı görüntüsü.....	30
Şekil 4.1. 3DES Şifreleme İşlemleri.....	40
Şekil 5.1. Sesin Sayısallaştırılması.....	43
Şekil 5.2. WAV dosya formatı.....	44
Şekil 5.3. Örnek WAV dosya başlığı.....	46
Şekil 5.4. WAV dosyası özellikler ekranı.....	46
Şekil 5.5. DIS yazılımı saklama işlemi akış diyagramı.....	50
Şekil 5.6. DIS yazılımı saklama ekranı.....	51
Şekil 5.7. Ses dosyalarına ait on altı değerler.....	52
Şekil 5.8. DIS yazılımı saklama işlemi.....	54
Şekil 5.9. DIS yazılımı örnek saklama ekranı.....	55

Şekil	Sayfa
Şekil 5.10. Taşıyıcı dosya ile 1.bit seviyesinde veri saklı dosyaya ait histogram çizelgeleri.....	58
Şekil 5.11. Taşıyıcı ve stego dosyalara ait histogram çizelgeleri.....	59
Şekil 5.12. Taşıyıcı ve stego dosyalara ait spectrogram grafikleri.....	60
Şekil 6.1. Geliştirilen STKK yazılımıyla gerçekleştirilen işlemlerin şemaları.....	64
Şekil 6.2. Saklama işlemi akış diyagramı.....	66
Şekil 6.3. SBFL yazılımı geri getirme işlemi akış diyagramı.....	72
Şekil 6.4. STKK Yazılımı veri saklama ekranı.....	77
Şekil 6.5. STKK Yazılımı veri geri getirme ekranı.....	78
Şekil 6.6. Saklama Yüzdesi & divider değişkeni ilişkisi.....	81
Şekil 6.7. Taşıyıcı ve stego ses dosyalarına ait spectrogram grafikleri.....	86

SİMGELER VE KISALTMALAR

Bu çalışmada kısaltmalar açıklamalarıyla birlikte aşağıda sunulmuştur.

Kısaltma	Açıklama
TÜİK	Türkiye İstatistik Kurumu
DES	Data Encryption Standard (Veri Şifreleme Standardı)
3DES	Triple Data Encryption Standard (Üçlü Veri Şifreleme Standardı)
AES	Advanced Encryption Standard (Gelişmiş Şifreleme Standardı)
SHA	Secure Hash Algorithm (Güvenli Özetleme Algoritması)
NIST	National Institute of Standards and Tehcnology (Ulusal Standartlar ve Teknoloji Enstitüsü)
SMS	Short Message Service (Kısa Mesaj Servisi)
BPCS	Bit Plane Complexity Segmentation (Bit Düzleminde Karmaşıklık Bölütlemesi)
LSB	Least Significant Bit (En az Önemli Bit)
RQP	Raw Quick Pairs (İşlenmemiş Hızlı Çiftler)
ANNTS	Artificial Neural Network Technology for Steganalysis (Steganaliz için Yapay Sinir Ağı Teknolojisi)
HTML	Hyper Text Markup Language (Üstün Metin İşaretleme Dili)
RGB	Red Green Blue (Kırmızı Yeşil Mavi)
AKT	Ayrık Kosinüs Transformasyonu
İİS	İnsan İşitme Sistemi
İGS	İnsan Görme Sistemi
XML	EXtensible Markup Language (Uzatılabilir İşaretleme Dili)

Kısaltma	Açıklama
DIS	Data Into Sound (Ses İçine Veri)
PCM	Pulse Code Modulation (Titreşim Kod Modülasyonu)
MIDI	Musical Instrument Digital Interface (Müzikal Enstrüman Sayısal Ara yüzü)
WAV	WAVEform ses biçimi
RIFF	Resource Interchange File Format (Kaynak Takas Dosya Biçimi)
ASCII	American Standard Code Information Interchange (Amerikan Kod Bilgi Takası Standardı)
XOR	eXclusive OR (Harici Mantıksal Veya İşlemi)
STKK	Stego Tabanlı Klasör Kilitleme (Yazılımı)
BMP	Bitmap (Nokta esaslı)
JPEG	Joint Photographic Experts Group (Birleşik Fotoğrafik Uzmanlar Grubu)
PNM	Portable Any Map (Taşınabilir Herhangi bir Harita)
GIF	Graphics Interchange Format (Grafik Takas Biçimi)
PCX	PC Paintbrush Graphic (Boya Fırçası Grafiği)
PNG	Portable Network Graphic (Taşınabilir Ağ Grafiği)
AIFF	Audio Interchange File Format (Ses Takas Dosya Biçimi)
PDF	Portable Document Format (Taşınabilir Doküman Biçimi)
DLL	Dynamic Link Library (Dinamik Bağlantı Kütüphanesi)
EXE	Executable (Çalıştırılabilir)
TB	Taşıyıcı Boyutu
KK	Klasör Kilitleme (Yazılımı)

Kısaltma**Açıklama****WAP**

Wireless Application Protocol (Kablosuz Uygulama Protokolü)

1.GİRİŞ

Bilişim teknolojilerinin hayatımıza daha fazla girmesi ve yaygınlaşmasıyla birlikte yapılan iş ve işlemler elektronik ortamlara kaymakta, bu ortamlarda bulunan, saklanan, işlenen ve transfer edilen bilgilerin ise korunması veya güvenliğinin sağlanması çok büyük önem arz etmektedir. Sayısal olarak veri iletişimi gerçekleştirilen bir ortamda, göndericiden alıcıya giden veriye yönelik izinsiz erişim, zarar verme, yok etme, değişiklik yapma ve yeniden üretme gibi birçok tehdit mevcuttur. Bu tehditlerin alınan önlemlere rağmen her geçen gün arttığı rapor edilmektedir [13,61]. Bu tehditlerin ortaya çıkmasına karşılık olarak bunların ortadan kaldırılması için çeşitli teknikler geliştirilmiştir. Şifreleme teknikleri bunların başında gelen çözüm yolları arasındadır.

Şifreleme veriyi kullanılamaz bir formata dönüştürüp verinin aslına ulaşılmasını zorlaştırır da yapılacak herhangi bir kriptanaliz atağını önleyemediği durumlar olabilmektedir. Ayrıca elektronik ortamlarda iki uç arasında şifreli bir iletişim yapıldığı anlaşıldığında ise iletişimi engelleme yoluna gidilebilmektedir. İşte bu noktada steganografi bilimi gündeme gelmektedir.

Steganografi kökleri binlerce yıl öncesine dayanan bir bilim dalıdır [12]. Kelime anlamı olarak gizli yazı veya örtülü yazı anlamına gelmektedir [25]. En temel amacı iletişimin gizliliğini sağlamaktır. Steganografide, sayısal bir verinin başka bir sayısal veri içerisine, fark edilebilir değişikliklere sebep olmadan saklanması gerçekleştirilmektedir. Örneğin şifrelenmiş bir metin, bir resim dosyasına saklanmakta ve sonuçta oluşan resim dosyası hem fiziksel olarak hem de görsel olarak orijinalinden farklı olmamaktadır. Böylece iki uç arasındaki iletişimi gözetleyenler, arada sadece transfer edilen bir resim görmekte, ama aslında bu resim yoluyla gizli bir mesajlaşma gerçekleştiğinin farkında olmamaktadırlar.

Sayısal veri dosyası formatlarının çeşitliliği sayesinde bir çok dosya türü içerisine steganografik yöntemlerle veri saklanabilmektedir [1,22,55,62,64,66,68]. Veri saklama yöntemlerinin temel mantığı, sayısal veri dosyası formatlarındaki gereksiz

veya çok önemli olmayan kısımların kullanılmasına veya insan duyularının istismar edilmesine dayanmaktadır.

İnsan görme sistemi (İGS), resim dosyalarındaki küçük değişimleri fark edememekte veya farkına varamamaktadır [67]. Örneğin bir Bitmap dosyasındaki piksel değerlerinin bir artırılması veya azaltılması sonucu oluşan renk değişimini göz fark edemez. Bu durumda resim dosyalarının bazı piksel değerleri, saklanacak verinin bitlerini barındırmak amacıyla değiştirilebilir ve bu değişiklikler insan gözü tarafından fark edilemez. Saklama işlemi sonucunda görsel açıdan ve boyut olarak orijinal dosya ile aynı bir resim dosyası oluşturulur.

İnsan işitme sistemi de (İİS), aynen görme sisteminde olduğu gibi sesteki çok küçük değişimleri fark edemez [26]. Mesela bir WAV dosyasında, ses örneği değerlerindeki küçük değişiklikler (artışlar veya azalışlar) kulak tarafından fark edilemez. Steganografi teknikleri işitme sisteminin bu özelliğinden faydalanarak, WAV dosyalarının ses örneği değerlerini, veri saklamak amacıyla değiştirebilirler.

Bazı steganografi yöntemleri ise veri dosyası formatındaki boşlukları veya gereksiz kısımları kullanma mantığına dayanmaktadır [30,66]. Mesela bir html dosyasını düşünelim: html etiketlerinin açılış ve kapanış kısımları vardır. Bir metne birden fazla etiket özelliği verilecekse, bunlardan hangisinin önce kapandığı görüntüyü değiştirmemektedir. Bu durum etiket kapama kısımlarının sıralanışının kodlanmasına ve böylece veri saklamaya imkan tanımaktadır. Çalıştırılabilir dosyalarda da durum böyledir. Bir çalıştırılabilir dosya komut setlerinden oluşmakta ve aynı işlevi gören birden fazla komutlar bulunmaktadır. Bu komutlardan hangisinin kullanılacağı kodlanarak çalıştırılabilir dosyalar içerisine de veri saklamak mümkün olabilmektedir. Aynı zamanda bir dokümandaki kelimeler arasında bir boşluk bulunması veya iki boşluk bulunması durumu kodlanarak doküman içerisine veri saklanabilmektedir [10].

Ses içerisine, ses dosyasının sıkıştırılarak başka bir formata dönüştürülmesi esnasında ses örneklerinin en az önemli bitlerinin modifikasyonu ile veya tekrarlanan

komutların varlığını kullanarak veri saklanabilmektedir [1,22,29]. WAV dosyaları en çok kullanılan ses dosyası türlerinden birisidir [10]. Genellikle 16 bitlik ses örneklerinden oluşmaktadır. Bu ses örneklerinin son bitlerinin değerindeki küçük artış veya azalışların oluşturacağı değişim işitme sistemi tarafından fark edilememektedir. Bu durum WAV dosyaları içerisine, ses örneklerinin son bitlerinin değiştirilerek veri gizlenmesi imkanı sağlamaktadır.

Literatürde ses dosyaları içerisine dönüştürme teknikleriyle [14] ve ses örneklerinin son bitlerinin modifikasyonu ile veri saklama [22] gibi çalışmalar mevcuttur. Bu tür çalışmaların sayısı son yıllarda artış gösterse de, resim içerisine veri saklayan çalışmaların sayısına göre daha azdır. Ülkemizde de steganografi alanındaki durum benzer şekildedir. Gri seviyeli Bitmap resimler içerisine veri saklayan Türkçe bir yazılım [62] ve renkli Bitmap resimler içerisine veri saklama hizmeti sunan Türkçe bir web sitesi [32] olmasına karşın, ses dosyaları içerisine veri saklayan ticari veya ücretsiz herhangi bir Türkçe güvenlik yazılımı bulunmamaktadır. Ülkemizde bilgisayar-internet kullanım oranları hızla artmaktadır [61]. Yüksek kapasitede, daha hızlı veri iletişimi teknolojilerinin gelişmesine paralel olarak ses dosyalarının da sayısal iletişimde sıklıkla kullanılmaya başlanmasına rağmen, bu konudaki gerek akademik çalışmaların azlığı gerekse ses içerisine veri saklayan bir Türkçe güvenlik yazılımının olmaması önemli bir eksiklik olarak değerlendirilmektedir.

Bu tezin amacı steganografik yaklaşımları incelemek, bu yaklaşımların bilgi güvenliği açıklarının giderilmesindeki katkılarını araştırmak, ülkemizde de konuya olan ilginin arttırılmasını sağlamak amacıyla konu ile ilgili steganografik yazılım geliştirmek ve sonuçta güvenlik sorunlarına steganografi bilimiyle çözüm geliştirerek ülkemizde daha etkin steganografi uygulamaları geliştirme yönündeki çalışmalara katkı sağlayabilecek analizlerde bulunmaktır. Ülkemizde ses tabanlı steganografi uygulamaları henüz yeterli seviyede olmadığı ve ses dosyalarının gelişen sayısal iletişim teknolojilerine paralel olarak ileride daha yaygın kullanılacağı değerlendirildiği için, bu çalışma kapsamında ses içerisine veri saklama analizleri yapılmış ve kişisel bilgi güvenliğinin sağlanmasında kullanılabilecek PCM

formatındaki WAV dosya seti içerisine klasör saklayan bir Türkçe steganografik yazılım geliştirilmiştir.

Bu tez çalışması yedi bölümden oluşmaktadır. Birinci bölümde bu çalışmanın gereği açıklanmıştır. İkinci bölümde steganografi konusu, tanımı, tarihçesi, steganografi yöntemleri, steganografik yazılımlar ile dünyada ve ülkemizde steganografi çalışmalarındaki son durum detaylı bir şekilde ele alınmıştır. Üçüncü bölümde, steganografinin tersini ifade eden, yani gizli iletişimin varlığını tespit etmeye çalışan steganaliz konusu ve yöntemleri incelenmiştir. Dördüncü bölümde, bu tez çalışması çerçevesinde gerçekleştirilen steganografik yazılımda kullanılan şifreleme algoritmalarına değinilmiş, bunun yanı sıra steganografiyle kriptografi arasındaki farklar da özetlenmiştir. Beşinci bölümde WAV dosyalarının farklı seviyelerdeki bit değerlerine veri saklanması üzerine analizler yapılmış, sonuçlar işitsel olarak ve histogram çizelgeleri ile spektrogram grafiklerinin karşılaştırılması yapılarak test edilmiştir. Altıncı bölümde ise, WAV dosya seti içerisine, DES, 3DES ve AES şifreleme algoritmalarıyla şifreleme imkanı da sunarak, belirlenen bir anahtar yardımıyla bir klasörü tüm içeriği ve hiyerarşisiyle saklayan, stego ve kriptografi tabanlı ilk Türkçe klasör kitleme yazılımının detaylarına, akış diyagramlarına, ekran görüntülerine, diğer yazılımlara göre avantaj ve dezavantajları ile spectrogram grafiklerinin analizine yer verilmiştir. Yedinci ve son bölüm olan sonuç bölümünde ise ses içerisine veri saklayan ve steganaliz ataklarına karşı dayanıklı steganografik yazılımların hangi özelliklere sahip olması gerektiği, ülkemizdeki steganografi çalışmalarında son durumun ne olduğu ve bu tezin ve geliştirilen yazılımın bu çalışmalara ne gibi katkılar sağlayacağı değerlendirilmiştir.

2. STEGANOGRAFI

Bu bölümde öncelikle steganografi bilimiyle ilgili temel konular ele alınmış ve konuyla ilgili literatür çalışmaları gözden geçirilmiştir. Steganografi konusuna genel bir bakış yapılmasının ardından ise steganografik yöntemler detaylı bir şekilde incelenmiştir. Son kısımda steganografik yazılımların tanıtılmasının ardından, steganografinin kişisel uygulamalarda kullanımı anlatılmıştır.

2.1. Steganografi

Steganografi resim, ses veya uygun formattaki bir sayısal dosyanın içerisine veri saklamada kullanılan bir bilgi güvenliği yaklaşımıdır. *Veri saklama* literatürde genel olarak *steganografi* terimini işaret etmektedir [3]. Steganografi kelimesi eski Yunan alfabesinden türemiş bir kelime olup *örtülü yazı* anlamına gelmektedir [25].

Steganografi hakkında literatürde çeşitli tanımlar yapılmaktadır. Bir tanıma göre steganografi gizli mesajın varlığının tespit edilemediği bir iletişim bilimidir [25]. Başka bir tanıma göre ise görünüşte zararsız bir mesajın içerisine veri saklama sanatıdır [12]. Bu bilim dalı askeri literatürde ise kısaca TRANSEC olarak adlandırılmaktadır [49,59].

Tanımlar farklı olsa da steganografide temel amaç iletişimin gizliliğinin sağlanmasıdır. Yani iki nokta arasında gerçekleştirilen iletişimde gizli bir veri aktarımının anlaşılmasının sağlanmasıdır. Bilgi güvenliğinde diğer bir önemli bilim dalı olan kriptografide ise asıl amaç verinin gizliliğinin sağlanmasıdır. Bu bağlamda yüksek seviyede güvenli bir iletişim için kriptografi ve steganografi bilimleri birlikte kullanılarak önce verinin daha sonra ise iletişimin gizliliğinin gerçekleştirilmesi yönüyle yüksek seviyede bir bilgi güvenliğinin oluşturulmasına katkılar sağlamaktadır.

Şu ana kadar gerek akademik gerekse endüstriyel çevrelerde steganografi, kriptografiye göre daha az ilgi çekmiştir [4]. Ancak 1996 da organize edilen ilk steganografi konferansıyla birlikte bu durum hızla değişmeye başlamıştır [4].

Sayısal veri dosya türlerinin çoğalmasıyla, özellikle son yıllarda, çok farklı dosya tipleri içerisine (resim, ses ve diğer sayısal dosya türleri) veri saklayan çalışmalar yapılmaya başlanmıştır [1,11,22,62,64,66].

Bender ve arkadaşlarının (1996) kaleme aldığı makalede, resim, ses ve metin gibi dosya türleri içerisine veri saklama teknikleri detaylı bir şekilde açıklanmıştır [10]. Ses içerisine, düşük bit kodlaması (Low Bit Encoding), faz kodlaması (phase coding), yayılmış spektrum (spread spectrum) ve yankı veri saklaması (echo data hiding) yöntemleriyle veri saklanması ve metin içerisine boşluk kullanımı, konuşma dilinin yapısı ve eş anlamlı kelimelerden faydalanarak veri saklama yöntemleri bu çalışmada ayrıntılı bir şekilde irdelenmiştir. Bölüm 3.2 ve Bölüm 3.3’de bu yöntemler detaylı olarak anlatılacaktır.

2000’li yıllardan sonra, LSB (Least Significant Bit) yöntemiyle [53,62], BPCS (Bit Plane Complexity Segmentation) yöntemiyle [55], dönüştürme tekniğiyle [56,68] ve permutasyon tekniğiyle [11] resim içerisine veri saklama çalışmaları gerçekleştirilmiştir.

Lee ve Chen (2000), LSB yöntemiyle ve anahtar kullanılarak, gri seviyeli resimlerde, piksel değerini oluşturan bitlerin ilk dördünün modifikasyonu ile, %50 ye yaklaşan kapasiteyle veri saklamışlardır [53].

Niimi ve ark. (2002), BPCS yöntemini temel alarak palet tabanlı resimler içerisine veri saklayan ve paletteki renk vektörlerinin sırasına bağlı olmayan bir metot geliştirmişlerdir [55].

Noda ve ark. (2002), kayıplı sıkıştırma gerçekleştiren resimler üzerinde BPCS yöntemiyle veri saklayan diğer bir çalışma yapmışlardır [56]. Bu çalışmada,

sıkıştırma işlemi esnasında, wavelet katsayılarının nicemleme (quantization) işlemiyle bit düzlemine döndürülmüş hali üzerinde BPCS yöntemiyle veri saklama gerçekleştirilmiş ve %9 ile %15 arasında değişen kapasitelerde veri saklanabilmektedir. Sağıroğlu ve Tunçkanat (2003), gri seviyeli Bitmap resimleri içerisine, görsel olarak fark edilmeksizin, en önemsiz 4. bit seviyesine kadar, LSB modifikasyonu yöntemiyle veri saklanabileceğini gösteren Türkçe bir yazılım geliştirmişlerdir [62].

Tseng ve Chang (2004), jpeg resimleri içerisine, yine sıkıştırma işlemi esnasında daha fazla saklama kapasitesi ile veri saklayabilen bir yöntem geliştirmişlerdir [68].

Brisbane ve ark. (2005), palet tabanlı renkli resimler içerisine, yüksek veri saklama kapasitesiyle, şeffaflığa zarar vermeksizin veri saklayabilen bir yöntem geliştirmişlerdir [11].

Akleyek ve Nuriyev (2005), geliştirdikleri açık anahtar ve gizli anahtar çiftiyle çalışan bir şifreleme sistemiyle, resim dosyalarının en az önemli bitlerini değiştirerek veri saklayan bir yöntem önermişlerdir. AS knapsack ismini verdikleri şifreleme sistemi sayesinde göndericinin alıcıya yolladığı veriyi inkar edememesi ve alıcının göndericinin ilettiği gerçek veriyi görebilmesi sağlanmıştır. Saklama sonrası oluşan resim dosyalarının fiziksel boyutunun, orijinal resim dosyasından farklı olduğu belirtilmiştir [2].

Shahreza (2006), mobil telefonlarda SMS (Short Message Service) altyapısıyla kullanılan siyah-beyaz resimler içerisine metin saklayarak, SMS yoluyla gizli mesajlaşmayı sağlayan bir çalışma sunmuştur [64]. Siyah-beyaz resimlerin, yapılacak modifikasyonlara karşı, renkli resimlere oranla çok daha dayanıksız olması ve saklama boyutunun küçüklüğü gibi dezavantajlarına rağmen, SMS’lerdeki siyah-beyaz resimler içerisine veri saklama yöntemi, İngiltere’de yakın zamanda hayata geçirilen Gizli-Metin SMS fonksiyonuna göre, 40 saniye içinde, alınıp görüntülenen mesajın imha edilmesinin gerekmemesi ve kullanılacak mobil telefonlarda WAP uyumluluğunun aranmaması gibi nedenlerle, daha avantajlı bir yöntem olarak değerlendirilmiştir.

Resim içine veri saklamalarına göre sayısı daha az olmakla birlikte özellikle son yıllarda, dönüştürme tekniğiyle [14], LSB modifikasyonu ile [1,15,22], tekrarlanan komutların varlığından faydalananarak [1] ve sıkıştırma esnasında [29] ses içerisine veri saklayan çalışmalar da gerçekleştirilmiştir.

Chou ve ark. (2001), transformasyon tekniklerini kullanarak, o dönemde mevcut olan ses içine veri saklama sistemlerinden daha fazla oranda veri saklanabileceğini tespit etmişlerdir [14].

Cvejic ve Seppanen (2002), LSB modifikasyonu ile ses içerisine veri saklama kapasitesini %33 oranında arttıran bir yöntem geliştirmişlerdir [15].

Gopalan (2003), ses dosyası içerisine, en az önemli bit modifikasyonu ile veri saklanması üzerine çalışma yapmış ve kokpit sesi gibi ses dosyaları içerisine daha fazla bit seviyesinde veri saklanabildiğini göstermiştir [22].

Adli ve Nakao (2005), internette sıkça kullanılan küçük boyutlu midi ses dosyaları içerisine, LSB yöntemi, tekrarlanan komut kodları algoritması ve sistem harici kodları algoritmasını kullanarak veri saklayan bir çalışmayı sunmuşlardır [1].

Resim ve ses dışındaki dosya türleri de veri saklama çalışmalarında kullanılmaya başlanmıştır [30,66].

Sui ve Luo (2004), yardımcı metin (hypertext) içerisine, boşluklar ekleme yerine biçimleme (markup) etiketlerinin pozisyonlarını değiştirerek veri saklayan bir çalışma sunmuşlardır [66]. Bu çalışmanın sonucunda, uyguladıkları yöntemin şu an için, her ne kadar ses ve resim içerisine veri saklama yöntemlerine göre henüz zayıf olsa da, ilerideki bilgi güvenliği sistemleri açısından potansiyel değerinin tahmin edilemez olduğunu rapor etmişlerdir.

Steganografi, zamanımızda yaygın olarak kullanılmaya başlamakla birlikte kökleri binlerce yıl öncesine dayanmaktadır [12]. Eski Yunan döneminde, hükümdar

Histaeus kölesinin saçını kestirip gizli mesajı kazıdıktan sonra tekrar saçı uzadığında köleyi mesajın ulaşması gereken yere göndermiştir [12,25]. Amerikan devrimi sırasında, ateşe tutulduğunda belirginleşen görünmez mürekkep, İngilizler ve Amerikalılar tarafından gizli iletişim amacıyla kullanılmıştır [25].

Steganografi 2.Dünya savaşında da kullanılmıştır. Aşağıda bir Alman ajanının steganografi yöntemiyle oluşturduğu bir metin yer almaktadır [12,25,49]:

“Apparently neutral’s protest is thoroughly discounted and ignored. Isman hard it. Blockade issue affects for pretext embargo on by-products, ejecting suets and vegetable oils”

Bu metindeki her kelimenin ikinci harfleri yan yana dizildiğinde anlamlı bir mesaj olan “Pershing sails for NY June 1” cümlesinin gizlendiği anlaşılmaktadır.

Verilen örneklerden de anlaşıldığı gibi, temel felsefesi açısından steganografi yeni bir kavram değildir ancak sayısal veri dosyası çeşitliliğinin artmasıyla birlikte çok farklı steganografi yöntemleri ve uygulamaları geliştirilmeye başlanmıştır. Steganografinin gerçek hayatta bir çok uygulaması mevcuttur [5]. Şu anda steganografi, güvenlik uygulamaları arasında en ilgi çekici konulardan birisidir.

Steganografi bilimi geliştikçe buna paralel olarak steganografiye özel bir terminoloji de oluşmaya başlamıştır [59]. Gizli mesajı saklamak için kullanılan medyaya (bu resim, ses veya uygun herhangi bir dijital dosya olabilir) *taşıyıcı (cover veya carrier)* denilmektedir. Saklanacak olan veriye ise *gömülü (embedded)* denilmektedir. Saklama işlemi sonucunda oluşan, orijinalinden ayırt edilemeyen medyaya ise *stego* denilmektedir. Saklama işleminde değişime uğrayan bitler içinse *barındırıcı (host)* kelimesi kullanılmaktadır. *Stego*, saklanan veriyle taşıyıcı medyanın bütünleşmiş haline denilmektedir. Bu durum aşağıdaki eşitlikle ifade edilebilir:

$$\text{taşıyıcı} + \text{gömülü} = \text{stego} \quad (2.1)$$

Saklama işleminde anahtar kullanılması durumunda ise aşağıdaki eşitlik söz konusu olacaktır:

$$\text{taşıyıcı} + \text{gömülü} + \text{anahtar} = \text{stego} \quad (2.2)$$

Bu tez çalışmasında bu tanımlara uygun olarak taşıyıcı dosya, stego dosya ve barındırıcı bitler gibi ifadeler kullanılmıştır.

Steganografi temel mantığı olarak binlerce yıldır uygulanan bir gizli iletişim yöntemi olmakla birlikte, günümüzde sayısal dosya çeşitliliğinin artmasıyla birlikte çok farklı veri saklama yöntemleri geliştirilmeye başlanmıştır. Steganografinin temelinde insan duyularının algılayamayacağı sınırlar içerisinde sayısal veri dosyalarının gizli mesajı içerecek şekilde modifikasyonu yatmaktadır. Örneğin görme sistemi (İGS) bir resimdeki çok küçük değişimleri fark edemez. İşitme sistemi de (İİS) belli bir seviyenin altındaki sesleri işitemez ve görme sistemine göre çok daha hassas olsa bile sesteki küçük değişimleri fark edemez. Steganografi ise görme ve işitme sistemlerinin zayıflıklarını kullanarak resim ve ses dosyaları içerisine veri saklanmasını desteklemektedir.

Steganografi sadece resim ve ses içerisine veri saklamakla sınırlı değildir. Bazı sayısal dosya türlerinde ise insanın işitme ve görme sistemlerinin zayıflıklarının değil dosya formatına ait bazı özelliklerin kullanılmasına dayalı steganografik yöntemlerin geliştirilmesi mümkündür. Örneğin çalıştırılabilir (exe) dosyaları içerisine aynı işlevi gören farklı komutların varlığından faydalanılarak, uygun bir kodlamayla mesaj saklamak mümkün olabilmektedir. HTML dosyalarında ise etiketlerin (tag) kapanış sırası görüntüyü değiştirmemektedir. Bu durumdan faydalanılarak ve yine uygun bir kodlama geliştirilerek HTML dosyalarını steganografik uygulamalarda kullanmak mümkündür.

Steganografide veri saklama amacıyla kullanılacak dosya türünde bir sınırlama yoktur. Gizli mesajı içeren bir dosyada hem fiziksel özellikleri açısından hem de görsel veya işitsel olarak fark edilir bir değişim oluşmadığı müddetçe, formatı uygun

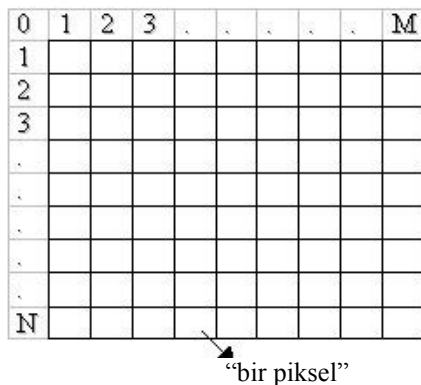
her türlü sayısal dosya veri saklamada kullanılabilir. Veri saklama amacıyla kullanılacak olan dosya yapısının ve modifikasyona toleranslı bölümlerinin çok iyi bilinmesi yüksek seviyede bir güvenlik için gereklidir. Takip eden başlıklarda farklı sayısal dosya türleri içerisinde veri saklama yöntemleri kısaca açıklanacaktır.

2.2. Resim İçerisine Veri Saklama

Şu ana kadar resim içerisine veri saklama konusunda birçok çalışma yapılmıştır [11,53,55-56,62,64-65,68]. Bu çalışmalarda jpeg formatındaki resimler içerisine [68], gri seviyeli bitmap resimler içerisine [62], siyah-beyaz resimler içerisine [64], LSB modifikasyonu, dönüştürme tekniği, maskeleye ve filtreleme gibi yaklaşımlarla [4,50] veri saklama bunlara verilebilecek örneklerdir. Bu yöntemler alt başlıklarda sırasıyla açıklanacaktır.

2.2.1. LSB modifikasyonu ve bitmap içerisine veri saklama

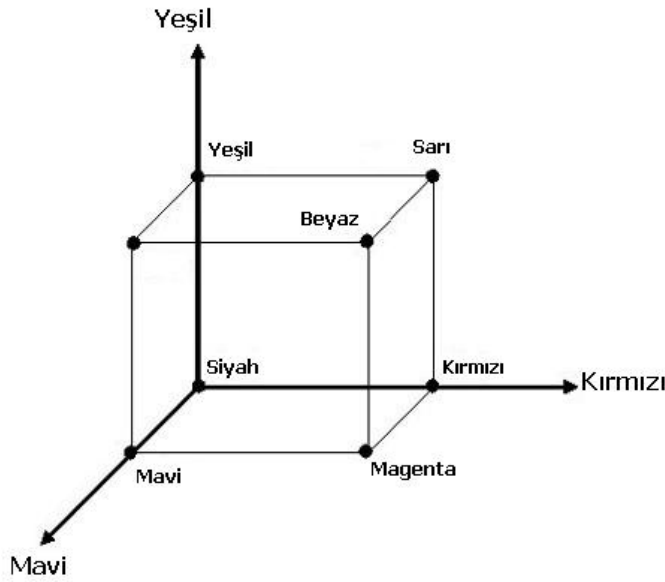
LSB modifikasyonu steganografide kullanılan en basit ve en yaygın yöntemdir [4]. Mesaj resmi oluşturan piksellerin son bitlerini değiştirerek saklanır ve sonuçta oluşan resim orijinalinden görsel olarak ayırt edilemez. Veri ardışık olarak saklanabileceği gibi bir anahtar yardımıyla rastgele olarak seçilen piksellere de saklanabilir.



Şekil 2.1. Resim dosyası yapısı

LSB modifikasyonu ile resim içerisine veri saklama bitmap resimler için aşağıdaki bölümde daha detaylı açıklanmıştır. Bitmap biçimindeki resim dosyalarına veri

saklamanın detaylarına girmeden önce bir resim dosyasının genel yapısı Şekil 2.1’de verilmiştir. Şekil 2.1’den de görülebileceği gibi bilgisayar açısından bir Bitmap resim dosyası piksellerin ışık değerlerini ifade eden iki boyutlu bir sayılar dizisidir. Yaygın olan resim boyutu 640x480 piksel ve 256 farklı piksel rengiyle oluşturulan resimlerdir. Sayısal resimler genel olarak 8, 16 veya 24 bitlik piksellerden oluşur [62]. 24 bitlik piksellerden oluşan resimler en fazla veri saklayabilen resimlerdir [50,62]. Bütün renk değerleri kırmızı, yeşil ve mavi (RGB) renklerin tonlarının karışımından elde edilir [50,62]. Temel renklerin her biri bir byte büyüklüğünde değerle ifade edilir ve 256 şar farklı renk tonuna sahiptir. Şekil 2.2’de renk küpü gösterilmektedir.

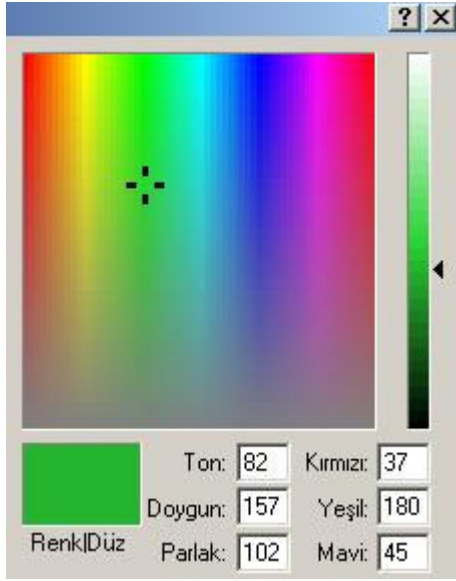


Şekil 2.2. Renk küpü [52]

24 bitlik bir resmin her bir pikseli 3 ana rengi ifade eden 3 byte boyutundaki piksellerden oluşur. Şekil 2.3’de örnek bir rengi oluşturan yeşil, mavi ve kırmızı renk yoğunlukları gösterilmektedir. 1024x768 boyutundaki 24 bitlik bir resim dosyasında her bir byte değerinin en az önemli bitleri veri saklama için kullanılsa toplam saklanabilecek veri boyutu:

$$1024 \times 768 \times 3 / 8 = 294912 \quad (2.3)$$

byte olmaktadır.



Şekil 2.3. Bir rengi oluşturan kırmızı, yeşil ve mavi tonların ait değerler

Daha önce de belirtildiği gibi LSB modifikasyonu İGS'nin küçük değişimleri ayırt edememesini kullanma mantığına dayanmaktadır. En az önemli bitlerine veri saklanan bir Bitmap resim dosyası orijinalinden görsel olarak farksız olmaktadır. Pikselleri oluşturan byte değerlerindeki değişimin oluşturduğu etki göz tarafından fark edilememektedir. Çalışmalar göstermiştir ki, sadece LSB değil, 4. seviyeye kadar ki bit değerlerinin modifikasyonu steganografide şeffaflığı bozmadan veri saklamak için başarılı bir şekilde uygulanmaktadır [62].

Resim içerisine LSB modifikasyonu ile veri saklamaya ilişkin aşağıda bir örnek verilmiştir. Saklanacak olan 9 bitlik bir verinin aşağıdaki bit değerlerinden oluştuğunu varsayalım:

1-1-0-1-0-0-1-1-0

Taşıyıcı resim dosyası ise 24 bitlik bir Bitmap resmi olsun. Bu durumda saklanacak olan mesajı barındırmak için 3 piksel yani 9 byte değerinin son bitleri yeterli olacaktır. Taşıyıcı dosyanın veri barındıracak olan piksellerinin en az önemli bitlerinin sırasıyla şu şekilde olduğunu varsayalım:

0-1-0-0-1-1-0-1-1

LSB modifikasyonundan sonra bu piksellerin en az önemli bitlerinin değerleri şu şekilde olacaktır:

1-1-0-1-0-0-1-1-0

Görüldüğü gibi 9 bit değerinin 6 tanesi (altı çizili olanlar) değişmiştir ancak bu tür değişimler görsel olarak ayırt edilebilir olmamaktadır. Geri getirme işleminde yeni piksel değerlerinin en az önemli bitleri sırayla okunup yan yana dizildiğinde saklanan mesaj doğru bir şekilde elde edilecektir.

2.2.2. Dönüştürme tekniği ve jpeg resimlerine veri saklama

Dönüştürme tekniği (transformation), dönüştürme alanına (transformation domain) ait katsayı değerlerinin bitlerinin modifikasyonu ile gerçekleştirilir [4]. AKT (Ayrık Kosinüs Transformasyonu) jpg resimlerinde kullanılan dönüştürme yöntemidir. Dönüştürme yöntemiyle veri saklama, resmin önemsiz alanlarında saklama yaptığından steganaliz ataklarına karşı daha sağlam olduğu değerlendirilebilir [4]. Dönüştürme tekniğiyle resim içerisine veri saklama jpeg formatındaki resimler için aşağıdaki bölümde daha detaylı açıklanmıştır.

Resimler sayısal iletişimde sıklıkla kullanılan bir medya tipidir. İnternette en çok kullanılan resim türü jpeg resimleridir. Bitmap resimleri sıkıştırma kullanmayan resimlerken jpeg türü resimlerde kayıplı sıkıştırma işlemi gerçekleştirilmektedir. Veri saklama işlemi ise sıkıştırma esnasında gerçekleştirilmektedir.

Sıkıştırma esnasında jpeg resmi 8x8'lik bloklara bölünür ve her bir blok üzerinde AKT dönüştürme işlemi gerçekleştirilir. Nicemlemeye tabi tutulan bu katsayılar daha sonra Run-Length ve Huffman kodlama teknikleriyle kodlanır. Nicemleme (quantization) sonrası oluşan AKT katsayılarının en az önemli bitleri veri saklamak

için kullanılabilir. Bu katsayılardaki yapılan değişiklikler resmin piksel değerlerini doğrudan etkilemediğinden oluşan değişim görsel açıdan ayırt edilemez. Bu durum jpeg resimlerini Bitmap resimlerine göre steganaliz ataklarına daha dayanıklı kılmakta ancak saklama kapasitesi Bitmap resimle kıyaslandığında çok düşük kalmaktadır. Jpeg resimleri üzerinde veri saklama algoritması gerçekleştiren çalışmaların neredeyse tamamı büyük oranlı saklama kapasitesine sahip değildir [68].

2.2.3. Maskeleye ve filtreleme

Maskeleye ve filtreleme yönteminde resmin, resim işleme teknikleri kullanılarak veri saklamaya en uygun alanları belirlenir ve saklama işlemi bu bölgelerde gerçekleştirilir. Genellikle 24 bitlik gri seviyeli resimlerde başarılı olan bir yöntemdir [4].

2.3. Ses İçine Veri Saklama

Resim dosyalarına benzer şekilde ses içerisine veri saklama yöntemleri de İİS'nin zayıflıklarını kullanmaktadırlar. 2000'li yılların başından itibaren ses içerisine veri saklama çok popüler hale gelmiştir. Ses dosyaları da steganografi uygulamalarında veri saklama amacıyla kullanılabilir. Bu konuda yapılan çalışmalarda genelde dönüştürme teknikleri [14] ve LSB modifikasyonu [15,22] gibi yöntemler kullanılmıştır. İİS çok geniş ve dinamik bir yelpazede işlediği için ses sinyalleri içerisine veri saklamak özellikle zordur [10].

Ses içerisine veri saklama yöntemleri düşük bit kodlaması (Low Bit Encoding), faz kodlaması (phase coding), yayılmış spektrum (spread spectrum) ve yankı veri saklaması (echo data hiding) olarak sınıflandırılmaktadır [10]. Düşük bit kodlaması, ses örneklerinin son bitlerinin saklanacak veriye göre değiştirilmesi yani LSB yöntemidir. Sesin analog ortamlara girmeksizin, tamamen sayısal ortamlarda transferi durumunda kullanılabilir. Faz kodlamasında ses dosyası bölümlere (segment) ayrılmakta ve bölümlere ait faz değeri, veriyi saklayacak şekilde yeniden

oluşturulmaktadır. Bu yöntem resim içine veri saklama yöntemlerinden dönüştürme tekniğine benzemektedir. Yayılmış spektrum yönteminde sese ait frekans spektrumu üzerinde veri gizlenmektedir. Bu yöntemde seste gürültüler oluşabilir. Yankı veri saklaması yönteminde ise ses sinyali üzerine yankı eklenmekte ve yankının farklı gecikme değerleri kodlanarak veri saklanabilmektedir.

Wav ve mp3 dosyaları sayısal ortamda en sık kullanılan ses dosyalarıdır. Aşağıdaki bölümlerde bu dosya türleri içerisine veri saklanması ayrıntıları yer almaktadır.

2.3.1. WAV dosyaları içerisine veri saklama

WAV dosyaları sayısal iletişimde sıklıkla kullanılan ses dosyası türlerinden biridir. Windows WAV dosyaları 8 veya 16 bitlik yani 1 veya 2 byte büyüklüğündeki değerlere sahip ses örneklerinden oluşmaktadır [24]. Ses örnekleri, doğası gereği, ses dalgasının belli bir andaki yaklaşık tahmini değerini ifade etmektedir [24]. Bu durum ses örneklerinin en az önemli bitleriyle oynama imkanı vermektedir.

Bir WAV dosyası temel olarak başlık ve veri bölümlerinden oluşmaktadır. Başlık kısmında yapılacak herhangi bir değişiklik, dosyanın yapısını bozup çalışmaz hale getirecektir. Ancak veri bölümündeki ses örneklerinin en az önemli bitlerinin modifikasyonu ses kalitesinde hissedilebilir bir fark oluşturmamaktadır. LSB yönteminde barındırıcı bit değeri değişime uğrarsa değeri saklanacak bite eşitleneceğinden ses örneğinin değerinde 1 artış veya 1 azalış gerçekleşecektir. WAV dosyalarının mp3 formatına dönüştürülmesi esnasında da veri saklama işlemi gerçekleştirilebilir. Bu değişim jpeg resimlerinde olduğu gibi ses örneklerinin değerini doğrudan değiştirmeyeceğinden steganaliz atakları karşısında daha sağlam olsa da, saklama kapasitesi açısından WAV dosyalarına göre dezavantajlı olmaktadır.

Sonuç olarak; İİS, İGS'ye göre çok hassas olmasına rağmen yine de küçük değişimler işitsel olarak hissedilememektedir. Bu durumdan steganografi uygulamalarında faydalanılarak veri saklama gerçekleştirilebilir. Bu tez

çalışmasında Windows WAV dosyaları içerisine LSB yöntemiyle veri saklama uygulaması geliştirilmiştir. Bu uygulama Bölüm 5’de detaylı olarak anlatılacaktır.

2.3.2. Mp3 içerisine veri saklama

Mp3 dosyaları kullanımı giderek yaygınlaşan sıkıştırılmış ses dosyalarından birisidir. Boyutunun küçük olabilmesi nedeniyle steganografik uygulamalar için uygun bir taşıyıcı dosya türüdür. Mp3 dosyaları içerisine veri saklayabilen az sayıda uygulama mevcuttur. Bunların en bilineni Mp3Stego yazılımıdır. Bu yazılım sıkıştırma esnasında Mp3 dosyaları içerisine veri saklayabilmektedir [29].

2.3.3. MIDI dosyaları içerisine veri saklama

MIDI dosyaları, müzikal performans için bir takım komutlar içeren küçük boyutlu ses dosyaları olup internette yaygın bir şekilde kullanılmaktadır [1]. Bir MIDI dosyasında en sık kullanılan komut olan “note on” komutunun sesi etkileyen parametresinin en az önemli bitlerinin değişmesi, işitilen seste farklılık oluşturmamaktadır ve bu durum setagnografi amaçlı kullanılabilir [1]. Ayrıca her bir komut, komut kodu ve komut parametreleri içermektedir ve ardışık komutların komut kodu aynı ise, ilk komut dışındakilerin komut kodu bilgisinin olup olmaması fark etmemektedir. Bu durum da steganografik amaçlı kullanılabilir [1]. MIDI dosyaları çok popüler ve küçük boyutlu olduğundan steganografik uygulamalarda kullanılması iyi bir seçim olabileceği vurgulanmaktadır.

2.4. Metin İçine Veri Saklama

Metin içerisine beyaz boşluklar kullanılarak veri saklanabilmektedir. Bu şekilde 3 ayrı yöntem kullanılmaktadır. Birinci yöntemde, her sonlandırıcı karakterin sonuna bir veya iki boşluk bırakarak kodlama gerçekleştirilir. İkinci yöntemde ise satır sonlarına boşluk bırakılarak kodlama yapılmakta ve veri saklanmaktadır. Üçüncü yöntemde de metnin sağa yaslanmasıyla oluşan boşlukların kodlanmasıyla veri saklama işlemi gerçekleştirilmektedir [10].

Metin içerisine veri saklama yöntemleri bunlarla sınırlı değildir. Örneğin bir metindeki kelimeler, ilk harfleri yan yana geldiğinde anlamlı bir mesaj oluşturacak şekilde sıralanarak da mesaj saklanabilir. Bununla ilgili olarak, 2. Dünya Savaşında gerçekleşmiş bir örnekten daha önce bahsedilmişti. Metin içerisine veri saklamayla ilgili ilginç bir uygulama spam mimic isimli bir web sitesi [27] tarafından gerçekleştirilmektedir. Bu sitede istenilen mesaj steganografik yöntemlerle kodlanarak spam e-posta metnine dönüştürülmektedir. Örneğin aşağıda yer alan metin spam mimic tarafından üretilmiş klasik bir spam örneğidir:

“Dear E-Commerce professional ;

This letter was specially selected to be sent to you . We will comply with all removal requests . This mail is being sent in compliance with Senate bill 2516 ; Title 2 ; Section 303 ! This is not multi-level marketing ! Why work for somebody else when you can become rich as few as 37 weeks . Have you ever noticed nobody is getting any younger plus people love convenience ! Well, now is your chance to capitalize on this ! We will help you increase customer response by 150% plus process your orders within seconds . You can begin at absolutely no cost to you . But don't believe us . Prof Simpson who resides in Montana tried us and says "Now I'm rich many more things are possible" . We are a BBB member in good standing ! DO NOT DELAY - order today ! Sign up a friend and your friend will be rich too ! Thank-you for your serious consideration of our offer ! Dear E-Commerce professional ; Your email address has been submitted to us indicating your interest in our newsletter . If you are not interested in our publications and wish to be removed from our lists, simply do NOT respond and ignore this mail . This mail is being sent in compliance with Senate bill 2716 ; Title 2 ; Section 305 ! This is different than anything else you've seen ! Why work for somebody else when you can become rich inside 38 weeks . Have you ever noticed more people than ever are surfing the web and more people than ever are surfing the web . Well, now is your chance to capitalize on this . We will help you deliver goods right to the customer's doorstep and decrease perceived waiting time by 100% ! You are guaranteed to succeed because we take all the risk . But don't believe us ! Prof Ames who resides in Wisconsin tried us and says "My only problem now is where to park all my cars" ! We assure you that we operate within all applicable laws ! We beseech you - act now . Sign up a friend and you'll get a discount of 30% ! Thanks . Dear Decision maker , Your email address has been submitted to us indicating your interest in our letter ! If you are not interested in our publications and wish to be removed from our lists, simply do NOT respond and ignore this mail ! This mail is being sent in compliance with Senate bill 2516 , Title 4 ; Section 307 ! THIS IS NOT A GET RICH SCHEME ! Why work for somebody else when you can become rich in 62 weeks ! Have you ever noticed nobody is getting any younger and most everyone has a cellphone ! Well, now is your chance to capitalize on this . WE will help YOU process your orders within seconds and turn your business into an E-BUSINESS ! You are guaranteed to succeed because we take all the risk ! But don't believe us . Mrs Jones who resides in Delaware tried us and

says "I was skeptical but it worked for me" . We assure you that we operate within all applicable laws ! If not for you then for your LOVED ONES - act now ! Sign up a friend and you'll get a discount of 80% ! Thank-you for your serious consideration of our offer .“

Bu metne bakıldığında kelimeler arasındaki boşlukların aslında gizli bir mesajın kodları olduğu anlaşılmamaktadır. Sıradan bir spam e-posta içeriği olabilecek bu metinde aslında “İki kişinin bildiği sır değildir” cümlesi saklanmıştır. Saklanan cümlelerin karakterlerine sayısal olarak karşılık gelen bit değerlerine göre spam metin içerisindeki boşlukların ayarlanmasıyla saklama işlemi gerçekleştirilmektedir. Steganografik uygulamalarda spam e-postaların potansiyel bir taşıyıcı olduğu değerlendirilmektedir [52].

Metin içerisine, metnin oluşturulduğu dilin söz dizim (syntax) yapısından ve eşanlamlı kelimelerin varlığından yararlanarak da veri gizlenebilmektedir [10]. Türkçe’de özne, nesne ve tümleçlerin cümledeki sırası anlamı değiştirmeden sıralabilir ve farklı sıralama kombinasyonları kodlanarak, bu özellik veri saklamada kullanılabilir. Örneğin, “Hayata bağlan Turkcell’le” cümlesindeki kelimelerin 3! Yani 6 farklı sıralanışı vardır ve vurgu farklılığı olsa da bütün sıralanışlar aynı anlamı vermektedir. Her bir sıralanış kodlanarak veri gizleme imkanı oluşturulabilir. Eşanlamlı kelimelerde de, kelimelerden birinin 0 diğerinin 1 değerini ifade etmesi kodlanarak metin içerisine veri saklanabilir.

Örneklerden de anlaşıldığı gibi söz dizim ve eşanlamlılık durumlarının kodlanmasıyla metin içerisine veri saklanabilmektedir. Ancak saklama kapasitesinin çok küçük değerlerde olacağını da not etmek gerekir.

2.5. HTML İçerisine Veri Saklama

Daha önce de bahsedildiği gibi bazı sayısal dosya türlerinde, dosyanın kendisine ait bazı özelliklerinden faydalanılarak steganografik amaçlı veri saklama işlemi gerçekleştirilebilmektedir. HTML dosyaları da bunlardan birisidir.

Sui ve Luo (2004), yardımcı metin (hypertext) içerisine biçimleme (markup) etiketlerinin pozisyonlarını değiştirerek veri sakladıklarını rapor etmişlerdir [66]. HTML kodlu web sayfalarında kullanılan etiketler (tag) aracılığıyla sayfanın görüntüsü bir biçimleme programlama diliyle şekillendirilir. HTML’de etiketlerin açılış ve kapanış bölümleri bulunur. Bir metne ait birden fazla etiket varsa bunların kapanış bölümlerinin sırası önemsizdir. Diğer bir ifadeyle bunların sırası oluşan görüntüyü etkilemez. Örneğin:

```
<td width="25%"><u><b><i>GAZİ</i></b></u></td>
```

satırı ile

```
<td width="25%"><u><b><i>GAZİ</b></i></u></td>
```

satırı aynı görüntüyü oluşturacaktır.

HTML etiketlerinin bu özelliğinden faydalanılarak steganografik uygulamalar geliştirilebilir. Yukarıdaki örneğe tekrar dönersek; hem koyu, hem eğik hem de altı çizili bir kelime oluşturmak için 3 ayrı etiket kullanmak gerekecektir. Bunların kapanış bölümlerinin sıralanışı görüntüyü değiştirmeyeceğine göre bu durum kodlanabilir.

Çizelge 2.1. Örnek HTML etiket kodlaması

Sıralama	Kod
</i></u>	1
</i></u>	2
</i></u>	3
</u></i>	4
</u></i>	5
</u></i>	6

Bu 3 etiketlin kapanış bölümleri 3! yani 6 farklı şekilde sıralanır. Her farklı sıralanışın 1 ile 6 arasındaki bir rakama denk gelmesini kodlarsak Çizelge 2.1'deki gibi bir kod tablosu elde edilir.

Şekil 2.4'de verilen görüntüye sahip taşıyıcı HTML dosyasına 5346 sayısını saklayacağımızı varsayalım:



Şekil 2.4. Taşıyıcı HTML dosyası

Bu taşıyıcı HTML dosyasına ait kaynak kod aşağıda verilmiştir:

```
<tr>
  <td width="25%"><u><b><i>GAZİ</i></b></u></td>
  <td width="25%"><u><i><b>ÜNİVERSİTESİ</b></i></u></td>
</tr>
<tr>
  <td width="25%"><u><b><i>GAZİ</i></b></u></td>
  <td width="25%"><u><i><b>ÜNİVERSİTESİ</b></i></u></td>
</tr>
```

Çizelge 2.1'de oluşturduğumuz kod tablosuna göre 5346 sayısını bu dosyaya sakladığımızda, yani etiketlerin kapanış bölümlerini 5-3-4-6 rakamlarına göre değiştirdiğimizde oluşan dosyanın kaynak kod aşağıdaki gibi olmaktadır.

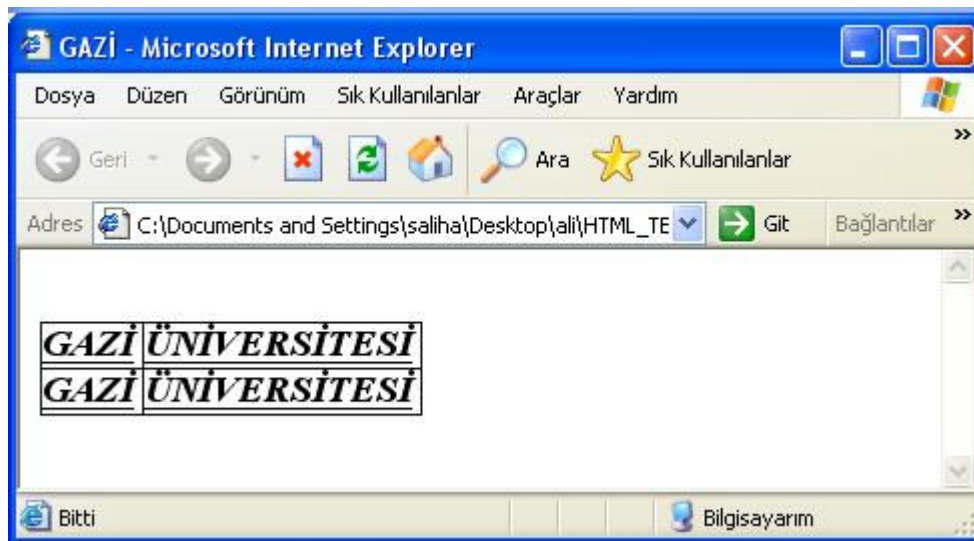
```
<tr>
  <td width="25%"><u><b><i>GAZİ</i></u></b></i></td>
```

```

<td width="25%"><u><i><b>ÜNİVERSİTESİ </b></i></u></td>
</tr>
<tr>
<td width="25%"><u><b><i>GAZİ</i></b></u></td>
<td width="25%"><u><i><b>ÜNİVERSİTESİ </b></i></u></td>
</tr>

```

Saklama sonucu oluşan bu dosyanın görüntüsü ise Şekil 2.5’de görüldüğü gibi orijinaliyle aynıdır:



Şekil 2.5. Stego HTML dosyası

HTML örneğine benzer şekilde başka dosya türlerinde de, o dosyaya ait bazı özellikler faydalanılarak veri saklama işlemi gerçekleştirilebilir.

2.6. XML (EXtensible Markup Language) İçine Veri Saklama

XML, HTML gibi bir biçimleme (markup) dilidir. Ancak HTML’den farklı olarak veri göstermeye değil tanımlamaya odaklı olarak tasarlanmıştır [28]. XML içerisine, aynı işlevi yerine getiren etiketlerin kodlanmasıyla veri saklanabilir. Örneğin imge etiketinin için `<img></img>` yazımı ile `<img/>` yazımı aynı işlevi görmektedir. Bu durumda `<img></img>` 1, `<img/>` ise 0 olarak kodlanacak olursa aşağıda verilen XML kodu içerisine 10001001 bitlerinden oluşacak bir karakterlik bir veri saklanmış olacaktır.


```

</img>



</img>


</img>

```

2.7. Çalıştırılabilir (EXE) Dosyalar İçerisine Veri Saklama

Aynı işlevi gören komutların varlığından faydalanılarak, çalıştırılabilir dosyalar içerisine de veri saklanabilmektedir. Hydan isimli program i386 komut setindeki tekrarlardan faydalanarak 1/110 kapasiteyle bir mesajı bir uygulama içerisine saklayabilmektedir [30].

2.8. Steganografik Uygulamalar Hakkında Genel Değerlendirmeler

Steganografi uygulamalarında kullanılacak dosya türünde bir sınırlama olmamakta, dosyanın yapısal özellikleri uygun olduğu müddetçe her türlü sayısal dosya veri saklamada taşıyıcı dosya olarak kullanılabilir. Örneğin MS Word dosyalarında metin rengiyle sayfa renginin aynı yapılmasıyla bir mesaj basitçe gizlenebilir ve bu durum steganografik amaçlı kullanılabilir. Steganografi uygulamalarında kullanılan taşıyıcı dosya türlerindeki çeşitlilik her geçen gün artmaktadır. Saklama işlemi sonucu oluşan yeni dosyanın şeffaflığının korunması ve saklanan verinin doğru bir şekilde tekrar geri getirilmesi sağlandıktan sonra saklama kapasitesi ve dosyanın iletişimdeki popülerliğini de göz önüne alarak, yapısı uygun her türlü dosya steganografi amaçlı kullanılabilir. Çizelge 2.2’de steganografik uygulamalarda kullanılan bazı dosya türleri yer almaktadır. Çok farklı sayısal dosya türleri içerisine veri saklama çalışmalarının gelecekte daha da artacağı değerlendirilmektedir.

Çizelge 2.2. Çeşitli taşıyıcı dosya türleri

	Dosya Uzantısı	Açıklama
1	BMP	Bitmap
2	JPEG	Joint Photographic Experts Group
3	PNM	Portable Any Map
4	GIF	Graphics Interchange Format
5	WAV	WAVEform sound
6	TXT	Metin dosyası
7	HTML	Hyper Text Markup Language
8	XML	eXtensible Markup Language
9	PCX	PC Paintbrush Graphic
10	PNG	Portable network graphic
11	AIFF	Audio Interchange File Format
12	MP3	MPEG-1 Audio Layer III
13	PDF	Portable Document Format
14	MIDI	Musical Instrument Digital Interface
15	EXE	Executable
16	DLL	Dynamic Link Library
17	TIFF	Tagged Image Format
18	DOC	Document

2.9. Steganografik Yazılımlar

Steganografi alanında ticari amaçlı veya ücretsiz bir çok yazılım geliştirilmiştir. Steganografik yazılımların çoğu stegoarchive.com sitesinde toplanmıştır. Aşağıda mevcut steganografik yazılımlardan bazıları kısaca tanıtılmaktadır.

2.9.1. OutGuess 0.2

OutGuess evrensel bir steganografi uygulamasıdır. Herhangi bir taşıyıcı veri içerisine saklama işlemi gerçekleştirmektedir. Taşıyıcı dosyanın barındırıcı bitlerini getirip,

veri saklandıktan sonra tekrar eklemeyi sağlayan bir araç sunulduğu müddetçe her türlü taşıyıcı dosya içerisine veri saklayabilmektedir. OutGuess 0.2 verisiyonu pnm ve jpg dosyalarını taşıyıcı dosya türü olarak doğrudan desteklemektedir [34].

2.9.2. Snow

Java ile geliştirilmiş bir steganografik uygulama olup ASCII metin içerisine, satırların sonuna boşluk ekleyerek veri saklamaktadır [36].

2.9.3. Puff

Bu yazılım İngilizce yardım dosyaları içeren İtalyanca bir steganografik yazılımdır. Bmp, jpg, pcx, png, tga, aiff, mp3 ve wav gibi çok farklı dosya türleri içerisine veri saklayabilmektedir. Ayrıca 16 farklı şifreleme algoritmasıyla, saklama işlemi öncesinde şifreleme imkanı da sunmaktadır [37].

2.9.4. GifShuffle

Gif resimleri içerisine veri saklayan bu uygulama, saklama işlemi öncesinde sıkıştırma ve şifreleme seçenekleri de sunmaktadır. Animasyon ve şeffaflık içeren tüm gif resimleriyle çalışabilmektedir [41]. Komut satırından çalışan Windows uyumlu bir yazılımdır [35].

2.9.5. Deogol

Perl ile geliştirilmiş olan bu uygulama komut satırından çalıştırılmakta olup html dosyaları içerisine, taşıyıcının boyutunu arttırmadan veri saklayabilmektedir [42].

2.9.6. Publimark 0.1.2

Komut satırından çalıştırılan bu uygulama ses dosyaları içerisine metin saklamaktadır.

Açık ve gizli anahtar çifti kullanmaktadır. Açık anahtarla saklanan veri ancak ilgili gizli anahtar ile geri getirilebilmektedir [43].

2.9.7. JPHS

JP Hide and Seek Windows 95/98/NT işletim sistemleriyle uyumlu olan ve jpg resimleri içerisine veri saklayan bir uygulamadır [35].

2.9.8. MP3Stego

MP3Stego mp3 dosyaları içerisine veri saklayan bir uygulamadır. Mp3 dosyalarının yüksek sıkıştırma oranından faydalanılarak veri saklama gerçekleştirilmektedir. Veri saklama Level III kodlaması esnasında gerçekleştirilir [29].

2.9.9. Steghide 0.5.1

Steghide çeşitli resim ve ses dosyaları içerisine veri saklayabilen bir steganografik uygulamadır. Sıkıştırma ve şifreleme imkanı da sunmaktadır. Jpg, bmp, wav ve au dosyalarını taşıyıcı olarak desteklemektedir [44].

2.9.10. Hydan

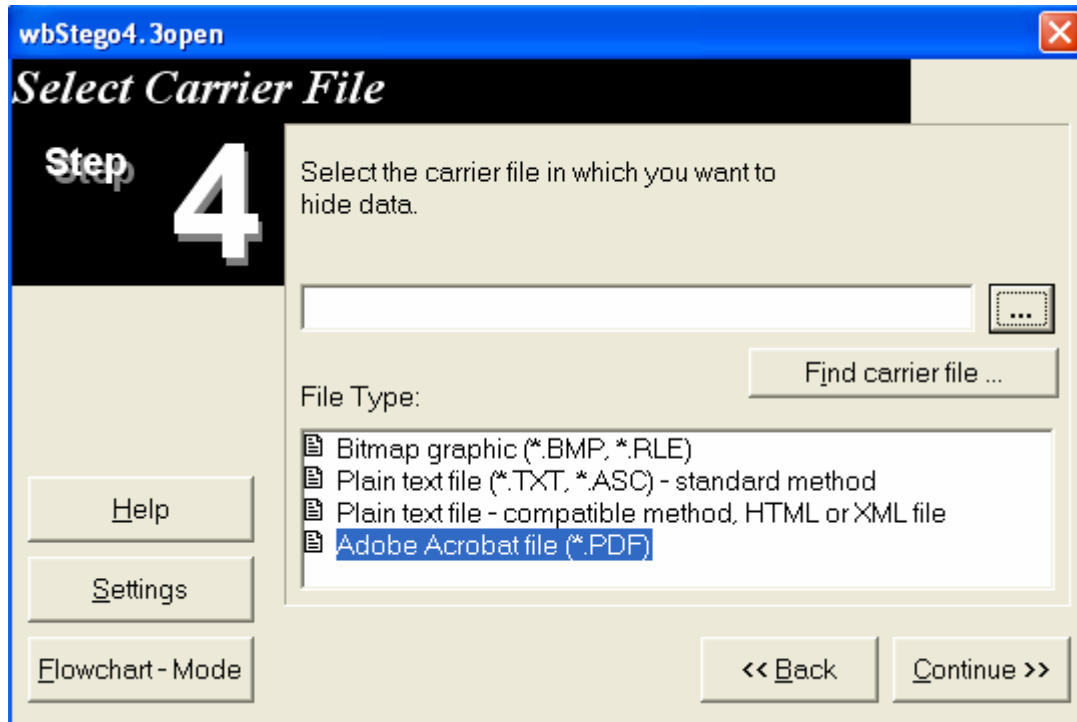
Hydan i386 komut setindeki aynı işlevi gören komutların varlığından faydalanarak exe dosyaları içerisine veri saklayan bir steganografik uygulamadır.

Hydan 1/110 saklama kapasitesine sahiptir [30].

2.9.11. wbStego4open

Windows ve Linux işletim sistemleri üzerinde çalışan açık kaynak bir steganografi uygulamasıdır. Bitmap, html, txt ve pdf türündeki dosyalar içerisine her türlü sayısal veriyi saklayabilmektedir [38]. İngilizce ve Almanca versiyonları mevcuttur [35].

wbStego4open yazılımının temel aldığı wbStego4 yazılımının ekran görüntüsü Şekil 2.6’da verilmiştir:



Şekil 2.6. wbStego4 yazılımının ekran görüntüsü

2.9.12. SecureEngine professional 1.0

Bu uygulama htm, Bitmap, gif ve png dosyaları içerisine veri saklayabilmekte olup Windows XP işletim sistemi üzerinde çalışmaktadır.

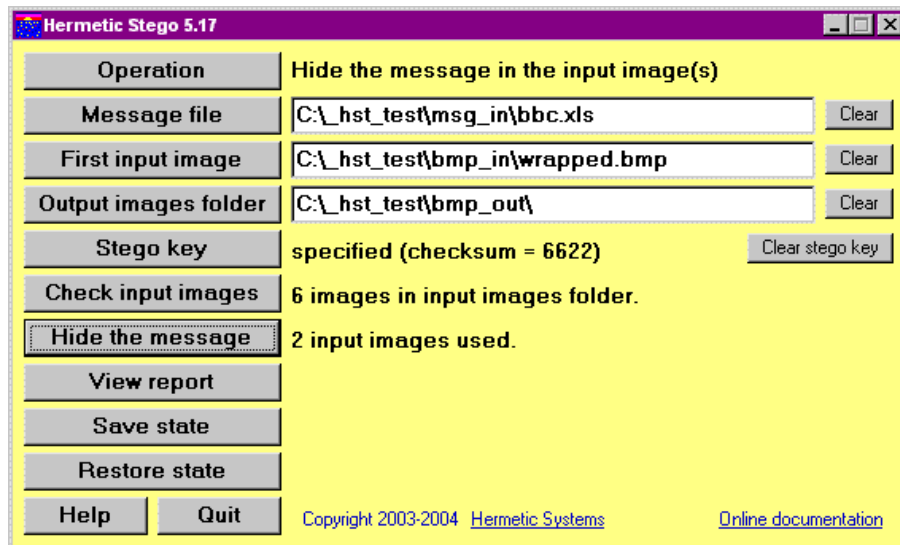
Saklama işlemi öncesinde veri AES, BlowFish, Gost veya Triple-DES algoritmalarından biriyle şifrelenebilmektedir. Ticari olmayan kullanımlar için yazılım ücretsizdir [39]. Uygulamanın ara yüz ekranı Şekil 2.7’de verilmiştir.



Şekil 2.7. SecureEngine Professional 1.0 yazılımının ara yüzü

2.9.13. Hermetic stego

Kullanışı kolay olan bu yazılım Bitmap resim seti içerisine veri saklamaktadır. Yeterli miktarda taşıyıcı resim dosyası sağlandığı müddetçe saklanacak olan verinin boyutunda herhangi bir sınırlama yoktur. Hem rastgele saklama hem de şifreleme için bir stego anahtar kullanılmaktadır [40]. Deneme sürümünün ara yüzü Şekil 2.8’ de sunulmuştur.

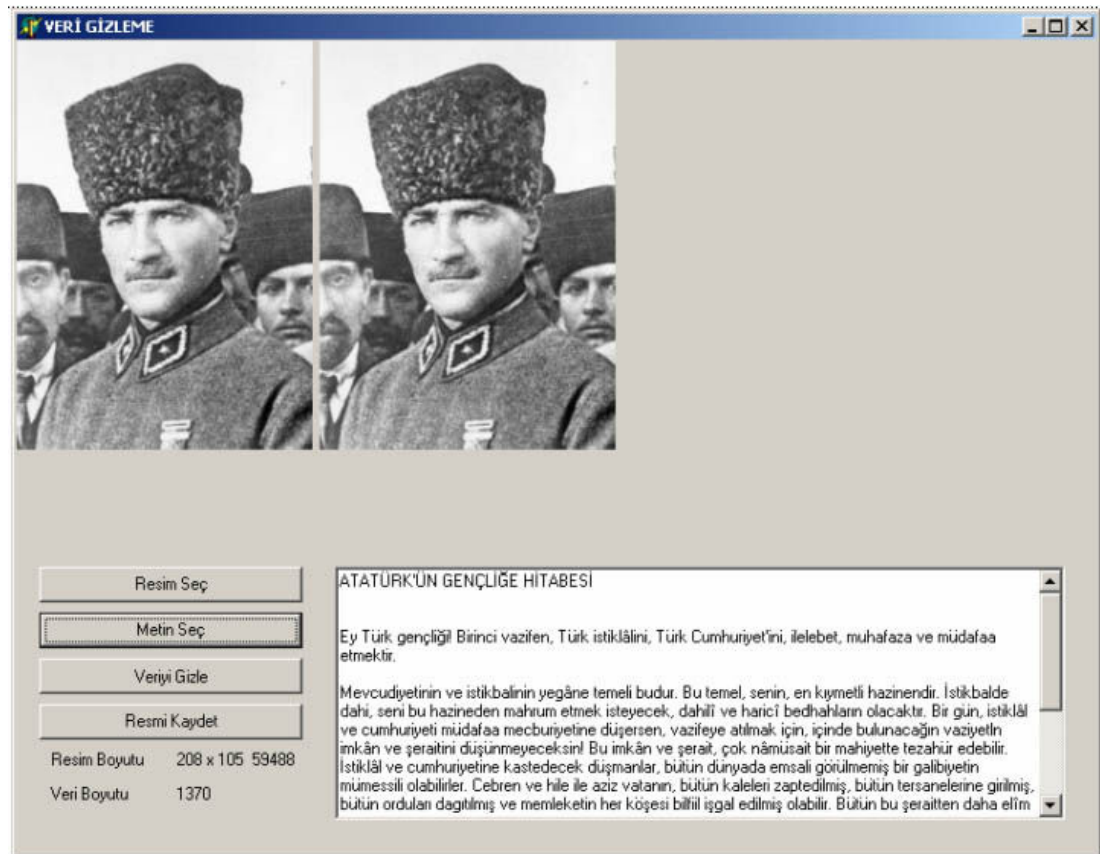


Şekil 2.8. Hermetic Stego demo versiyonu ara yüzü

2.10. Türkiye’de Yapılan Steganografik Çalışmalar

Steganografi tüm dünyada olduğu gibi ülkemizde de henüz çok yeni bir bilim dalıdır. Türk akademisyen ve mühendisleri tarafından özellikle 2000’li yıllardan sonra steganografi üzerine tezler ve makaleler yazılmaya başlanmış ve steganografik uygulamalar geliştirilmiştir.

Steganaliz konusunda son birkaç yıl içerisinde Türk akademisyenler tarafından çeşitli makaleler yayınlanmıştır [7-9,63]. Steganografi konusunda ise 2003 yılında gri-seviyeli resimler içerisine veri saklayan TürkSteg yazılımı geliştirilmiş ve bu yazılım konuya olan ilginin artmasına katkıda bulunmuştur [62].



Şekil 2.9. TürkSteg yazılımı ara yüz ekranı

Yine Türk akademisyenlerce, steganografi konusundaki önemli çalışmaların gerçekleştirildiği bir ortam sunan steganografik kütüphane oluşturulmuştur [33].

Bu kütüphanede dört ana paket yer almaktadır. Bunlar:

- Yer deęiřtirme algoritmalarını temel alan yöntemler paketi
- Frekans düzleminde katsayıların deęiřtirilmesini temel alan yöntemler paketi
- Yayılmış spektrum kullanımını temel alan yöntemler paketi ve
- İstatistiksel algoritmaları temel alan yöntemler paketleridir.

Önceki paragraflarda belirtildięi gibi, Türkiye’de son yıllara resim içine veri saklayan çeřitli yazılımlar geliştirilmiřtir. Saęıroęlu ve Tunçkanat (2003), gri seviyeli Bitmap resimleri içersine, 1. seviyeden 4. LSB seviyesine kadar yapılan farklı denemelerle, steganografik amaçlı kullanılmak üzere TürkSteg yazılımını geliřtirmiřlerdir. TürkSteg’e ait ekran görüntüsü Şekil 2.9’da sunulmuřtur.

Veri Gizle

Önce taşıyıcı olarak kullanacaęınız yani içine veri gizleyeceęiniz resmi seçin, daha sonra anahtar kelimeyi giriniz şifreleme işlemi bu anahtara göre yapılacak, son olarak gizlemek istedięiniz yazıyı yazın ve veri gizleme işlemini başlatın.

Hedef Resim Bilgisayarında:

Gözet...

(sadece .bmp resim seçin!)

Anahtar :

Gizlenecek Yazı:

Gizle >>

Veri Çöz

Önce içinde gizlenmiş veri olan taşıyıcı resim dosyasını seçin, daha sonra anahtar kelimeyi girip veri çözme işlemini başlatın.

Kaynak Resim Bilgisayarında:

Gözet...

(sadece .bmp resim seçin!)

Anahtar :

Çöz >>

Şekil 2.10. verigizle.com sitesinin işlem ekranı görüntüsü

Resim içine veri saklamaya yönelik bir dięer uygulama ise verigizle.com sitesi tarafından sunulan web tabanlı steganografik sistemdir. Bu sitede kullanıcılar

seecekleri bir Bitmap resim ierisine, sıkıştırma ve şifreleme seenekleriyle mesaj saklayabilmekte ve ierisine veri gizlenmiş bir resim dosyasından saklanmış mesajı geri getirebilmektedirler. Sitenin ekran görüntüsü Şekil 2.10’da verilmiştir.

Ülkemizde steganografi ve steganaliz konusunda çeşitli çalışmalar yapıılıyor olsa da, genel olarak ve özellikle ses ierisine veri saklamaya yönelik çalışmaların henüz yeterli olmadığı değerlendirilmektedir.

2.11. Steganografi Temelli Kişisel Güvenlik Uygulamaları

Kişisel veya kurumsal veriler depolandığı ortamlarda gerek fiziksel gerekse yazılımlar ve donanımlarla çok iyi korunsalar bile, bu verilerin iletişiminin gizliliği tam olarak sağlanamazsa verilerimiz yine tehdit altında olacaktır. Bilgisayar ve internet kullanımının yaygınlaşmasına paralel olarak verinin farklı noktalar arasında transferi de kolaylaşmış ve yaygınlaşmıştır. Önceki bölümlerde anlatılan steganografi yöntemleri, kişisel uygulamalarımızda sayısal olarak iletilen verimizin güvenliğini ve gizliliğini sağlama imkanı sunmaktadır.

Günümüzde elektronik posta veya herhangi bir manyetik medya ile kişisel verilerimizi transfer etmek durumunda kalabiliyoruz. Bankamatik şifrelerimizi, kimlik bilgilerimizi veya sadece kendimizi ilgilendiren özel bilgilerimizi elektronik posta ile iletmek zorunda kalabiliyoruz. Bunları direk iletmek olumsuz sonuçlar doğurabilmektedir. Bundan korunabilmek için verimizi hem şifreleyen hem de fark edilmeyecek şekilde bir başka sayısal veri ierisine saklayan steganografik sistemler kullanmamız gerekmektedir. Örneğin bir arkadaşımıza, sadece kendi aramızda bilinmesini isteyeceğimiz bir mektup yollamak istiyoruz. Bu durumda istediğimiz şey mektubumuzun aradaki etkenler tarafından erişilmesini engellemek ve mektuplaşıldığının bilinmesini önlemek olacaktır. Bunu sağlayabilmek için ücretsiz olarak kullanıma sunulan TürkSteg yazılımı veya verigizle.com sitesi aracılığıyla göndereceğimiz mektubu önce şifreleyip sonra da bir resim ierisine saklayıp, karşı tarafa bu resmi gönderebiliriz. Böylece aradaki etkenler arada sadece bir resim

transferi olduğunu görecek fakat aslında bu resim aracılığıyla kişiye özel bir mektubun iletildiğinin farkında olmayacaktır.

Önceki bölümlerde steganografi uygulamalarının çeşitliliğinden bahsedilmişti. Çizelge 2.6’da listelendiği üzere bir çok farklı sayısal dosya türü içerisine veri saklanabilmektedir. HTML dosyaları da bunlardan birisidir. Şu anda bir çok ücretsiz web sitesi açma imkanı sunan siteler mevcuttur. Hazırlayacağımız HTML sayfalarını, hizmeti sunan web sitesine göndererek istediğimiz bir içeriği web üzerinden yayınlatabiliriz. HTML dosyalarına veri saklama özelliğini kullanarak küçük boyuttaki ve belli bir süre için gizli kalması gereken verimizi ücretsiz olarak basit bir web sayfası içeriğinde saklayarak bir başka ülkedeki arkadaşımıza bu şekilde de mesaj yollayabiliriz.

Bölüm 6’da anlatılan ve bu tez çalışmasında geliştirilen STKK yazılımı şifreleme ve steganografi bilimlerinin her ikisini de kullanan bir klasör kitleme yazılımı olarak başkaları tarafından erişilmesini ve görülmesini istemediğimiz dosya ve klasörlerimizi saklama imkanı sunmaktadır. Özetlemek gerekirse, kişisel verilerimizi çok farklı steganografi yöntemlerini veya steganografik yazılımları kullanarak güvenli bir şekilde iletebiliriz.

3. STEGANALİZ

Steganografinin gelişmesine paralel olarak gelişen diğer biri bilim dalı ise steganalizdir. Steganalizin amacı ise steganografinin tam tersine bir resim, ses veya herhangi bir dosyadaki gizlenmiş verinin varlığını ortaya çıkarmaktır. Steganaliz orijinal objelerle stego objeler arasındaki ayrımı tespit etmek üzere tasarlanmış teknikler bütünü olarak ifade edilmektedir [57].

Steganaliz çalışmalarının temelinde saklanan verinin taşıyıcı nesne üzerinde bir takım parmak izleri bıraktığı düşüncesidir. Yani saklama işlemi sonrasında oluşan stego nesne görsel, işitsel veya işlevsel olarak orijinalinden ayırt edilemez olsa da istatistiksel olarak farklılıklar taşıyabilmektedir.

Steganaliz konusunda özellikle resim ve ses dosyalarının steganalizinde 2000’li yıllarda birçok çalışma yapılmıştır [7-9,16-17,19-21,57-58,60,63,69].

Bir steganografi uygulamasının güvenliği, stego nesnelerin ayırt edilebilir farklılıklar içermemesiyle ölçülmektedir [18]. Eğer stego nesne içerisinde veri saklı olduğunu rastgele yapılacak tahminlerden daha iyi oranla tahmin eden bir algoritma mevcutsa o steganografi sistemi kırılmış anlamına gelmektedir [18].

Gizli mesajın varlığının tespit edilmesi, mesajın uzunluğunu tahmin edilmesi ve gizli verinin tamamen ortaya çıkarılması konuları steganalizin sahasına girmektedir [18]. Steganografik uygulamalara karşı geliştirilen steganaliz ataklarından bazıları şunlardır [31]:

- Stego dosyanın bilindiği atak (stego-only)
- Taşıyıcı dosya ile stego dosyanın ikisinin de bilindiği atak (known cover)
- Saklı bir mesajın bilindiği atak (known message)
- Algoritma ve stego dosyanın bilindiği atak (chosen stego)
- Algoritma ve saklı bir mesajın bilindiği atak (chosen message)

Steganografide kullanılan saklama algoritmasına ihtiyaç duyulmaksızın evrensel tespit yapabilen steganaliz çalışmalarının sayısı artmaktadır. ANNTS (Artificial Neural Network Technology for Steganalysis) isimli projede de yapay sinir ağıları kullanılarak dijital resimler istatistiksel olarak analiz edilip saklı bir veri olup olmadığının evrensel olarak tespit edilmesine çalışılmaktadır [54].

Şu anda internette herkese açık çok fazla ve basit steganografi uygulamaları olmasına rağmen, güvenilir tespit yapabilen steganaliz sistemleri çoğunlukla mevcut değildir [18]. Gelecek günlerde bu ve buna benzer çalışmalarda yoğun artışlar olacağı değerlendirilmektedir. Aşağıdaki bölümlerde resim ve ses steganalizi konusunda yapılan çalışmalar ve geliştirilen yöntemler daha detaylı bir şekilde anlatılacaktır:

3.1. Resim Steganalizi

Resim steganalizi konusunda birçok çalışma yapılmıştır [8-9,16-17,19-21,58,63,69]. Westfeld ve Pfitzmann (2000), resimlere mesajın saklanması sonucu değişen değer çiftlerinin istatistiksel analizine dayanan bir steganaliz metodu geliştirdiler. Bu yöntem ardışık saklama gibi durumlarda daha güvenilir sonuçlar vermiştir [69].

Fridrich ve ark. (2000), 24 bitlik renkli resimlerde LSB yöntemiyle gerçekleştirilen veri saklamayı tespit etmek için, birbirine yakın renk değeri çiftlerinin istatistiksel analizine dayanan bir çalışmada tekil renk oranının toplam piksel sayısının %30 undan az olduğu resimlerde güvenilir sonuçlar verebilen bir yöntem sunmuşlardır. Bu yöntemin sıkıştırılmamış yüksek çözünürlüğe sahip, yani günümüzdeki sayısal fotoğraf makinelerinin çektiği resimlerde başarılı olamayacağı ve ayrıca gri seviyeli resimlere bu yöntemin uygulanamayacağını bildirmişlerdir [20].

Provos ve Honeyman (2001), jpeg resimlerinin steganalizi üzerine kapsamlı bir çalışma yaptılar ve geliştirdikleri yazılımla internet üzerinde binlerce şüpheli resmi tespit edebilmişlerdir [58].

Farid (2001), taşıyıcı ve stego resimlerden oluşan bir veritabanı üzerinde yaptığı analizlerden ve özellik çıkarımlardan sonra steganografi algoritmasının bilinmesine ihtiyaç duymaksızın steganaliz tahmininde bulunan evrensel tespit çalışmasını sunmuştur [16].

Fridrich ve ark. (2001), renkli ve gri seviyeli resimlerin steganalizi üzerine RS (Regular - Singular) ismini verdiği bir metot geliştirmişlerdir. Bu metotta veri saklama sırasında değişen değer çiftlerinin yanı sıra bu değişimin gerçekleştiği piksel konumlarının da istatistiksel analizi yapılmıştır. Bu yöntemin başarısı da gömme işleminin rastgele olduğu durumlarda ardışık saklamaya göre daha düşüktür [17,19].

Fridrich (2006), veri saklama sonucu oluşan değişim ile değişim sayısı arasındaki ters orantıyı analiz ettiği çalışmada, herhangi bir tespit edilebilirlik profilinde, (tespit edilebilirlik profili gömme etkisinin dağılımını ifade etmektedir) sadece en küçük gömme etkisine sahip piksellerin kullanılmasının asla optimum olmadığını belirlemiştir [21].

Resimler için steganaliz yaklaşımlarından bazıları şunlardır [18]:

- Görsel tespit
- Histogram analizleri
- Yüksek seviyeli istatistiksel analizler (RS analizi) Algoritmaya ve türe özel tespitler ve
- Evrensel tespit sistemleridir.

Görsel tespit, veri gömme sonucu oluşan resmin gözle orijinalinden farkının anlaşılması veya orijinali yoksa resmin üzerindeki normal bir resimde olması beklenmeyen bozuklukların fark edilmesiyle gerçekleşir. Ancak bu durumda da kendinden gürültülü resimlerle stego resimleri ayırt edebilmek mümkün olmayacaktır. Görsel atakların güvenilirliğinin tartışılır olduğu açıktır.

Histogram analizleri resme veri gömme sonucu değişen piksel değeri çiftlerinin istatistiksel analizlerini esas almaktadır.

Yüksek dereceli istatistiksel analizler ise histogram analizinden fazla olarak değişimin gerçekleştiği piksellerin konumlarına yönelik istatistiksel analizleri de içermektedir.

Algoritmaya veya türe özel tespitler de algoritması bilinen steganografi uygulamalarıyla oluşturulan stego resimlerdeki mesajı ortaya çıkarmaya yönelik yöntemlerle jpeg gibi resim türlerine özel steganaliz çalışmalarını içermektedir.

Evrensel tespit sistemleri ise resme ait bazı özelliklerin çıkarımıyla orijinal resimlerle stego resimler arasındaki ayrımı yapmaya yönelik yöntemleri içermektedir.

3.2. Ses Steganalizi

Resim steganalizi konusundaki kadar çok olmasa da ses steganalizi konusunda da bazı çalışmalar yapılmıştır [7,57,60].

Ru ve ark. (2006), mevcut bazı steganografi yazılımlarıyla WAV dosyaları içerisine yapılan gömme işlemini tespit etmeye yönelik bir yöntem geliştirmişlerdir [60].

Özer ve ark. (2006), saklama algoritmasının bilinmesine gerek duyulmaksızın, ses dosyaları içerisinde saklı mesajın varlığını evrensel bir şekilde tespit etmeye yönelik yaptıkları çalışmada %75 ile %90 arasında değişen bir başarı yüzdesi elde etmişlerdir [57].

Ses steganalizi konusundaki yaklaşımları ise aşağıdaki şekilde listeleyebiliriz [18]:

- İşitsel tespit
- Algoritmaya özel tespitler ve
- Evrensel tespit yoludur.

Bu yaklaşımları kısaca açıklayalım:

İşitsel tespit, stego ses dosyasının ses kalitesi yönüyle orijinalinden ayırt edilmesini veya orijinal dosya elde yoksa normal bir seste olmaması gereken bozuklukların tespitini ifade eder. Ancak kendinden gürültülü seslerle stego sesleri ayırt etmek yine kolay olmayacaktır.

Algoritmaya özel tespitler, belirli algoritmalarla yapılan saklama işlemlerinin varlığını tahmin etmeye çalışan yöntemlerdir.

Evrensel tespit ise orijinal ses dosyaları ile stego ses dosyaları arasındaki çeşitli özelliklere göre farklılıkları belirleyip, bir ses dosyasının stego ses dosyası olup olmadığını tespit etmeye yönelik yöntemleri içermektedir.

3.3. Steganaliz Yazılımları

Steganografik yazılımlar kadar fazla olmasa da, steganaliz konusunda da gerek ticari amaçlı gerekse ücretsiz çeşitli yazılımlar geliştirilmiştir. Bunlardan bazılarını aşağıda değinilmiştir:

3.3.1. StegSpy

StegSpy yazılımı dosyada veri saklı olup olmadığını ve saklıysa hangi yazılımla saklandığını tespit edebilmektedir. Evrensel tespit yapamıyor olsa da, JPHideandSeek, Invisible Secrets gibi birkaç steganografik yazılımla oluşturulan stego dosyaları tespit etmektedir. Yazılım ücretsiz olarak indirilebilmektedir [45].

3.3.2. Stego suite

Stego Suite resim ve ses içerisine saklanmış verinin varlığını, saklama algoritmasının bilinmesine ihtiyaç duymadan evrensel olarak tespit edebilen en gelişmiş ticari amaçlı steganaliz yazılımlarından birisidir [46]. Bmp, gif, png, jpg ve wav dosyalarının steganalizini yapabilmektedir [47].

3.3.3. Stegdetect

Outguess 0.1, Invisible Secrets gibi yazılımlarla içerisine veri saklanmış jpeg resimlerini tespit edebilen ücretsiz bir steganaliz yazılımıdır [48].

3.4. Steganaliz Konusunda Değerlendirmeler

Steganaliz konusunda yukarıdaki anlatılanlar çerçevesinde, başarılı yani şeffaflığı sağlayacak bir steganografi uygulaması için şu değerlendirmeleri yapabiliriz:

- Taşıyıcı olarak kullanılacak resim veya ses dosyasının bilinen orijinali kolay elde edilebilir dosyalar olmaması gerekir. Görsel veya işitsel steganaliz ataklarına karşı bu gerekli olabilir.
- Renkli resimlere uygulanabilen RQP (Raw Quick Pairs) gibi bazı istatistiksel analizler, gri seviyeli resimlere uygulanamaz. Bazı steganografi uzmanları gri seviyeli resimleri en iyi taşıyıcı resimler olarak tavsiye etmektedir [6].
- Renkli resimlerde renk çeşitliliği fazla olan sıkıştırılmamış yüksek çözünürlüğe sahip resimler daha uygun olmaktadır.
- Saklama oranı, saklama kapasitesini düşürecek olmasına rağmen istatistikselsteganaliz karşısındaki dayanıklılığı arttırması açısından yeterince düşük tutulmalıdır.
- İletişim gizliliğini arttırmak için saklama işlemi ardışık değil belirli bir anahtar değerine göre rastgele yapılmalıdır.

4. ŞİFRELEME VE ÖZETLEME ALGORİTMALARI

Şifreleme bilimi çok geniş bir güvenlik konusudur. Veri iletişimi esnasında, iletilen verinin aradaki bir etken tarafından olduğu gibi elde edilememesi için verinin anlamsız bir biçime dönüştürülmesine şifreleme, bu dönüştürme işleminde kullanılan algoritmalar ise şifreleme algoritmaları olarak bilinmektedir. Şifreleme yaklaşımları simetrik ve asimetrik olmak üzere ikiye ayrılır [61]. Asimetrik şifreleme algoritmaları bu çalışmanın kapsamında yer almadığından bu bölümde sadece, bir sonraki bölümde anlatılacak olan ve bu tez çalışması çerçevesinde geliştirilen steganografi temelli klasör gizleme yazılımında kullanılan simetrik şifreleme algoritmaları ve özetleme algoritmalarına değinilecektir.

4.1. Simetrik Şifreleme Algoritmaları

Simetrik şifreleme algoritmaları gizli anahtarlı geleneksel algoritmalar olup çoğunda şifreleme anahtarı ile şifre çözme anahtarı aynıdır [61].

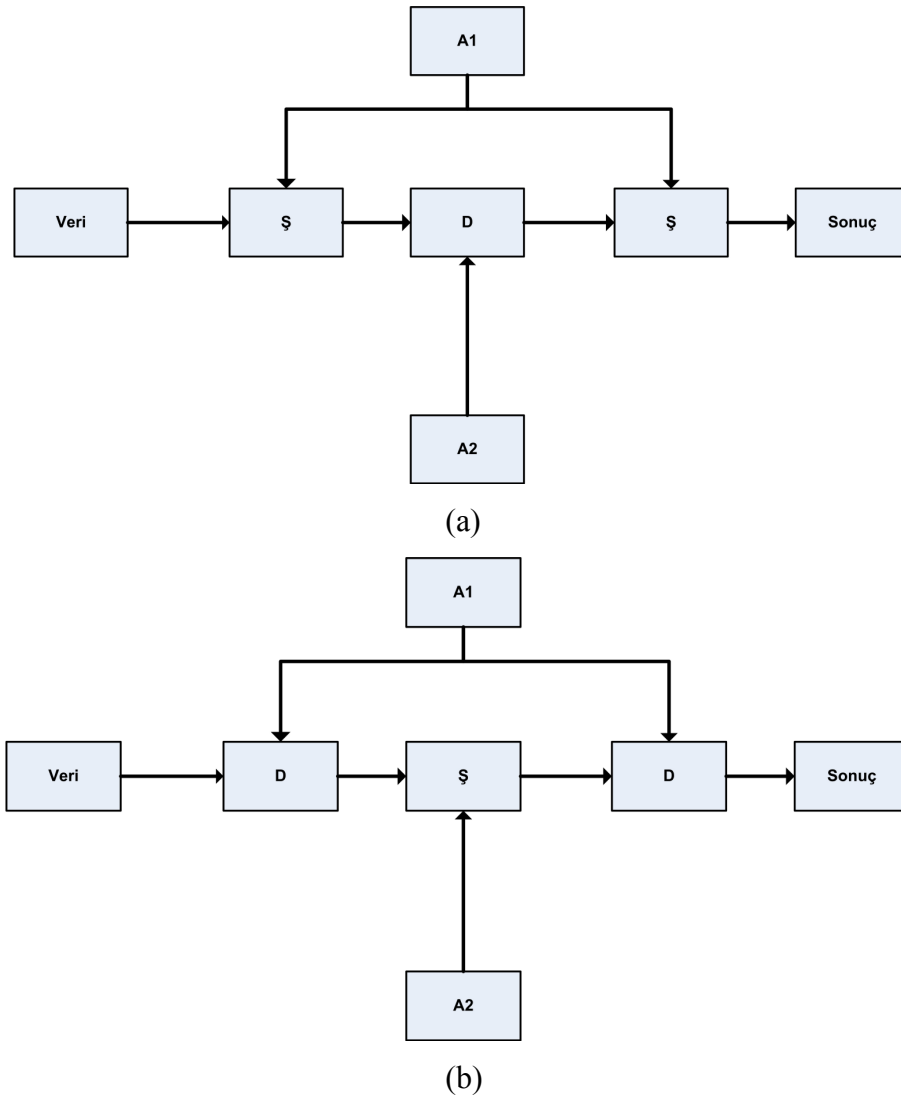
Simetrik şifreleme algoritmalarından DES, 3DES ve AES algoritmaları, bu tez çalışması kapsamında uygulandığı için aşağıda verilen alt bölümlerde kısaca anlatılmıştır.

4.1.1. DES

DES 1977 yılında IBM tarafından geliştirilmiş bir algoritma olup, NIST tarafından ticari ve diğer bazı uygulamalarda kullanılmak üzere yayınlanmıştır [51]. 56 bitlik anahtar kullanarak 64 bitlik veri bloklarını 64 bitlik şifreli veri bloklarına dönüştürmektedir [51]. 64 bitlik veri bloğundan bir başlangıç dizilimi elde edilmekte, 56 bitlik anahtar ise, her bir döngüde kullanılmak üzere 16 adet 48 bitlik anahtarlara dönüştürülmekte ve her döngü içerisinde bu anahtarlar kullanılarak bir sonraki döngüye girdi olmak üzere 64 bitlik çıktılar oluşturulmaktadır [51].

4.1.2. 3DES

DES algoritmasını birden fazla şifreleme ile daha güvenli yapma yöntemi 3DES olarak adlandırılmaktadır [51]. Bu algorithmada 2 anahtar kullanılmaktadır. Mesaj ilk anahtarla şifrelenip, ikinci anahtar ile deşifre edilip tekrar ilk anahtarla şifrelenerek şifreli hale dönüştürülür. Şifreli bir mesaj ise ilk anahtar ile deşifre edilip, ikinci anahtarla şifrelenerek son olarak tekrar ilk anahtarla deşifre edilip orijinal veriye dönüştürülmektedir. 3DES algoritmasının şifreleme ve şifre çözme işlemlerindeki işleyişi Şekil 4.1’de gösterilmektedir [51].



Şekil 4.1. 3DES Şifreleme İşlemleri a) 3DES Şifreleme İşlemi b) 3DES Deşifre İşlemi

4.1.3. AES

DES algoritmasının anahtar uzunluğunun kısa olması, 3DES algoritmasının da yavaş olması yeni bir şifreleme algoritmasını ihtiyacını gündeme getirmiştir [51]. AES algoritması bu ihtiyacı karşılamak üzere Rijndael tarafından geliştirilmiştir. Bu algorithma 128 bitlik veri bloğu, 128, 192 veya 256 bitlik anahtarlarla şifrelenmektedir [51]. Algorithma kullanılacak döngü sayısı S_D

$$S_D = 6 + \max(S_B + S_A) \quad (4.1)$$

şeklinde hesaplanır [51]. Bu formülde S_B veri bloğundaki 32 bitlik kelime sayısı, S_A ise anahtardaki 32 bitlik kelime sayısıdır. AES-256 versiyonu 14 döngüyle çalışmaktadır.

4.2. Özetleme Algoritmaları

Farklı uzunluklardaki mesajları işleyerek sabit uzunlukta veri oluşturma işlemi özetleme olarak isimlendirilir [61]. Özetleme işlemi bilgi güvenliğinde bütünlük kontrolünün sağlanması için kullanılır. Örneğin uzunlukları farklı her bir kelimeden, bir anahtar yardımıyla 128 bitlik farklı özetler üretilebilir. Özetleme algoritmalarında kelimelerin birbirine benzerliği veya uzunluklarının aynı olması oluşturulacak özetlerin benzerliğini neden olmamalıdır. Ayrıca bir özet, birden fazla kelime için üretilmemelidir. Literatürde çeşitli özetleme algoritmaları olmakla birlikte, 6.bölümdeki çalışmanın kapsamında olduğu için sadece SHA özetleme algoritmasına değinilecektir.

SHA özetleme algoritması NIST tarafından yayınlanmıştır. SHA-1 en fazla 2^{64} bit uzunluğundaki kelimelerden 160 bit uzunluğunda özetler oluşturmaktadır [51]. SHA özetleme yöntemi biraz yavaş olsa da daha güvenliği olduğundan kullanılması tavsiye edilmektedir. Daha önceki bazı özetleme algoritmalarında, farklı kelimelere aynı özetin oluşturulması gibi güvenlik açıkları tespit edilmiştir. Geliştirilmekte olan SHA-2 algoritmasında 256 bitlik özet oluşturma hedeflenmektedir [51].

4.3. Steganografi & Kriptografi

Steganografi ile kriptografi arasındaki temel fark kriptografide verinin aslına ulaşılmasa bile gizli bir iletişim olduğunun belli olmasıdır. Kriptografide amaç *verinin gizliliğinin* sağlanması iken steganografide amaç *iletişimin gizliliğinin* sağlanmasıdır. Bu iki bilim dalı her ne kadar farklı hedefleri amaçlasa da güvenli bir iletişim için birbirinin alternatifi değil tamamlayıcısıdır. Herhangi iki nokta arasında gerçekleştirilecek bir mesajlaşmada verinin önce şifrenmesi daha sonra ise şifreli verinin bir resim veya ses dosyası içerisine, şeffaflık bozulmadan saklanıp bu resmin veya sesin gönderilmesi sadece şifreleme veya sadece veri saklama yöntemine göre daha güvenli bir iletişim sağlayacaktır.

Steganografi ile kriptografinin farklı tanımlara işaret etmesine paralel olarak, sistemin kırılması konusu da bu iki bilim dalı için farklı anlamlara gelmektedir. Kriptografi açısından saldırgan gizli mesajı okuduğunda sistem kırılmış iken, steganografide saldırgan gizli iletişimin varlığını tespit ettiğinde sistem kırılmış demektir [4]. Bu iki yöntemin birlikte kullanıldığı bir uygulamada ise sistemin kırılması için önce gizli iletişimin varlığının tespit edilmesi, sonra gizli verinin geri getirilmesi ve nihayetinde şifreli verinin deşifre edilmesi gerekmektedir ki, bu durumda sistemin kırılması çok zor hale gelecektir.

5. SES İÇİNE VERİ SAKLAMA UYGULAMASI

Bu çalışmada steganografik yöntemlerden olan ses içerisine veri saklama yöntemini kullanan DIS (Data Into Sound) yazılımı geliştirilmiştir. Taşıyıcı ses dosyası olarak PCM (Pulse Code Modulation) ses dosyası biçimindeki Microsoft WAV dosyaları kullanılmıştır. Bu bölümde yer alan alt başlıklarda, PCM, WAV dosya yapısı, LSB yöntemi, geliştirilen uygulamanın modülleri, saklama işlemi ve detayları, farklı bit seviyelerinde veri saklama analizleri ve histogram analizleri sunulmuştur.

5.1. PCM

Ses sinyalleri analog sinyallerdir. Sesi sayısallaştırma işlemi analog sinyalleri örneklendirerek PCM gibi bir yöntemle nümerik değerlere dönüştürülerek gerçekleştirilir [52]. Şekil 5.1’de ses sinyallerin sayısallaştırılmasına ilişkin bir gösterim verilmiştir.



Şekil 5.1. Sesin sayısallaştırılması [52]

PCM gerçek ses sinyalini tahminen en uygun nümerik değere dönüştürmektedir [52]. Dolayısıyla PCM ile elde edilen ses örnekleri bir ses dalgasının belli bir andaki yaklaşık değerini ifade etmekte ve bu durum PCM ile sayısallaştırılmış ses örnekleriyle veri saklama amacıyla oynama imkanı sunmaktadır.

5.2. WAV Dosya Yapısı

Yüksek kaliteli ses örneklerini temsil eden en popüler format, 16 bitlik Windows Audio-Visual (WAV) dosya formatıdır [10]. Bir WAV dosyası, diğer dosya türlerinde de olduğu gibi bir başlık bölümüyle, bunu takip eden ve içerisinde iki adet alt veri parçası barındıran bir “WAVE” veri parçasından oluşmaktadır. Bu veri parçasının içindeki alt veri parçalarından birincisi “fmt” veri parçası olup, WAV dosyasının veri biçimiyle ilgili bilgileri içermektedir. Diğer alt veri parçası olan “data” ise asıl ses örnekleri verisini içermektedir. WAV dosyasının veri bölümü ses örneklerinden (audio samples) oluşmaktadır. Bir WAV dosyasının ses örnekleri 8 veya 16 bit uzunluğunda olmaktadır [24]. Yani bir ses örneği bir veya iki byte olabilmektedir.

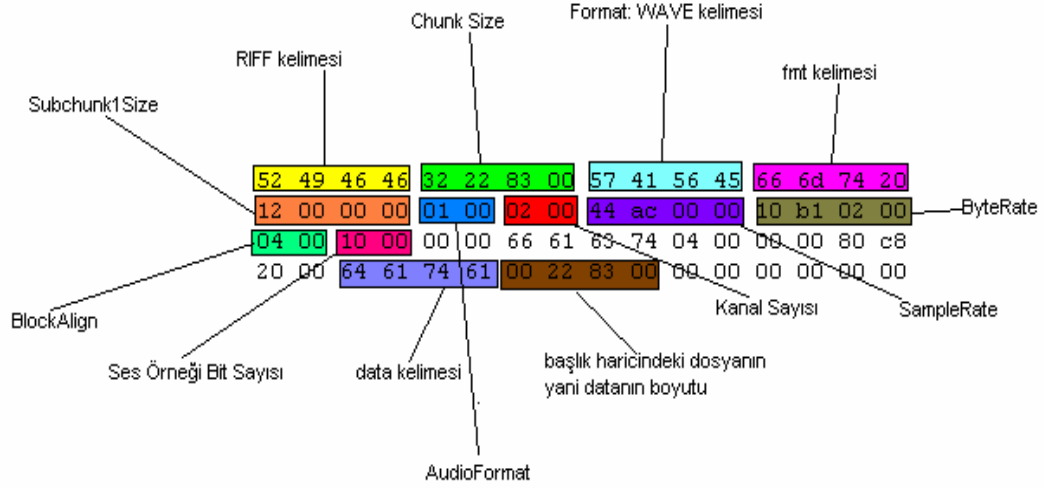
Bölüm No
Bölüm Boyutu
Dosya Biçimi
Alt Bölüm1 No
Alt Bölüm1 Boyutu
Ses Biçimi
Kanal Sayısı
Ses Örneği Oranı
Byte Oranı
Blok Sırası
Örnek Bit Sayısı
Alt Bölüm2 No
Alt Bölüm2 Boyutu
VERİ

Şekil 5.2. WAV dosya formatı [23]

Ses örnekleri, doğası gereği, bir ses dalgasının belirli bir andaki kesin olmayan yaklaşık değerini ifade etmektedir [24]. Bu durumun bize, bu ses örneklerinin en az önemli bitleriyle oynama imkanı verdiğini belirtebiliriz. Şekil 5.2’de WAV dosya formatı gösterilmiştir. Bu şekilde gösterilen veri alanlarından uygulama açısından önemli olanları açıklayalım:

Bölüm No (ChunkID) kısmında “RIFF” kelimesi yazılıdır. RIFF WAV dosyalarını da içeren genel dosya formatının ismidir. Bölüm Boyutu (ChunkSize) alanındaki değer, kendisinden sonra gelen tüm bölümlerin toplam boyutunu byte cinsinden belirtmektedir. Dosya Biçimi (Format) kısmında “WAVE” kelimesi yazılıdır. Alt Bölüm1 No (Subchunk1ID) kısmında “fmt” yazılıdır ve format ile ilgili bilgiler başlamaktadır. Alt Bölüm1 Boyutu (Subchunk1Size) değeri ise, kendisiyle Alt Bölüm2 No (Subchunk2ID) alanı arasında kalan bölümlerin toplam boyutunun ne kadar byte olduğunu göstermektedir. Ses biçimi (AudioFormat) WAV dosyasının biçimini gösterir. PCM için bu değer 1 olmalıdır. Bunun dışındaki değerler sıkıştırma olduğu anlamına gelmektedir. Kanal Sayısı (NumChannels) değeri kaç kanal olduğunu gösterir. Mono için 1, Stereo için 2 değerini alır. Örnek Bit Sayısı (BitsPerSample) verisi bir ses örneğinin kaç bitlik bir değere sahip olduğunu göstermektedir. WAV dosyaları için bu değer 8 veya 16 olduğunu daha önce belirttiğimizi hatırlatmak isteriz. Alt Bölüm2 No (Subchunk2ID) kısmında “data” yazmaktadır. Alt Bölüm2 Boyutu (Subchunk2Size) kısmında başlık kısmının haricindeki asıl verinin büyüklüğünü vermektedir. Şekil 5.3 WAVE formatındaki bir ses dosyasının başlık kısmındaki veriyi göstermektedir.

Şekil 5.3’de görüldüğü gibi, WAV dosyası en başta ASCII formatında RIFF kelimesini içermektedir. Daha sonraki bölüm, yani ChunkSize değeri, kendisinden sonra arta kalan tüm dosyanın boyutunu göstermektedir. Yani tüm dosya boyutundan, RIFF için ve kendisi için ayrılan 8 byte uzunluğundaki kısım haricindeki boyutu ifade eder.



Şekil 5.3. Örnek WAV dosya başlığı

Şekil 5.3’de verilen örnek dosya için ChunkSize bölümündeki değer:

$$256^0 * (1 * 2 + 16 * 3) + 256^1 * (1 * 2 + 16 * 2) + 256^2 * (1 * 3 + 16 * 8) + 256^3 * (1 * 0 + 16 * 0) = 8593970 \quad (5.1)$$

byte büyüklüğündedir. Buna 8 byte daha eklendiğinde tüm dosya boyutunu yani 8593978 byte değerini elde ederiz. Ses dosyasının özellikler ekranına (Şekil 5.4) bakarak bunu doğrulayabiliriz:



Şekil 5.4. WAV dosyası özellikler ekranı

Şekil 5.4’den de görülebileceği gibi dosyanın boyutu, başlık kısmındaki ChunkSize değerinden yola çıkarak hesapladığımız değeri göstermektedir.

Bunun haricinde ses örneği (audio sample) bit sayısı bölümündeki değer:

$$256^0*(1*0 + 16*1) + 256^1*(1*0 + 16*0) = 16 \quad (5.2)$$

Örnek WAV dosyası 2 byte büyüklüğündeki ses örneklerinden oluşmaktadır. Bu değerın bilinmesi saklama işleminde önemlidir. Saklama işlemi yapılırken, taşıyıcı ses dosyasının başlık kısmındaki tüm bölümler okunmakta ve herhangi bir problem olduğu anlaşıldığında saklama işlemi iptal edilmektedir.

5.3. Ses Uygulaması için LSB Yöntemi

Bu uygulamada veriyi saklamak için, veriye ait bit değerleri taşıyıcı WAV dosyasının veri bölümündeki ses örneklerinin en son bitlerine dağıtılmaktadır. Dağıtmaktan kasıt, taşıyıcı ses dosyasının belirlenen ses örneklerinin en az önemli bitlerinde (LSB), saklanacak olan bit değerlerini barındırmaktır. Saklama işlemi ise, barındırıcı bit değerini saklanacak olan bit değerine eşitlemek anlamına gelmektedir. Örneğin, veri kısmı aşağıdaki 8 byte değeriyle başlayan bir WAV dosyası olduğunu varsayalım: 11000110 – 11101001 – 01001101 – 10010011 – 11110100 – 00110010 – 01101110 – 11100001.... Yine varsayalım ki bir byte uzunluğundaki 11100111 verisi saklanacaktır. Saklama işleminden sonra, ses dosyasının veri kısmındaki ilk 8 byte lık uzunluğun yeni değeri; 11000111 – 11101001 – 01001101 – 10010010 – 11110100 – 00110010 – 01101111 – 11100001 şeklinde olacaktır. Altı çizili sayılar saklama sonucu değişen bit değerlerini göstermektedir. Bu yeni haldeki byte değerlerinin son bitlerini yan yana sıraladığımızda sakladığımız veriyi elde ettiğimizi göreceğiz.

5.4. Programın Modülleri

Uygulama Visual Studio.Net ortamında C# programlama diliyle kodlanmıştır. Uygulamada, yukarıdaki formattaki bir WAV dosyası taşıyıcı dosya olarak alınmakta ve içine herhangi bir tür kısıtlaması olmaksızın bir yada birkaç dosya saklanmaktadır. Burada dosyaların türünün ne olacağı ve kaç dosya saklanacağı

konusunda bir kısıtlama yoktur. İstenilen sayıda her türlü dosya saklanabilmektedir. Ancak saklanacak olan dosyaların toplam boyutu taşıyıcı dosyanın saklama kapasitesini aşamaz. Taşıyıcı dosyanın başlık kısmındaki veriler aynen oluşacak olan stego dosyaya kopyalanmaktadır. Yani bu bölümde herhangi bir bozulma meydana gelmemelidir. Saklanacak olan veri, WAV dosyasının veri kısmına saklanmaktadır.

Programda WAV dosyasının veri kısmının byte değerlerinin son bitlerine veri saklanmaktadır. Dolayısıyla saklanacak olan dosyaların toplam boyutu taşıyıcı dosyanın boyutunun 1/16'sını aşamaz. Saklama sonrasında taşıyıcı dosyanın boyutu değişmemektedir. Sadece veri kısmının bazı son bitleri değişmektedir. Yapılan bazı testlerde son bitlerdeki değişim oranının %50 civarında gezinen değerler aldığı görülmüştür. Saklama işleminde taşıyıcı dosyaya ait bir bit değerinin değişme ihtimali %50 olduğundan, bu istatistiksel olarak beklenen bir sonuçtur.

Burada öncelikle saklanacak olan verilerin toplam uzunluğu hesaplanmaktadır. Saklanacak olan veriler bir bütün haline getirilmekte ve ayrıca çıkarma işleminde kullanılmak üzere bir başlık bilgisi de oluşturularak en başa eklenmektedir. Başlık kısmında kaç dosya saklandığı, her bir dosyanın türlerinin ne olduğu, her bir dosyanın uzunluk değerleri ve başlık kısmının uzunluk değeri gibi bilgiler tutulmaktadır. Başlık kısmında dosya uzantılarının her bir karakterinin sayısal değerleri sırayla yazılmakta ve böylece uzantısı ne olursa olsun her türlü veri saklanabilmektedir. Eğer taşıyıcı dosyada yeteri kadar yer yoksa program uyarı mesajı vererek çıkmaktadır. Eğer yeteri kadar yer varsa, veri dosyası mümkün olduğunca yayılmaya çalışılıyor. Yani her byte değerinin son biti yerine bazı byte değerlerini atlayarak gitme imkanı varsa bu şekilde saklanıyor.

Yazılımda iki temel sınıf (class) kullanılmıştır. Bunlardan ilki *WaveStream* sınıfıdır. Bu sınıf WAV dosyasını almakta, başlık bilgileri çıkarmakta ve oluşacak olan stego dosyaya kopyalama gibi işlemleri gerçekleştirmektedir.

Yazılımda kullanılan ikinci temel sınıf *SesToSes* sınıfıdır. Bu sınıfın üç temel fonksiyonu bulunmaktadır:

- Sakla(..)
- Cıkart(..)
- YerlestirmeAraligi(..)

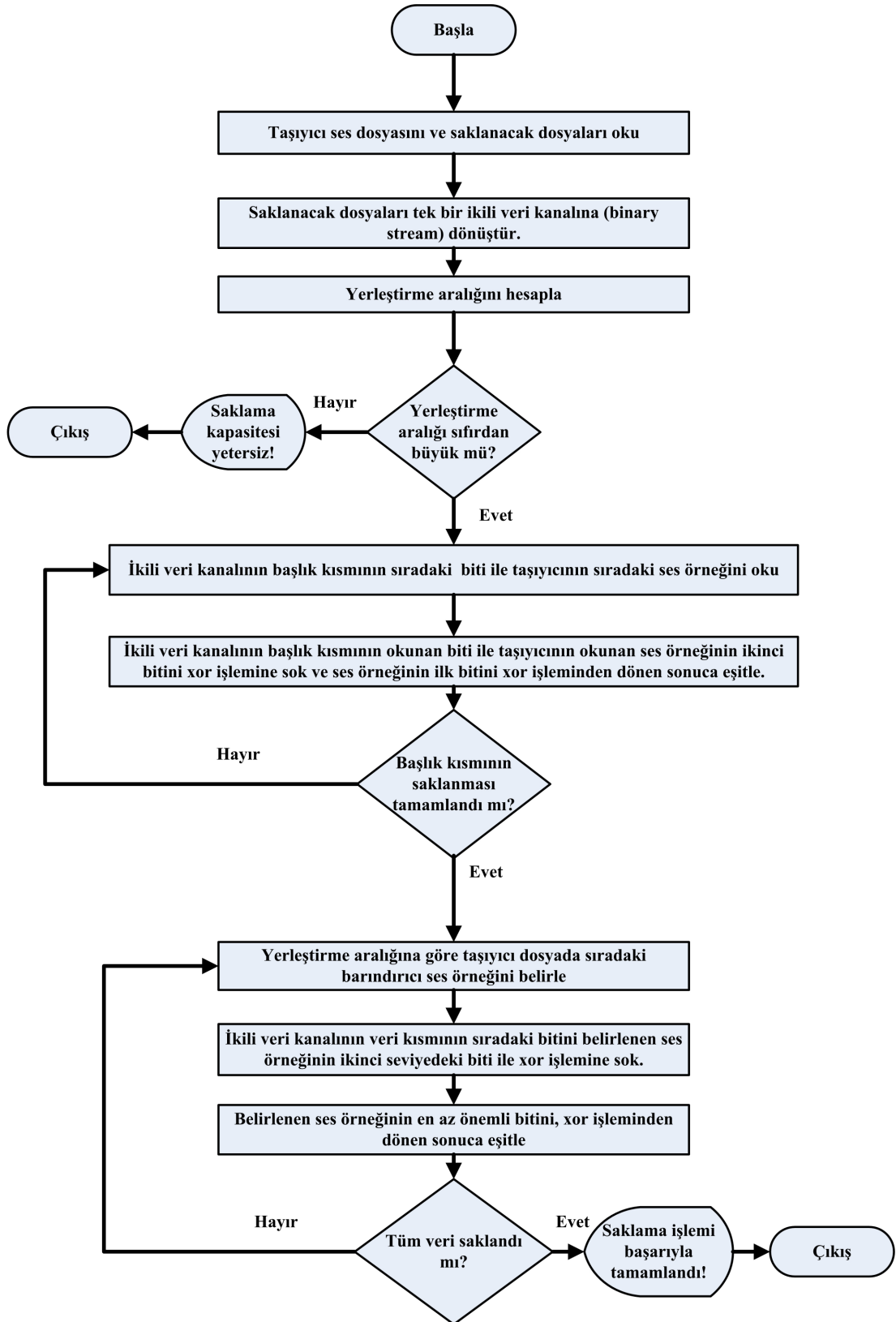
Sakla(..) fonksiyonu wav dosyası içerisine veri saklanması işlemini gerçekleştiren fonksiyondur. *Cıkart(..)* fonksiyonu ise *Sakla(..)* fonksiyonunun tam tersini yani içerisine veri saklanmış stego WAV dosyasının içerisinden saklı veriyi geri getiren fonksiyondur. *YerlestirmeAraligi(..)* ise saklanacak olan dosyanın kaç byte atlanarak saklanabileceğini dönmektedir. Bu fonksiyon 1 değerini dönerse her byte değerinin son bitine veri yazılacak demektir. 2 değerini dönerse bir byte değerinin sonuna yazılırken bir sonrakine yazılmayacak demektir. Eğer 0 değerini dönerse veriyi saklayacak yer yok demektir.

5.5. Saklama İşlemi ve Detayları

Bu uygulamadaki saklama işlemi, saklanacak bitleri mümkün olduğunca aralıklı bir şekilde dağıtarak saklama mantığına dayanmaktadır. Saklama işlemi üç temel aşamadan oluşmaktadır:

İlk aşamada, saklanacak olan tüm veri tek bir ikili (binary) veri bütünü haline getirilir. Saklanacak olan tüm dosyaların içeriğiyle, bunlara ve saklama parametrelerine ait meta bilgi, tek bir ikili (binary) veri bütünü haline getirilir. Bu verinin başlık kısmında yer alan meta bilgide, başlık kısmının uzunluğu, saklanan dosya sayısı, bu dosyaların boyutları, isimleri ve uzantıları gibi, saklanan verileri geri getirme işleminde ihtiyaç duyulacak bilgiler yer almaktadır.

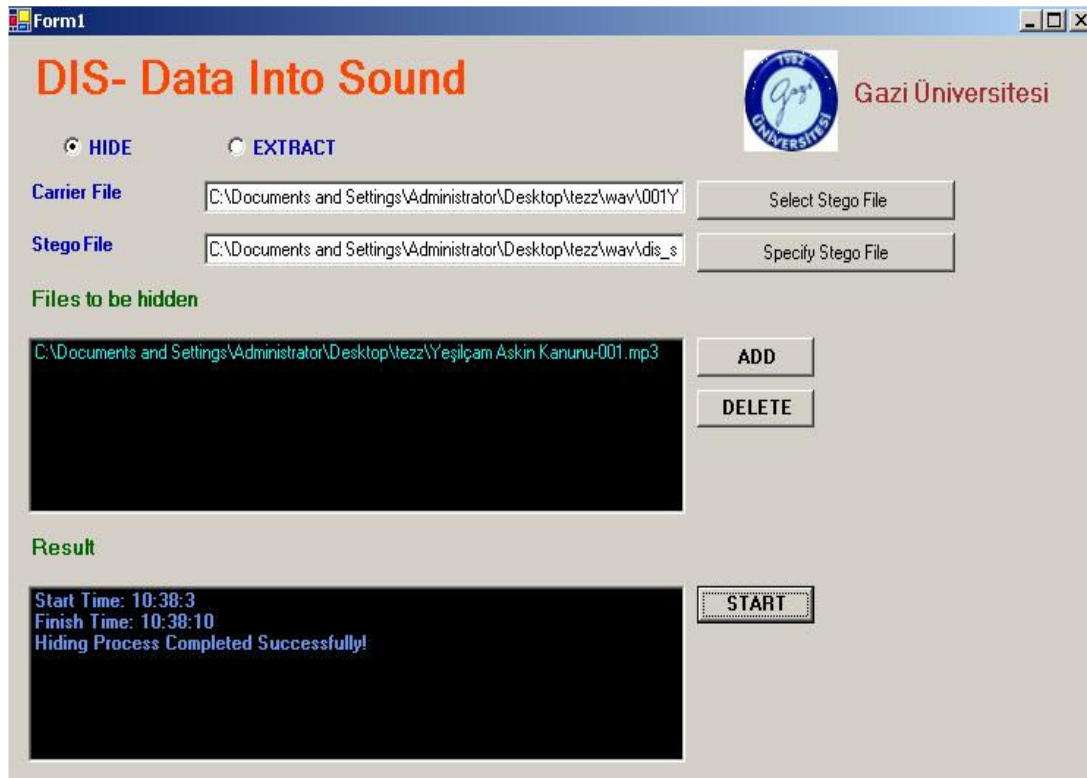
Saklama işlemine ilişkin akış diyagramı Şekil 5.5’de verilmiştir:



Şekil 5.5. DIS yazılımı saklama işlemi akış diyagramı

İkinci aşamada taşıyıcı ses dosyasının saklama aralığı belirlenir. Yani bir sonraki biti yerleştirmek için ne kadar ses örneğinin atlanacağı hesaplanarak yerleştirme aralığı dediğimiz değer hesaplanır. Bu değer, bir sonraki bitin saklanacağı barındırıcı ses örneğinin pozisyonunu belirlemek için şu anda bulunulan konumdan sonra ne kadar ses örneğinin atlanacağını belirtmektedir. Bu konuyla ilgili ayrıntılı açıklamalar sonraki kısımlarda yer alacaktır.

Son aşamada ise saklanacak olan veriye ait bitler sırayla, belirlenen pozisyonlardaki ses örneklerinin en az önemli bitlerine yerleştirilir. Saklanacak olan bit değeri, barındırıcı ses örneğinin ikinci seviyesindeki bit değeri ile XOR fonksiyonuna sokulur ve elde edilen değer o ses örneğinin en az önemli bitine yani birinci seviyedeki bitine saklanır. Saklamadan kasıt, birinci seviyedeki bit değerinin, XOR işlemi sonucu oluşan değere eşitlenmesidir. XOR operasyonu ise bir derece daha güvenlik katmak amacıyla yapılmaktadır. Geliştirilen yazılımın saklama işlemine ilişkin ara yüzü Şekil 5.6’da verilmiştir.



Şekil 5.6. DIS yazılımı saklama ekranı

Şekil 5.6’da verilen ekranda bir WAV dosyasının içerisine mp3 biçimindeki başka bir ses dosyası saklanmıştır. Oluşan stego dosyanın, orijinal ses dosyasından boyut olarak hiçbir farkı bulunmamaktadır. Saklama işleminde değişen tek şey, ses örneklerinin en az öneme sahip bitlerinden bazılarının değerleridir. Bu değişimi, WAV dosyalarının içeriğini on altılı sayı biçiminde görüntüleyen bir yazılım vasıtasıyla (HHD Free Hex Editor yazılımı gibi) taşıyıcı ses dosyası ile stego ses dosyasının saklama öncesi ve saklama sonrası ses örnekleri Şekil 5.7’de gösterilmektedir. Kutucuk içerisine alınan birkaç değişim örneği bu şekilde verilmiştir.

7e f5 1a f3	7f f5 1b f3
5c f6 77 f3	5c f6 76 f3
2f fe e2 00	2f fe e3 00
97 21 e1 1d	97 21 e1 1d
0b 52 de 41	0a 52 de 41
bd 5f 3e 52	bd 5f 3f 52
ba 49 31 42	bb 49 30 42
de 31 5a 2f	df 31 5b 2f
f8 34 51 2e	f8 34 51 2e
be 25 82 24	bf 25 82 24
98 1d 6c 13	99 1d 6d 13
78 fd df ed	78 fd df ed

(a)

(b)

Şekil 5.7. Ses dosyalarına ait on altılı değerler a) Taşıyıcı ses dosyası b) Stego ses dosyası

Şekil 5.7’yi incelediğimizde, stego ses dosyasındaki bazı ses örneklerindeki en az önemli bitlerin değiştiğini görürüz. Örneğin taşıyıcı ses dosyasına ait şekildeki, 2. satırın 3. değeri 77 iken stego ses dosyasında aynı pozisyondaki değer 76 olmuş yani 1 azalmıştır. Buna karşılık, yine taşıyıcı ses dosyasına ait şekildeki 11. satırın 3. değeri 6c iken stego ses dosyasına ait şekilde karşılık gelen pozisyondaki değer 6d olmuş yani 1 artmıştır. Program sadece en az öneme sahip bitlerde değişiklik yaptığı için, değişime uğrayan ses örneklerinin değerinin sadece 1 artması veya sadece 1 azalması beklenen bir sonuçtur.

5.5.1. Saklama kapasitesi ve saklama aralığı

Bu uygulamada, taşıyıcı ses dosyasının saklama kapasitesini aşmadığı müddetçe, sayı sınırlaması olmaksızın her türlü dosya ses içerisine saklanabilmektedir. Saklama işlemi için ses dosyasının, yani WAV biçimindeki dosyanın veri bölümündeki ses örneklerinin en az öneme sahip birinci seviyedeki bitleri (LSB) kullanılmaktadır. Veriyi saklarken taşıyıcı ses dosyasındaki barındırıcı (host) ses örneklerinin pozisyonu bir anahtara göre belirlenmemekte, dolayısıyla rastgele yerine sabit aralıklarla ardışık saklama yapılmaktadır. Bunun temel mantığı da, saklanacak olan verinin bitlerini mümkün olduğunca aralıklı saklama düşüncesine dayanmakta, böylelikle steganaliz ataklarına sağlam olması beklenmektedir. Saklama kapasitesine ait sınır değerlerin ve saklama aralığının hesaplanması için aşağıdaki tanımlamaları yapalım:

b_s = Ses örneği bit sayısı (WAV için 8 veya 16)

L = WAV dosyasının veri bölümünün boyutu (Başlık kısmındaki Subchunk2Size değeri).

Bu durumda taşıyıcı ses dosyasına saklanabilecek maksimum veri boyutu L/b_s byte olmaktadır. Saklama işlemi öncesinde, yukarıdaki akış şemasından da görüldüğü üzere, saklanacak dosyalara ait tüm veri, bir başlık kısmı da eklenerek tek bir ikili veri kanalı (stream) haline dönüştürülmektedir. Bu başlık kısmında, çıkartma işleminde de ihtiyaç duyulacak olan saklama işlemine özgü çeşitli parametreler tutulmaktadır. Bunlar, başlık kısmının uzunluğu, saklanan dosya sayısı, saklanan dosyaların isimleri ve uzunlukları gibi değişkenlerdir.

Saklanacak olan verinin sıradaki bitini barındıracak olan ses örneğinin pozisyonunu belirlemek için saklama aralığı değeri kullanılmaktadır. Saklama aralığının değeri:

$$i = \lceil (8*L/b_s) / (L_c*8) \rceil = \lceil L/b_s*L_c \rceil \quad (5.3)$$

şeklinde hesaplanır. Bu eşitlikte;

i = saklama aralığı

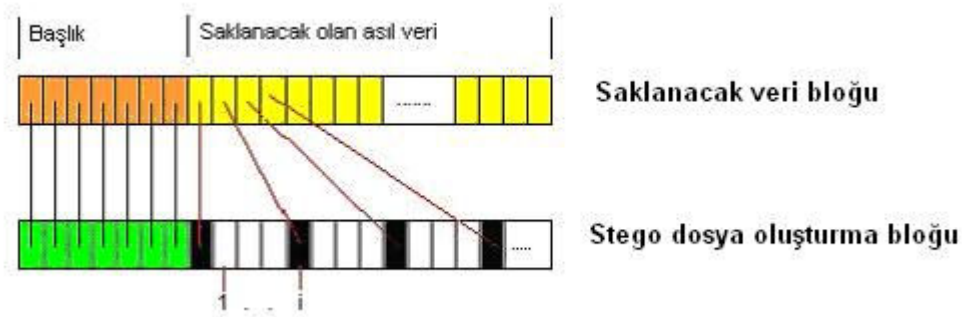
L_c = saklanacak olan verinin byte cinsinden boyutu

L = WAV dosyasının veri bölümünün boyutu

b_s = Ses örneği bit sayısı (WAV için 8 veya 16) olarak tanımlanmaktadır.

Burada tam değer fonksiyonu, içerisindeki değer in ondalık kısmının atılmış halini ifade etmektedir.

Hesaplanan bu i değerinin en küçük değeri 1 olmaktadır. Bu durumda ses örneklerine ait en az öneme sahip bitlerin, yani birinci seviyedeki bitlerin tamamı saklama işleminde kullanılacak demektir. Saklanacak olan veriye ait başlık kısmı, i değeri kullanılmaksızın, ilk ardışık ses örneklerinin en az öneme sahip bitlerine saklanmaktadır. Daha sonra asıl veri saklanırken bir sonraki bitin nereye saklanacağını belirlemek için i değeri kullanılmaktadır. Bu durum Şekil 5.8’de gösterilmektedir:



Şekil 5.8. DIS yazılımı saklama işlemi

Şekil 5.8’den ve Eş. 5.3’ de verilen hesaplamalardan da anlaşılacağı gibi, eldeki asıl veri saklanırken belli bir pozisyondan başlanıp ardışık saklama yapılmamaktadır. Bunun yerine verinin mümkün olduğunca saçılması esas alınıp, barındırıcı bitlerin arasının mümkün olduğunca uzak olması sağlanmaktadır. Bir WAV dosyasının ses örneklerinin tamamının en az öneme sahip birinci seviyedeki bitleri değişse bile, orijinal dosya ile oluşan dosya arasında işitsel olarak bir fark bulunmadığı yaptığımız testler sonucu gözlemlenmiştir. Bu testlerde ekran ara yüzünden (Bkz. Şekil 5.9) de görülebileceği gibi ses dosyasına ait ses örneklerinin birinci

seviyesindeki bitlerin tamamının değişeceği şekilde bir ayarlama yapılmıştır. Bu işlem sonucunda oluşan stego ses dosyası, orijinal dosyadan işitsel olarak farksızdır.



Şekil 5.9. DIS yazılımı örnek saklama ekranı

5.5.2. XOR işlemi

Saklanacak olan veri bitlerini direk saklamak yerine başka bir forma dönüştürerek saklamak daha uygun olacaktır. Bu nedenle, saklanacak olan veri bitleri, saklama öncesi XOR operasyonuna sokulmaktadır. XOR operasyonu için doğruluk tablosu Çizelge 5.1’de verilmiştir.

Çizelge 5.1. XOR doğruluk tablosu

A	B	A XOR B
0	0	0
0	1	1
1	0	1
1	1	0

$b_1(n)$ ve $b_2(n)$ değerlerinin n . ses örneğinin birinci ve ikinci derecedeki en az öneme sahip bitleri olduğunu, ve b değerinin ise saklanacak olan sıradaki bit olduğunu varsayalım. Bu durumda, ekleme Eş. 5.4’de verildiği gibi yapılmaktadır:

$$b_1(n) = b_2(n) \text{ XOR } b \quad (5.4)$$

Çıkartma işleminde ise, b değeri tekrar Eş. 5.5 ile bulunmaktadır:

$$b = b_1(n) \text{ XOR } b_2(n) \quad (5.5)$$

5.5.3. Hata oranı

Genel olarak, eğer bir ses örneğine n ($n < 16$ veya $n < 8$) bit saklayacak olursak, n LSB değerinin değişmesi sonucu oluşacak maksimum hata $2^n - 1$ olacaktır. Bu uygulamada sadece birinci seviyedeki en az öneme sahip bitler kullanıldığından $k = 1$ değerini alacak ve bir ses örneğine eklenecek hata değeri en fazla 1 olacaktır. Saklama işleminde ses örneklerine eklenen hata miktarları ve değişime uğrayan ses örneklerinin birbirine yakınlığı gibi konular oluşan ses dosyasının şeffaflığını etkileyeceğinden, hissedilir bir işitsel fark oluşmaması için bu hata hesaplaması önemlidir. Saklama işleminde hedef ses örneklerinde oluşacak hatanın minimuma indirgenmesinin, stego dosyada işitsel ve istatistiksel olarak şeffaflığın bozulması olasılığını azaltacağı açıktır.

5.6. Değişik Seviyelerde Saklama İşleminin Analiz Edilmesi

Steganografi uygulamalarında oluşan stego dosyanın şeffaflığının sağlanması ilk şart olmakla birlikte, saklama kapasitesinin yüksekliği de önemli bir konudur. Saklama kapasitesi ile şeffaflık arasında bir ters orantı vardır. Yani saklanan veri miktarı ve dolayısıyla saklama işleminde kullanılan bit seviyesi arttıkça şeffaflık azalabilmektedir. Bu uygulama çerçevesinde 16 bitlik ses örneklerinden oluşan PCM biçimindeki bir WAV dosyasının bit seviyelerinin her birine saklama işlemi yapılmış ve işitsel ve grafiksel olarak sonuçlar analiz edilmiştir. Oluşan ses dosyalarını işitsel olarak incelediğimizde ise 10. seviyeden itibaren açık bir şekilde ses kalitesinin bozulduğu anlaşılmaktadır. 9. seviyede çok açık olmasa da bir bozulmadan söz edilebilir ancak 1. seviye ile 8 seviye arasındaki tüm seviyelerde ses kalitesi olarak hissedilebilir bir fark bulunmamaktadır.

Burada k. seviyedeki bir bitin değişmesi sonucu ses örneğinde oluşacak hata 2^k olacaktır. Matematiksel olarak $k > 1$ ve $1 < n < k$ olmak üzere

$$\sum 2^n = 2^k - 2 \quad (5.6)$$

olmaktadır. Örneğin 8. seviyedeki bitin değişmesi sonucu oluşacak değişim $2^8 = 256$ olacaktır. Eğer 8. seviyedeki bit değişmeyip ondan önceki ilk 7 bitin tamamı değişecek olsa toplam değişim

$$2^1 + 2^2 + 2^3 + 2^4 + 2^5 + 2^6 + 2^7 = 254 \quad (5.7)$$

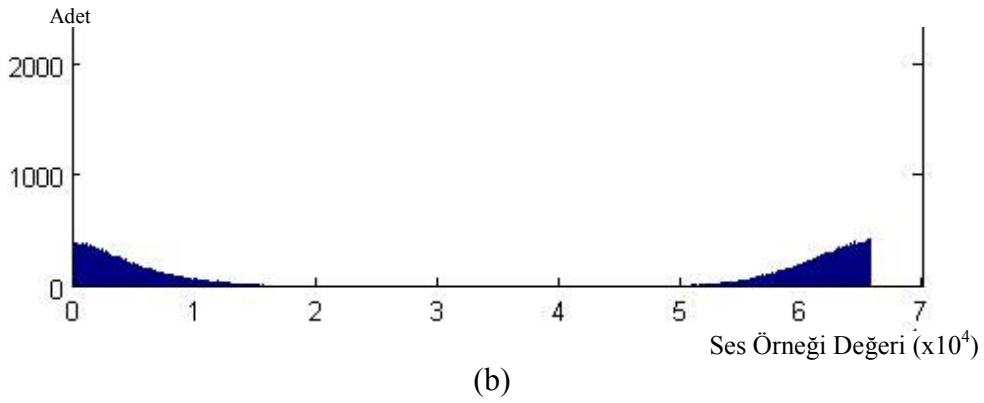
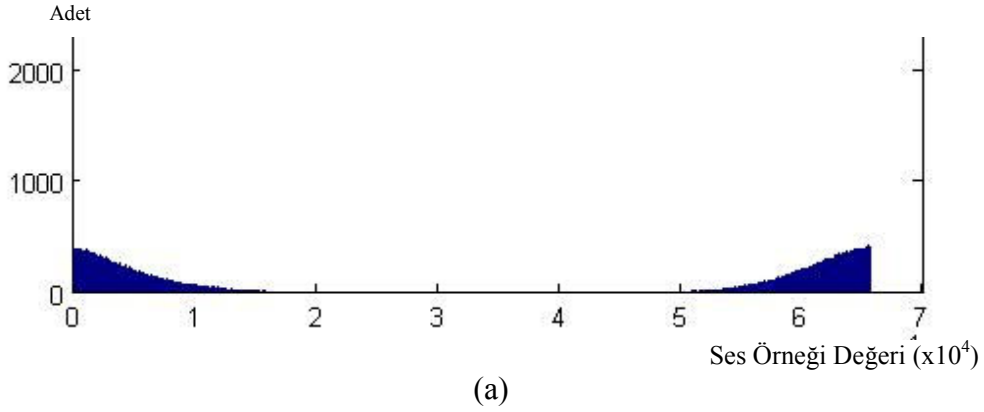
olacaktır. Yaptığımız testlerde sadece 8. seviyede yapılan bir saklama işleminin işitsel olarak hissedilebilir bir fark oluşturmadığını belirtmiştik. Bu durumda ilk 7 seviyedeki bitlerin tamamı kullanılarak yapılacak olan bir saklama işleminin işitsel açıdan teorik olarak hissedilebilir bir fark oluşturmayacağını söyleyebiliriz. Bu durumda 16 bitlik PCM biçimindeki WAV ses dosyaları %50 ye yaklaşan oranlarda, işitsel olarak ayırt edilemeyecek şekilde saklama işleminde kullanılabilecektir.

5.7. Histogram Analizleri

Histogram analizleri steganaliz yöntemlerinden birisi olup en sık kullanılanıdır. Bu yöntemde taşıyıcı dosya ile stego dosyanın ayırt edilmesine çalışılır [22]. 16 bitlik ses örneklerinden oluşan PCM formatında WAV dosyasının tüm bit seviyelerinde veri saklama ve sonuçlarını işitsel olarak analiz etme testi Bölüm 5.6’ da sunulmuştu.

Şekil 5.10’da yer alan şekiller birinci seviyedeki saklama işlemi sonucu oluşan stego dosya ile taşıyıcı dosyanın, Matlab yazılımıyla oluşturulmuş histogram grafiklerini göstermektedir.

2. seviye ve üzeri bit seviyelerinde saklama işlemi gerçekleştirilmiş stego dosyalara ait histogram çizelgeleri EK-1’de verilmiştir.

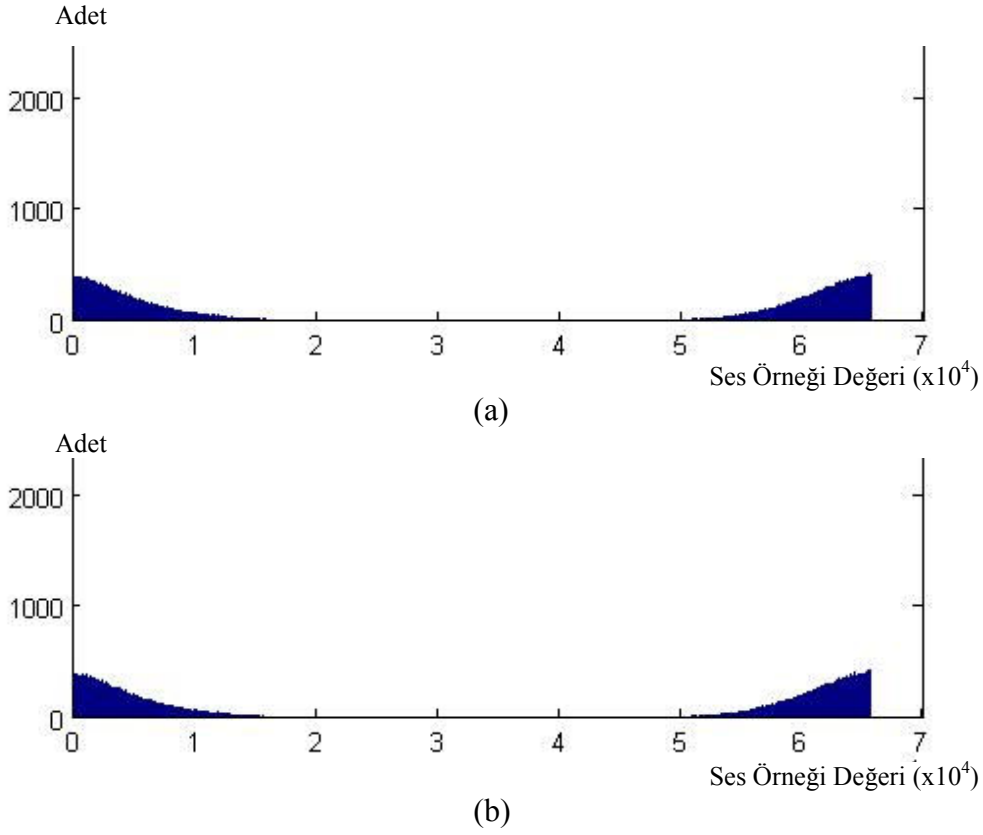


Şekil 5.10. Taşıyıcı dosya ile 1. bit seviyesinde veri saklı dosyaya ait histogram
Çizelgeleri a) Taşıyıcı ses dosyası histogram çizelgesi b) 1. seviyede veri saklanan stego dosyaya ait histogram çizelgesi

Bu çizelgelerden de görülebileceği gibi 9. seviye ve üst seviyelerde yapılan saklama işlemleri ayırt edilebilir farklar oluşturmaktadır.

Şekil 5.11’de, orijinal ses dosyası ile bu dosyanın en az önemli bitlerinin tamamının değiştirildiği dosyaya ait histogram çizelgeleri yer almaktadır. Oluşan yeni dosya işitsel olarak herhangi bir fark içermemekle birlikte, işletim sistemi açısından da herhangi bir farklı fiziksel özelliğe sahip değildir.

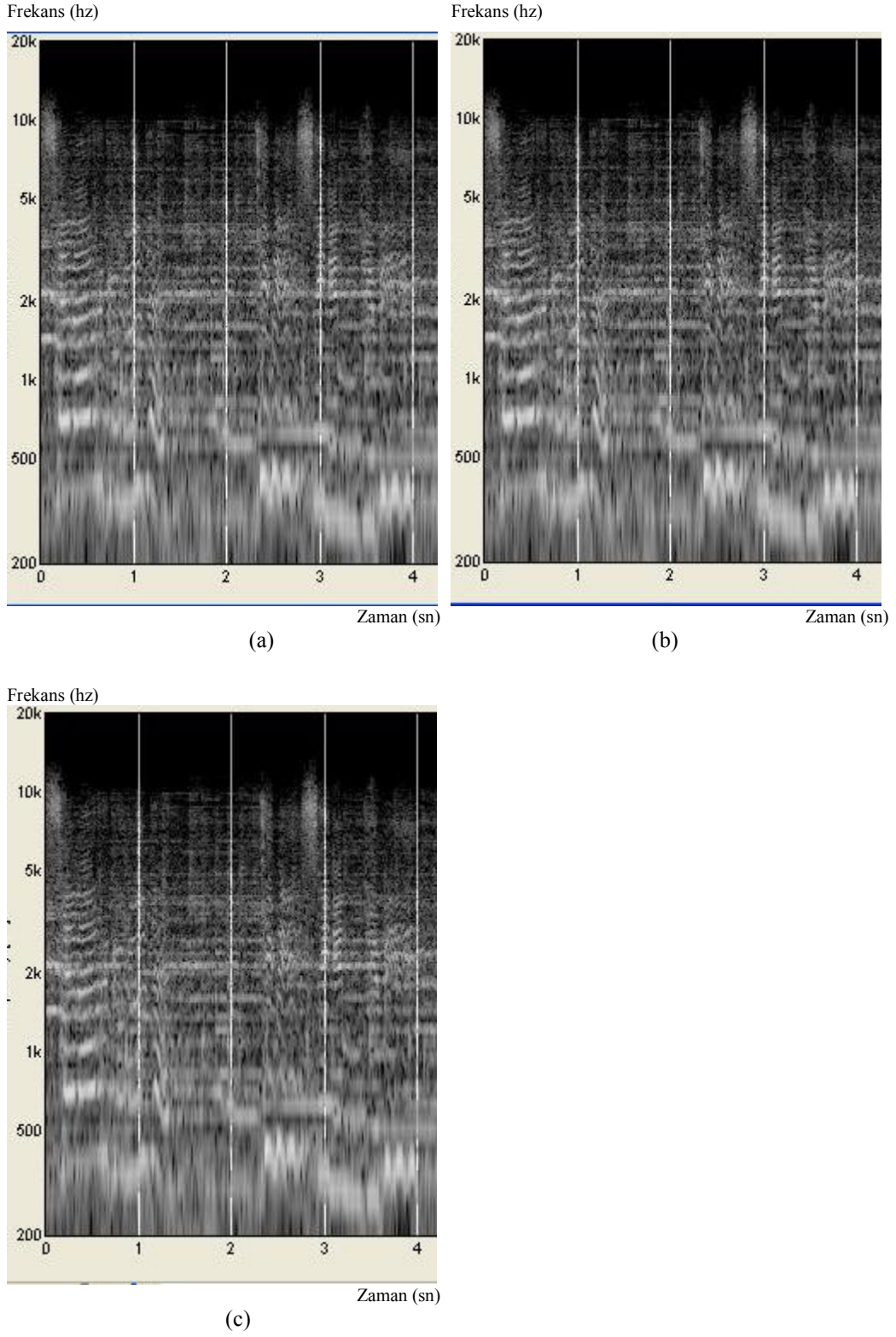
Şekil 5.11’den görüldüğü gibi WAV dosyasının en az öneme sahip bitlerinin tamamının değişmesi bile histogram çizelgesinde belirgin bir fark oluşturmamaktadır.



Şekil 5.11. Taşıyıcı ve stego dosyalara ait histogram çizelgeleri a-) orijinal dosyaya ait histogram çizelgesi b-) 1. seviyedeki tüm bitleri değişmiş stego dosyaya ait histogram çizelgesi

Şekil 5.12’de orijinal ses dosyasının, 1. seviyede veri saklanmış stego ses dosyasının ve 1. seviyedeki bitleri tamamen değişmiş stego dosyanın 4 saniyelik bir bölümlerine ait, Realtime Analyzer yazılımıyla elde edilen spektrogram grafikleri verilmiş olup bu grafiklerde de belirgin bir farklılık gözlenmemektedir.

Daha üst seviyelerdeki saklama işlemleri sonucu oluşan stego ses dosyalarına ait spektrogram grafikleri EK-1’de sunulmuştur. Bütün bu analizler sonucunda PCM formatındaki bir WAV dosyasına LSB modifikasyonu ile yapılacak veri saklama işlemi, tüm bitlerin değişmesine yol açsa bile işitsel olarak bir farklılık oluşturmayacak ve histogram analizine dayalı steganaliz atakları karşısında herhangi bir dayanıklılığın azalması problemine neden olmayacaktır.



Şekil 5.12. Taşıyıcı ve stego dosyalara ait spektrogram grafikleri a) Orijinal dosya b) 1. seviyede veri saklanan stego dosya c) 1. seviyedeki tüm bitleri değişmiş stego dosya

5.8. Çıkartma İşlemi ve Detayları

Çıkartma işlemi, steganografi uygulamalarının doğası gereği saklama işleminin tam tersi aşamalardan oluşmaktadır. Bu uygulamada saklanan veriye ilişkin meta bilgilerin tutulduğu başlık kısmının oluşturulması çıkartma işleminin ilk aşamasıdır. Stego dosyanın ilk bitlerinden itibaren ardışık okunan bitler kullanılarak saklama işleminde kullanılan parametreler sırayla elde edilir. Saklama işlemi bölümünde anlatılan *saklama aralığının* doğru hesaplanması ve saklamaya başlanılan bit pozisyonunun doğru belirlenmesi çıkartma işleminin doğru bir şekilde gerçekleşmesi için mutlaka gerekmektedir. *Saklama aralığı* ve *başlangıç pozisyonu* belirlendiğinde, bu iki değişkene göre ilgili pozisyonlardaki tüm bitler sırayla okunur ve saklanan veri geri getirilir. Başarılı bir steganografi uygulamasının ilk şartı şeffaflığın (transparency) sağlanması diğeri ise saklanan verinin düzgün bir şekilde geri getirmesidir [22]. DIS yazılımı bu iki şartı da sağlamakta, oluşan stego dosyalar işitsel olarak orijinalinden farksız olmakta ve çıkartma modülü yardımıyla ise saklanan veri doğru bir şekilde geri getirilmektedir.

5.9. Sonuç ve Değerlendirmeler

Geliştirilen DIS yazılımıyla yapılan denemeler ve analizler sonucunda aşağıdaki sonuçlar elde edilmiştir.

- Ses içerisine veri gizleme uygulaması yazılımı başarıyla gerçekleştirilmiştir. Geliştirilen yazılım üzerinde farklı denemeler ve analizler yapılmıştır.
- 16 bitlik ses örneklerinden oluşan WAV formatındaki ses dosyaları steganografide veri saklama amacıyla kullanılabilmekte ve yüksek saklama kapasitesi sunabilmektedir.
- 1. seviye ile 8. seviye arasındaki tüm bit seviyelerine yapılan saklama işlemi işitsel olarak fark edilir bir gürültü oluşturmamakta, şeffaf bir şekilde saklama yapılabilmektedir.

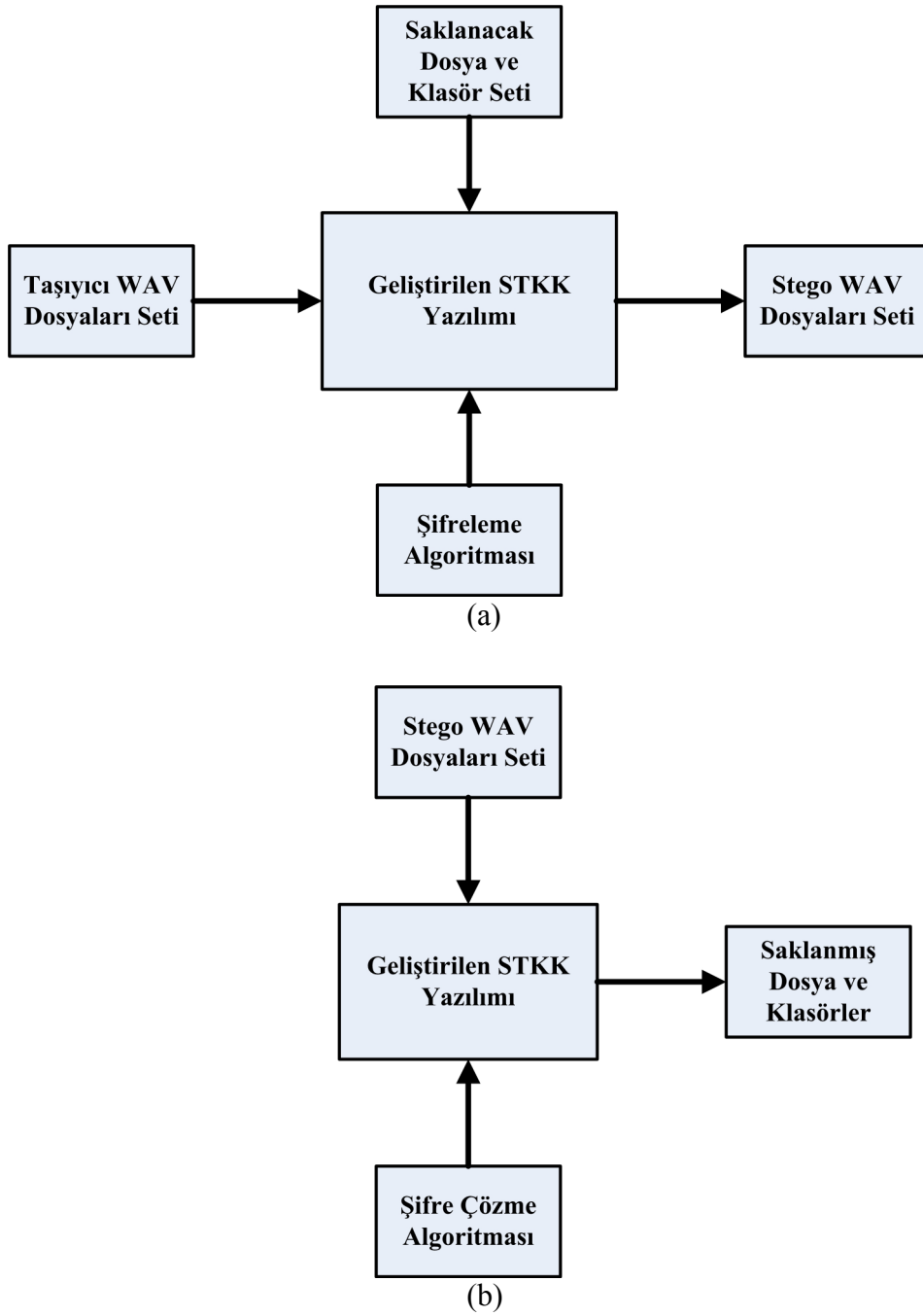
- 9. seviyede yapılan saklama deęişim oranının az olması durumunda işitsel olarak hissedilmemekte ancak bu oran fazla olduęunda oluşan gürültü belirgin hale gelmektedir.
- 10. seviye ile 16. seviye arasındaki yapılan saklama işlemlerinde deęişim oranı çok düşük bile olsa şeffaflık bozulmaktadır.
- Saklama işlemi sonucu şeffaflığın bozulması konusu taşıyıcı ses dosyasının ses kalitesiyle de ilgilidir. Kalitesi iyi olmayan ses dosyalarında yapılan saklamalarda 9. seviyede yapılan saklama işleminde de işitsel olarak hissedilir fark oluşmamaktadır.
- En az öneme sahip bit seviyesinde yani 1. seviyede yapılan saklama işleminde taşıyıcı bitlerin tamamı deęişse bile işitsel olarak, oluşan stego ses dosyasının şeffaflığı bozulmamaktadır.
- Herhangi bir bit seviyesinde yapılan saklama işleminde, saklama için belirlenen pozisyonlardaki bitlerin ortalama %50 sinin deęeri deęişmektedir.
- 8. seviyede şeffaflık bozulmadan saklama yapılabiliyor olması teorik olarak yaklaşık %50 kapasiteyle işitsel fark oluşturmada saklama yapılabileceğini göstermektedir.
- Bir anahtar eşliğinde rastgele yapılmayan saklama işlemi sonucunda oluşan dosyaların istatistiksel ataklara sağlamlığı azalmaktadır.
- Farklı bit seviyelerine bilgi gömülmesi güvenlięi arttıracak gibi steganalizi de zorlaştıracak deęerlendirilmektedir.

6. STEGO TABANLI KLASÖR KİLİTLEME YAZILIMI

Ses içine veri saklama yöntemini kullanarak kişisel bilgi güvenliğinin sağlanmasına yönelik STKK (Stego Tabanlı Klasör Kilitleme) isminde bir klasör kilitleme yazılımı geliştirilmiştir. Klasör kilitleme programları bir klasörü ve içeriğini başkalarına karşı görünmez ve erişilmez hale getiren programlardır ve kendi başına ayrı bir güvenlik yazılımı alanı olarak değerlendirilebilir. Dolayısıyla geliştirilen bu yazılım aslında steganografi, kriptografi ve klasör kilitleme gibi üç farklı güvenlik konusunu bir araya getirme özelliği de taşımaktadır. Bu yazılımlar genelde işletim sistemine ait fonksiyonları kullanarak çalışır ve aslında disk üzerinde bir yerlerde var olan bir klasörü, kullanıcılara karşı görünmez ve arama sonucunda bulunmaz hale getirmektedirler. Geliştirmiş olduğumuz bu yazılım ise, bir klasörün saklanması işini steganografi tekniği kullanarak ve işletim sistemi fonksiyonlarına ihtiyaç duymaksızın gerçekleştirmektedir. STKK kişisel bilgi güvenliğini daha üst seviyede sağlamak ve güncel hayatta bilgi güvenliğinin önemli unsurlarından birisi olan gizlilik unsurunu desteklemek üzere geliştirilmiş bir yazılımdır.

Geliştirilmiş bu yazılımın temel işlevi, belirlenen bir klasörü veya klasörler ve dosyalar setini, tüm içeriğiyle birlikte, yine belirlenen WAV formatındaki ses dosyaları seti içerisine, şifreleme imkanı da sunarak saklamaktır. Sonuçta içerisinde bir klasöre ve içeriğine ait bilgileri barındıran WAV dosyaları oluşmaktadır.

STKK yazılımı stego sistemlerin bir gereği olarak saklı olan verinin tekrar geri getirilmesi işlevine de sahiptir. Saklama sonucu oluşan stego dosyalar ve saklamada kullanılan şifreleme algoritması yardımıyla da verilerin uygun anahtar kullanılarak deşifre edilmesi durumunda STKK yazılımının geri getirme ara yüzü kullanılarak saklanan tüm dosya ve klasörler başarıyla geri getirilebilmektedir. Geliştirilen yazılımın genel olarak yapısı (saklama ve geri getirme işlemleri) Şekil 6.1’de özetlenmiştir. Geliştirilen yazılımla ilgili detaylar alt başlıklarda açıklanmaktadır.



Şekil 6.1. Geliştirilen STKK yazılımıyla gerçekleştirilen işlemlerin şemaları
a) Saklama işlemi b) Geri getirme işlemi

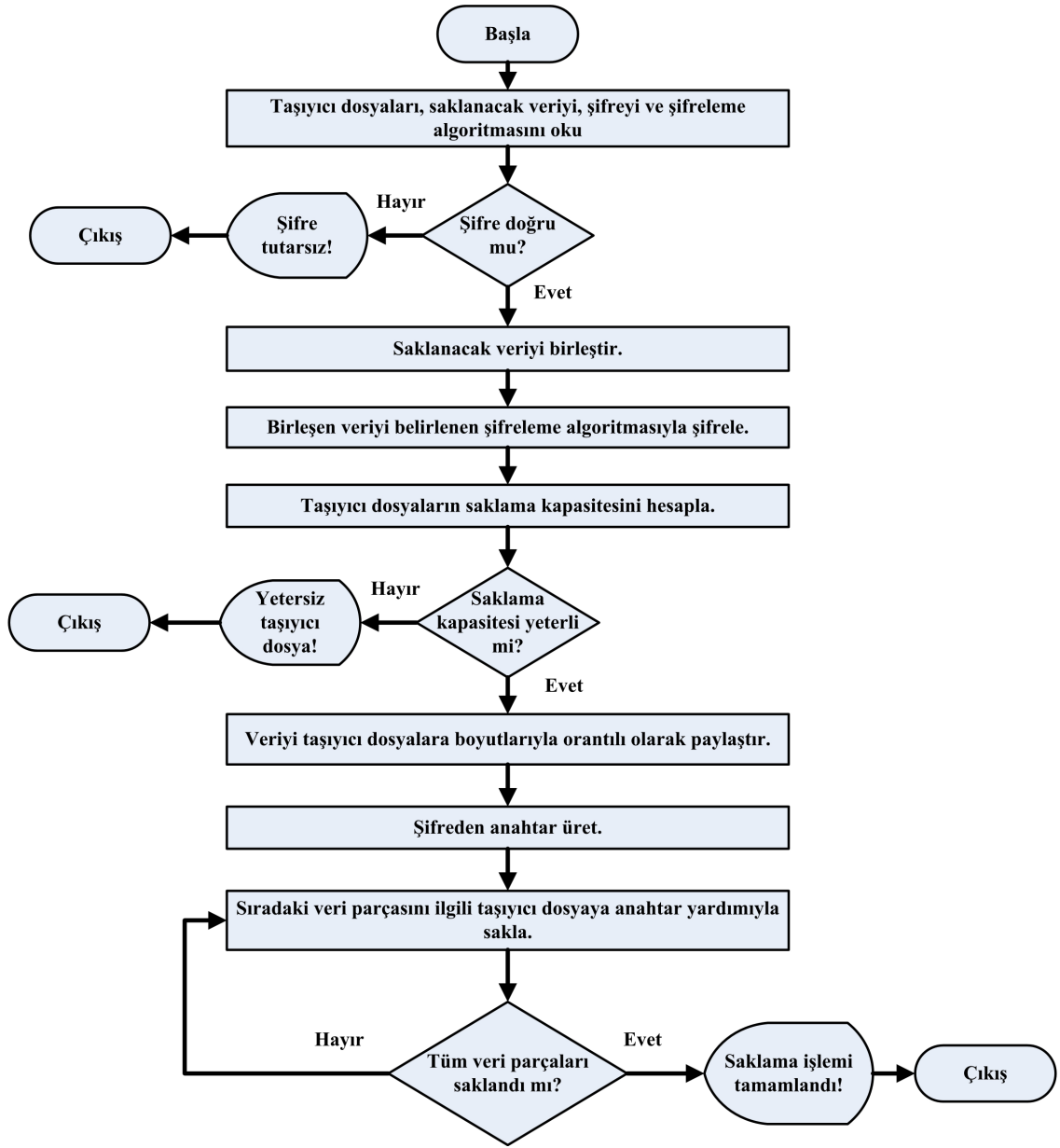
6.1. Saklama İşlemi

Saklama işlemi gerçekleştirilirken öncelikle, saklanacak olan dosyalar ve klasörler belirlenip, liste haline getirilmektedir. Daha sonra ise bunların saklanacağı WAV

dosyaları belirlenmektedir. Bir sonraki aşamada ise kullanıcının şifre belirlemesi ve veri güvenliğini daha da arttırmak için Bölüm 4’de açıklanan şifreleme algoritmaları olan AES, 3DES ve DES algoritmalarından birini seçmesi gerekir. Bu konuda herhangi bir seçim yapılmazsa AES şifreleme algoritması kullanılır. En son aşamada, saklama işlemi sonucunda oluşacak stego WAV dosyalarının oluşturulacağı klasör belirlenmektedir.

Geliştirilen yazılımda saklama işlemi, belirli adımlar takip edilerek gerçekleştirilmektedir. Öncelikle girilen şifre kontrol edilmekte, eğer tutarsızlık varsa, bu durum bir uyarı mesajıyla kullanıcıya bildirilerek işlem sona erdirilmektedir. Şifrenin doğru belirlenmiş olması durumunda, bu şifre kullanılarak bir anahtar değeri elde edilmektedir. Yani şifre metni özetleme fonksiyonundan geçirilerek elde edilen sonucun byte değerleri dizisi rastgele saklama yapabilmek için anahtar olarak kullanılmaktadır. Saklanacak olan tüm dosya ve klasörlere ait bilgiler ise tek bir ikili veri objesi haline getirilmektedir. Yine kullanıcının belirlediği şifre ve algoritma ile, şifreleme objesi oluşturuluyor ve tek bir ikili dosya haline getirilen veri şifrelenmektedir. Sonraki aşamada, şifrelenmiş olan veri, WAV dosyalarının saklama kapasitelerine göre, taşıyıcı WAV dosyası sayısı kadar parçaya ayrılmakta ve bir döngü içerisinde, her bir veri parçası ilgili WAV dosyası içerisine saklama fonksiyonu kullanılarak saklanmaktadır. Saklama fonksiyonunun detaylarını incelemeden önce, şifreleme objesinin nasıl oluştuğuna değinelim. Şifreleme objesi oluşturulurken, yine kullanıcının en başta belirlediği şifre SHA fonksiyonu ile özetlenmekte, elde edilen byte değerleri kullanılarak seçilen algoritmaya uygun anahtarlar üretilip, belirlenen algoritmayı uygulayacak olan şifreleme objesi hazırlanmaktadır. Saklama işlemine ait akış diyagramları Şekil 6.2’de verilmiştir.

Saklama işleminin kalbi saklama fonksiyonudur. Saklama fonksiyonu temel olarak, saklanacak olan verinin bitlerini, WAV dosyasının, anahtar yardımıyla belirlenen pozisyonlarına saklanmasını sağlıyor. Burada bir bit değerini başka bir bit içine saklamaktan kasıt, taşıyıcı dosyadaki bit değerini, saklanacak olan bit değerine eşitlemek veya değiştirmektir.



Şekil 6.2. Saklama işlemi akış diyagramı

Saklama fonksiyonu modüler bir fonksiyon olarak tasarlanmıştır. Yani bir ikili verinin içerisine başka bir veriyi, bir anahtar yardımıyla, taşıyıcı dosyanın örneklerinin istenilen seviyesindeki bitlerine saklamaktadır. Burayı biraz açacak olursak, taşıyıcı yani barındırıcı verinin, örneklerden (samples) oluştuğu varsayılmaktadır. Mesela bu stereo bir WAV dosyası ise 2 byte değerindeki örneklerden oluşur. 24-bit Bitmap resim dosyası ise 3 byte büyüklüğündeki piksel veya örneklerden oluşur. İşte belirlenen bir parametre ile, taşıyıcı dosyaya ait

örneklerin kaç byte değerinden oluşacağı belirtilmektedir. Böylece, Bitmap, WAV veya benzer bir dosya türü için veri saklayabilir bir fonksiyon haline gelmektedir. Ayrıca sadece en az önemli bite değil, parametre ile belirtilen seviyedeki bitlere veri saklayabilmektedir.

Saklama fonksiyonu, anahtardan bir byte değerini okur ve okunan değer kadar örnek atladıktan sonraki örneğin belirtilen seviyesinde bite saklama işlemini yapar. Dolayısıyla saklama kapasitesi, anahtardaki byte değerlerine bağlıdır. Bu değerler büyük olursa, kapasite düşer. Fonksiyonda kullanılan bir parametre, bu değerlerin küçültülerek saklama kapasitesinin artırılması yönünde müdahale imkanı sunmaktadır.

Saklama işleminin sonucunda, başlangıçta belirtilen taşıyıcı WAV dosyaları, artık içlerinde veriyi saklayacak şekilde, ancak dosya özellikleri olarak orijinaliyle aynı, ses kalitesi olarak da orijinalinden ayırt edilemez WAV dosyaları olarak, belirlenen klasörün içine oluşturulmaktadır.

6.1.1. Saklama kapasitesi

STKK yazılımı taşıyıcı WAV dosyalarının sadece en az önemli bitlerini (LSB) kullanmaktadır. Ayrıca ardışık (sequential) değil, anahtardan okunan pozisyonlara rastgele (random) saklama yapmaktadır. Bu durum saklama kapasitesini düşürmektedir. Ancak saklama kapasitesi ile steganaliz ataklarına karşı dayanıklılık (robustness) arasında bir ters orantı (trade off) vardır. Saklama kapasitesinin düşmesine karşılık steganaliz karşısındaki sağlamlık artmaktadır.

WAV dosyaları ses örneklerinden (audio samples) oluşmaktadır. Bir ses örneği 8 veya 16 bit yani 1 veya 2 byte olabilmektedir. Saklama işleminde kullanılan anahtar değeri bir byte dizisidir. Saklanacak verinin okunan her biti, anahtardan okunan byte değeri kadar ses örneği (audio sample) atlanarak gelinen pozisyona gömülmektedir. Burada gömülmekten kasıt, belirlenen pozisyonadaki ses örneğinin son bitinin saklanacak olan bite eşitlenmesidir. Anahtarda okunan byte değerleri bittiğinde,

sıradaki saklanacak bitin saklanacağı pozisyonun bulunabilmesi için anahtardaki byte dizisinin başına dönülmektedir.

Saklama kapasitesinin hesaplanması için gerekli ifadeleri

B_{as} = ses örneği byte sayısı (1 veya 2)

N = anahtardaki byte dizisinin boyutu

I = saklamaya başlanılan ilk ses örneğinin pozisyonu

A_n = anahtardaki n. byte değerinin atlatacağı pozisyon sayısı

S = WAV dosyasının veri kısmının byte cinsinden boyutu

olarak tanımlayalım.

Veri saklama işlemi başladığında ilk olarak I değeri kadar ses örneği atlanacaktır. Geline pozisyona ilk bit saklanacaktır. Daha sonra anahtardan her byte değeri okunduğunda, o kadar ses örneği atlanacak ve gelinen pozisyona bir sonraki bit değeri gömülecektir. Böylece anahtarda her başa gelindiğinde N kadar bit saklanmış olacak ve anahtardaki byte dizisindeki değerlerin toplamı kadar yani $\sum A_n$ kadar ses örneği tüketilmiş olacaktır. Ses dosyasında S/B_{as} kadar ses örneği bulunacaktır ve bunun I kadar saklama işleminde kullanılmayacaktır. Saklama kapasitesinin hesaplanması için anahtarda kaç defa başa gelineceğinin bulunması gerekmektedir. Ses dosyasında saklama amacıyla kullanılacak ses örneği sayısını anahtardaki byte dizisindeki değerlerin toplamına böldüğümüzde saklama işlemi süresince

$$((S/B_{as}) - I) / \sum A_n \quad (6.1)$$

defa başa gelinecektir. Anahtarda her başa gelindiğinde N kadar bit yani $N/8$ kadar byte saklandığına göre, saklama kapasitesinin byte cinsinden yaklaşık değeri:

$$(((S/B_{as}) - I) * N) / (8 * \sum A_n) \quad (6.2)$$

olarak hesaplanacaktır. Bu değer hangi aralıkta yer alacağını ise şu şekilde belirleyebiliriz. Anahtar dizisindeki her byte değeri en düşük değere sahip olursa ardışık saklama yapılmış olacak, yani

$$\sum A_n = N \quad (6.3)$$

olacak ve saklama kapasitesinin en büyük üst sınırı

$$((S/B_{as}) - I) / 8 \quad (6.4)$$

byte olacaktır. Anahtardaki byte değerlerinin tamamı en yüksek değerde olduğunda ise;

$$\sum A_n = N * 2^8 \quad (6.5)$$

olacak ve saklama kapasitesinin en düşük değeri

$$((S/B_{as}) - I) / (8 * 2^8) = ((S/B_{as}) - I) / 2^{11} \quad (6.7)$$

olacaktır. Bu durumda saklama kapasitesinin değer aralığı aşağıdaki gibi ifade edilebilir:

$$((S/B_{as}) - I) / 2^{11} \leq (((S/B_{as}) - I) * N) / (8 * \sum A_n) \leq ((S/B_{as}) - I) / 8 \quad (6.8)$$

6.1.2. Oluşan hata

Veri saklama işleminde her bir bit saklandığında, taşıyıcı ses dosyasının belirlenen pozisyonunda bit değeri, saklanacak olan bite eşitlenmekteydi. Eğer saklanacak olan bit, değişecek olan bit değeri ile aynıysa hiçbir değişim gerçekleşmemekte ancak eğer farklıysa ses dosyasına ait sayısal verilere belli bir miktarda hata girilmiş olmaktadır. Saklama işlemi sonucunda ne kadar hata oluşacağını hesaplamak için:

L = saklama yapılan bit seviyesi

D = saklama sonucu değişen yaklaşık bit sayısı

S = saklanan verinin byte cinsinden boyutu

tanımlarını yapalım. Bu tanımlardan, saklama işleminde bit değeri değişirse, o ses örneği için oluşturulan hata miktarı 2^L olacaktır. Toplam hata ise yaklaşık olarak;

$$D * 2^L \quad (6.9)$$

olmaktadır.

Ortalama olarak taşıyıcı dosyadaki barındırıcı bitlerin %50 oranında değiştiğini varsayarsak;

$$D = S * 8/2 \quad (6.10)$$

olacak ve girilen oluşturulan yaklaşık hata miktarı

$$4 * S * 2^L = S * 2^{L+2} \quad (6.11)$$

olacaktır. STKK uygulamasında sadece LSB'ye saklama yapıldığı için L değeri "0" dır ve bu uygulamada girilen ortalama hata miktarı $4 * S$ olarak hesaplanabilir.

Saklama sonucu oluşan gürültü bu hatadan kaynaklanmaktadır. İstatistiksel olarak bu hata değerinin yanı sıra, değişen bitlerin pozisyonları ve birbirleriyle olan uzaklık ve yakınlıkları da steganaliz ataklarına karşı parmak izleri oluşturmaktadır. Geliştirilen STKK yazılımı L değerini en düşük seviyede tutarak, yani sadece LSB ye veri saklayarak saklama kapasitesi düşürülse de, buna karşılık steganaliz ataklarına karşı riskin de düşürüldüğünü belirtmekte fayda vardır.

6.2. Çıkartma İşlemi

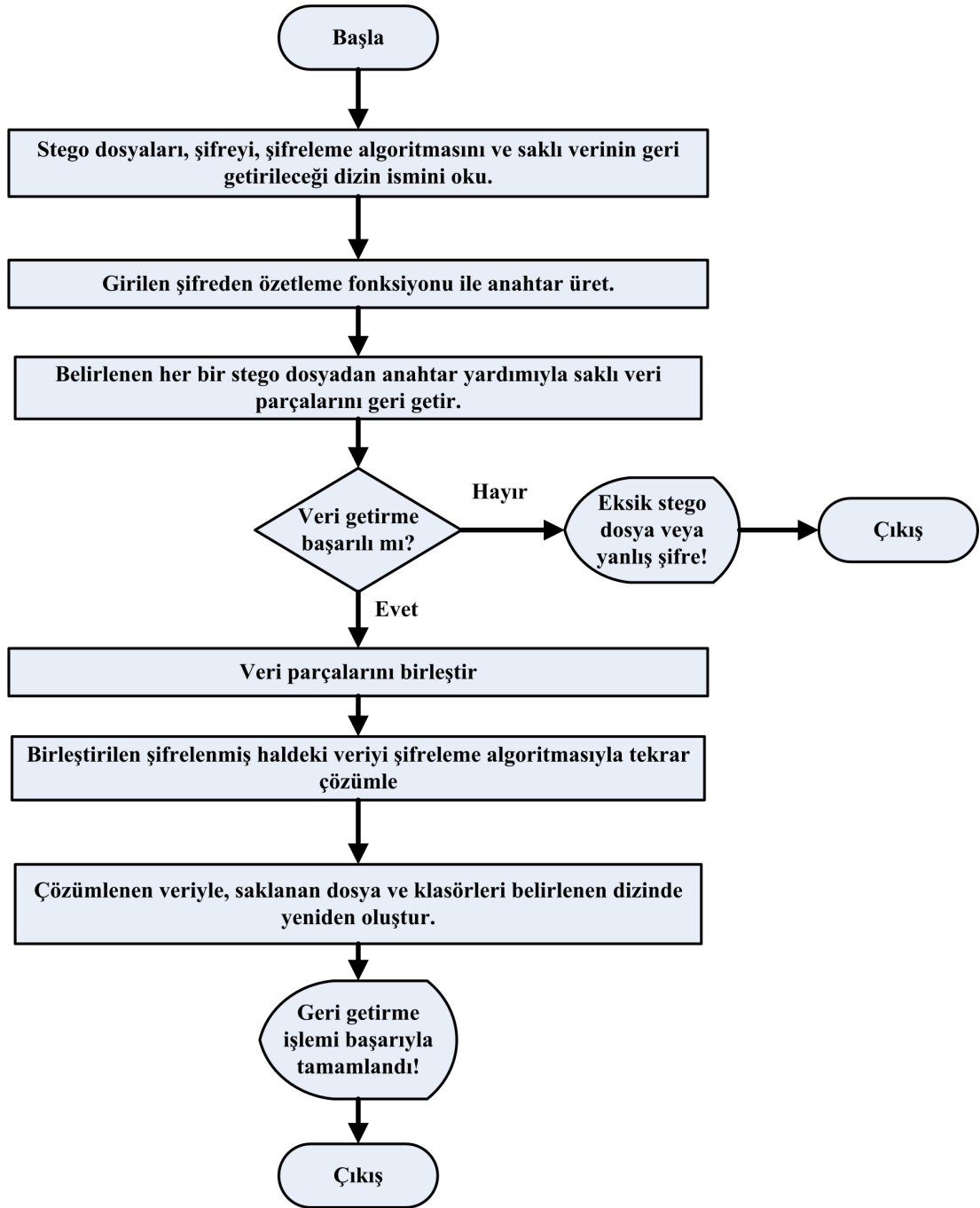
İçerisinde veri saklı olan WAV dosyalarından, bu veriyi geri getirme işlemi, daha önce de belirtildiği gibi, saklama işlemindeki aşamaların tersten yapılmasıyla gerçekleştirilmektedir. Yazılımda öncelikle stego dosya setinin belirtilmesi gerekmektedir. Sonraki aşamada, belirtilen stego WAV dosyaları içerisindeki verinin

hangi dizin veya klasöre geri getirileceği belirlenmektedir. Ayrıca saklama işleminde kullanılan şifre ve seçilen şifreleme algoritması da aynen belirtilmelidir. Aksi halde çıkartma işlemi başarıyla tamamlanamayacaktır. Bütün bunların belirlenmesinin ardından çıkartma işleminin arka planında bazı işlemlerin gerçekleştirilmesi gerekmektedir.

Girilen şifre saklama işleminde kullanılan aynı özetleme (hash) fonksiyonu ile byte dizisi haline getirilir ve anahtar değeri elde edilir. Daha sonra belirtilen stego WAV dosya setindeki her bir WAV dosyasından, elde edilen anahtar değeri yardımıyla belirlenen pozisyonlardaki bit değerleri okunur ve bir bit kanalı (stream) içerisine yazılır. Her bir stego WAV dosyasından bu şekilde okunan ikili veriler, veriyi belirleyici tekil numara yardımıyla birleştirilerek tek bir ikili veri haline getirilir. Bu veri, saklanan orijinal verinin, saklama işleminde oluşturulan şifreleme algoritması objesiyle kodlanmış halidir. Dolayısıyla orijinal veriyi elde edebilmek için şifre ve algoritma kullanılarak yeniden şifreleme sınıfı objesinin oluşturulması ve bu kez deşifre fonksiyonu ile şifrelenmiş verinin tekrar çözülmesi gerekmektedir.

Deşifre işlemi sonrasında oluşan veri ise, saklama işleminde belirtilen dosya ve klasörlere ait tüm bilgilerin birleştirilerek dönüştürüldüğü bir ikili veri ile aynıdır. Bu durumda, yine ters işlemleri yapan bir fonksiyon yardımıyla, bu ikili veri, dosya ve klasör bilgilerine dönüştürülmekte, dosya ve klasörlerin orijinal halindeki hiyerarşisine göre yeniden oluşturulmaktadır. Geri getirme işleminin başarıyla sonuçlanabilmesi için saklama sonucu oluşan stego dosyaların eksiksiz olması gerekmektedir. Ayrıca, saklama işleminde kullanılan şifre, şifreleme algoritması, özetleme fonksiyonu, verinin saklandığı bit seviyesini belirleyen parametre değeri ve saklama kapasitesini ayarlama imkanı sağlayan parametre değerlerinin birebir aynı olması gerekmektedir. Geri getirme işleminde her bir stego dosyadan, saklanan verinin çıkartılmasını sağlayan fonksiyon bu işlemin temelini oluşturmaktadır. Bu fonksiyon, kullanıcının belirlediği şifreden elde edilen anahtar değerini ve saklama işleminde kullanılan parametre değerleri ile stego WAV dosyasını girdi değişkenleri olarak alarak sonuçta saklanmış ikili veriyi döndürmektedir. Stego dosyanın en

sonuna gelene kadar bir döngü içerisinde, anahtardan okunarak tespit edilen pozisyonundaki bit değerleri sırayla okunup, ikili sonuç dosyası içerisine yazılmaktadır. Yazılımın geri getirme işlemine ait akış diyagramı Şekil 6.3’de verilmiştir.



Şekil 6.3. STKK yazılımı geri getirme işlemi akış diyagramı

6.3. Yazılımın Özellikleri

STKK yazılımı, başarılı bir steganografi yazılımı için iki temel gereksinim olan saklama işleminin şeffaf bir şekilde gerçekleştirilip oluşan stego dosyalardaki farklılığın hissedilememesi ve saklanan verinin tam ve düzgün bir şekilde tekrar geri getirilmesi gereksinimlerinin ikisini de karşılamaktadır. Oluşan stego müzik dosyaları dinlendiğinde orijinaliyle aralarında fark anlaşılamamaktadır. Ayrıca yazılımın geri getirme modülü tam düzgün bir şekilde saklanan tüm dosyaları ve aralarındaki ilişkileri geri getirebilmektedir.

Ayrıca steganografi uygulamalarının steganaliz ataklarına karşı dayanıklı olması gerekmektedir. Steganalizde amaç sadece saklanan veriyi ortaya çıkartmak değil, bir dosya içerisinde veri saklı olup olmadığını da tahmin etmektir. Normal şartlarda ses dosyalarının en az öneme sahip 4. bitlerine kadar veri saklanabilmektedir. Saklama kapasitesinin önemli olduğu uygulamalarda bu tercih edilebilir. Ancak bu durum istatistiksel ataklara karşı uygulamayı zayıflatmaktadır. Veri saklama işleminin yapıldığı bit seviyesi arttıkça ses dosyası içerisine eklenen hata yükü de artmakta ve istatistiksel izler bırakmaktadır. Bu durum bize steganografi uygulamalarında, saklama kapasitesi ile istatistiksel ataklara karşı dayanıklılık arasında bir ters orantı olduğunu söylemektedir. Ayrıca bir ses veya resim dosyasının en az öneme sahip bitlerine (LSB) ardışık veya periyodik bir şekilde ardışık veri saklama yöntemlerinin de steganalizde kullanılabilecek istatistiksel izler bıraktığı belirtilmektedir.

STKK yazılımı saklama işlemini bir anahtar aracılığıyla rastgele gerçekleştirmektedir. Bu durum istatistiksel steganaliz ataklarına karşı da yazılımı daha dayanıklı hale getirmektedir. STKK veri saklamak için sadece LSB'yi kullanmaktadır. Hem anahtardan okunan değerlerle rastgele pozisyonlara saklama yapılmasının hem de sadece LSB'ye verinin saklanması, STKK yazılımını istatistiksel ataklara karşı oldukça dayanıklı yapması beklenmektedir.

Saklama kapasitesi de bazı uygulamalar için önemli hale gelmektedir. Ancak yukarıda da bahsedildiği gibi steganalize karşı sağlamlık ile saklama kapasitesi

arasında ters orantı bulunmaktadır. STKK yazılımı bu duruma da dolaylı bir çözüm bulmaktadır. Tek bir ses dosyası içerisine veri saklama oranı, steganaliz ataklarına dayanıklılık nedeniyle düşüktür ancak yazılım birden fazla ses dosyası içerisine yani ses dosyası setine veri saklayabilmektedir. Bu da steganalize karşı dayanıklılığı arttırmaktadır. Bu durumda yeterince taşıyıcı ses dosyası olduğu müddetçe saklanabilecek verinin boyutunda bir sınırlama yoktur.

Geliştirilen STKK yazılımı aşağıdaki özelliklere sahiptir:

- Klasörler ve dosyalar setini, WAV ses dosyaları seti içerisine şeffaf bir şekilde saklayabilmekte ve geri getirebilmektedir.
- Saklama işleminin yanı sıra , AES, DES ve 3DES algoritmalarından biriyle şifreleme imkanı da sunmaktadır.
- Saklama işlemini sadece bilgisayar üzerindeki değil, usb disk, flash bellek veya herhangi bir harici disk üzerindeki WAV dosyaları seti içerisine saklayabilmekte, saklanan klasörü de, başka bir bilgisayar üzerinde geri getirebilmektedir. Yani temel de bir steganografi uygulaması olmakla birlikte, stego tabanlı klasör kitleme uygulaması olarak da kullanılabilir.
- Saklama işlemi sonrasında oluşan ses dosyaları, diskteki boyutları ve ses kalitesi yönüyle orijinalinden farksızdır.
- Yeterince ses dosyası olduğu müddetçe saklanacak verinin boyutunda herhangi bir sınırlama yoktur.
- Hangi tipten olduğu fark etmeksizin her türlü dosya saklanabilmektedir.
- Belirlenen bir anahtar yardımıyla rastgele saklama yapılarak istatistiksel ataklara karşı dayanıklılık sağlamaktadır. Ayrıca bu anahtar bilinmeden saklanan veriyi geri getirmek mümkün olmayacaktır.
- Saklama işleminin güvenliği saklama algoritmasının bilinmesine değil belirlenen anahtar değerine bağlıdır. Saklama işlemi sonrasında kullanım özelliğine göre saklanan verinin boyutu kadar alan diskten boşaltılmaktadır. Yani saklanan veri disk gözlemleme yapan yazılımlar açısından saklanmış değil, silinmiş gibi algılanacaktır.
- Kriptografi ve steganografi yaklaşımlarının her ikisini de içerdiğinden yüksek bir güvenlik sağlamaktadır.

- Oluşan stego ses dosyaları ses kalitesi yönüyle herhangi bir gürültü içermemekte ve steganografinin temel mantığına da uygun olarak herhangi bir verinin saklı olduğu fark edilememektedir.
- Kullanıcıya saklama kapasitesinin ayarlanması konusunda esneklik sağlanabilmektedir.
- Saklama işlemi sonrasında saklanan klasör ve dosyaları işletim sisteminin arama araçları kullanarak bulmak mümkün değildir.
- Yazılıma ait dosyalar disk üzerinde çok küçük bir yer kaplamaktadır.
- Yazılım Visual Studio .Net 2003 ortamında C# programlama diliyle gerçekleştirilmiştir. Yazılımın çalışabilmesi için bilgisayarda .net framework kurulu olması gerekmektedir.
- Yazılım ara yüzü ve mesajları Türkçe'dir ve ülkemizde ses dosyaları seti içerisine veri saklayan ilk yazılım olduğu değerlendirilmektedir.

6.4. STKK ve Diğer KK Yazılımları

Geliştirilen STKK yazılımının işletim sistemi tabanlı klasör kilitleme uygulamalarına göre dezavantajı, saklanacak veriden çok fazla büyük boyutlarda taşıyıcı ses dosyası setine ihtiyaç duymasıdır. Ayrıca oluşan stego ses dosyaları silindiğinde saklanan veriyi geri getirmek mümkün olmayacaktır. Ancak bununla birlikte STKK yazılımının, bu bölümün önceki alt başlıklarında açıklanan ve sistem tabanlı çalışan diğer KK (Klasör Kilitleme) yazılımlarına karşı çok önemli iki avantajı bulunmaktadır:

1. KK yazılımları klasörü sakladığında diskten silmemekte sadece kullanıcıya görme ve erişim izni vermemektedir. Dolayısıyla saklı klasörün boyutu kullanıcıya görünmediği halde kullanılan disk miktarının içerisinde yer alacaktır. Ancak STKK yazılımıyla, bir klasör, bilgisayarın kendi diskindeki veya harici bir diskteki ses dosyalarına saklanacak ve silinecektir. Böylece disk üzerinde hem görünmeyip hem yer kaplayan bir veri yer almayacaktır.

2. KK yazılımlarında, saklanacak klasör özel bir klasör içine taşınmakta ve bu durum bazı saklanmış klasörler olduğunu belli etmektedir. Ancak STKK ile bilgisayar üzerinde saklı bir klasör olduğunun farkında bile olunmayacaktır.
3. Sadece iletişimi değil veri gizliliğini de tek bir yapı altında sunmaktadır.

Özetlemek gerekirse, STKK yazılımı bir anahtar yardımıyla, ses dosyaları içerisine rastgele veri saklayan ve aynı zamanda şifreleme imkanı sunan bir steganografi uygulamasıdır. Anahtar kullanarak rastgele veri gömmesi özelliğiyle istatistiksel steganaliz ataklarına karşı dayanıklılığı arttırmakla beraber, sonuçta oluşan stego ses dosyaları, yani içinde veri saklı olan ses dosyaları işitsel olarak orijinalinden farksızdır. Ardışık saklama yerine rastgele saklama yapması steganaliz ataklarına karşı yüksek bir koruma sağlarken saklama kapasitesini ise düşürmektedir.

Geliştirilen STKK yazılımının veriyi bir tek ses dosyası yerine ses dosyaları setine saklaması ve WAV dosya tipi gibi büyük oranda veri saklama imkanı sağlayan bir ses dosyası türü içine saklaması yönüyle, saklama kapasitesine dolaylı bir çözüm sunmaktadır. Yani yeterince taşıyıcı ses dosyası olduğu müddetçe saklanacak verinin boyutunda bir sınır bulunmamaktadır. Bu açıdan STKK esasta bir steganografi uygulaması olmakla birlikte, lokal bilgisayardaki WAV dosyalarını, bir klasörün saklanabilmesi amacıyla kullanarak, klasör kitleme uygulaması olarak da tanımlanabilir. İşletim sistemi fonksiyonlarını kullanan klasik KK uygulamalarının aslında diskte varolan bir klasörü gizleme özelliğine karşılık, STKK yazılımının, klasörü tamamen diskten silerek başka bir veri içerisine saklama özelliği, bu tür KK yazılımlarına karşı bir üstünlüğüdür.

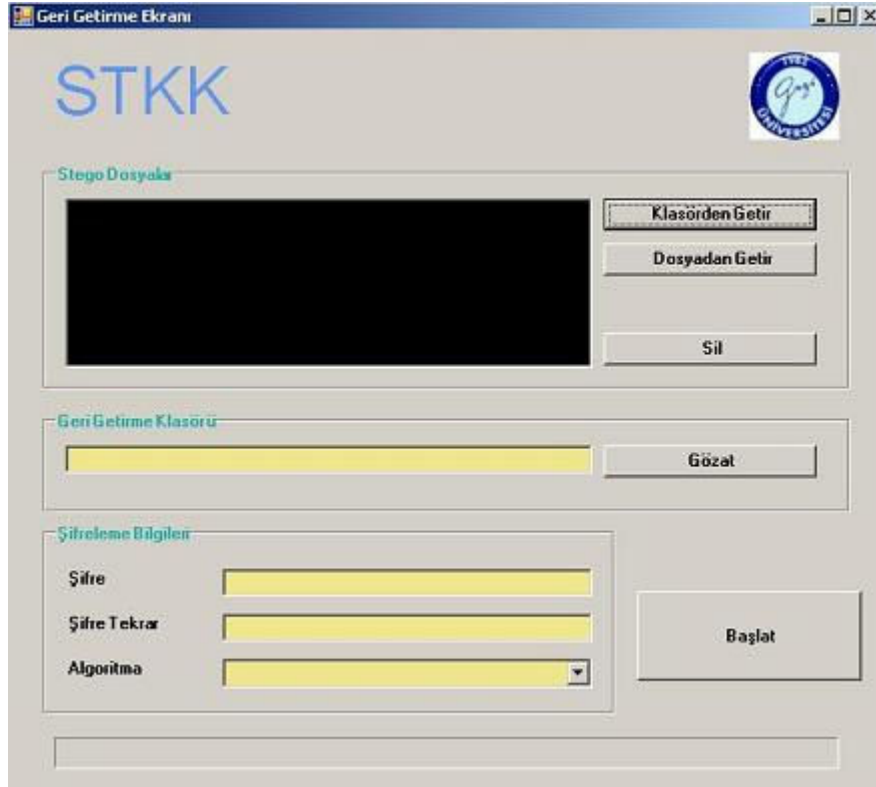
Şekil 6.4’de geliştirilen STKK yazılımının saklama işlemi ara yüzü verilmiştir. Bu ara yüzden de görülebileceği gibi, ekranın üst kısmında saklanacak olan klasör veya dosyalar ve klasörler seti belirlenmektedir.



Şekil 6.4. STKK Yazılımı veri saklama ekranı

“Klasörden Getir” tuşu ile saklanması düşünülen klasörler tek tek belirlenebilir. “Dosyadan Getir” tuşu ile de saklanması düşünülen dosyalara tek tek göz atılarak bu dosyalar listeye eklenebilir. Bir alt bölümde ise taşıyıcı WAV dosyaları belirtilerek listeye eklenmektedir. Bu dosyalar “Dosyadan Getir” tuşu ile tek tek göz atılıp listeye eklenebileceği gibi “Klasörden Getir” tuşu ile WAV dosyalarının topluca yer aldığı klasör seçilerek tüm dosyalar topluca seçilip listeye eklenebilir. Hem üst kısımdaki hem de bir alt bölümdeki “Sil” tuşları ise isminden de anlaşılacağı gibi listedeki seçili bir nesneyi listeden kaldırma durumunda kullanılmaktadır. Sağ üst köşede şifreleme seçenekleri yer almaktadır. Kullanıcı buradaki metin kutusunu kullanarak şifresini tekrarlayarak girer ve bir şifreleme algoritması seçer. Herhangi bir algoritma seçimi olmazsa uygulama AES algoritmasını kullanarak şifreleme yapmaktadır. Sağ alt köşe de ise sonuçta oluşacak

stego WAV dosyalarının hangi dizinde oluşturulacağı belirlenmektedir. “Başlat” tuşuna basılmasıyla birlikte detayları daha önce anlatılan saklama işlemi aşamaları sırayla gerçekleşmeye başlar ve işlemin ilerleyişi, açılan ara yüzün alttaki barında görüntülenir. Şekil 6.5’de ise geri getirme işlemi için geliştirilen ara yüz verilmiştir.



Şekil 6.5. STKK Yazılımı veri geri getirme ekranı

Ekranın üst kısmında stego WAV dosyaları bulundukları klasörden topluca veya bulundukları dizinlerden tek tek “Klasörden Getir” ve “Dosyadan Getir” tuşlarıyla listeye eklenir. Orta bölümde ise geri getirilecek verinin oluşturulacağı dizin belirlenir. Alt kısımda saklama ekranında olduğu gibi şifre ve algoritma girişleri yapılır. Burada belirtilen bilgiler saklama işlemindeki ile aynı olmazsa geri getirme işlemi tamamlanamaz. “Başlat” tuşuna basılmasıyla, detayları yine yukarıda anlatılan çıkartma işlemi aşamaları sırayla gerçekleşerek saklanan verinin geri getirilmesi ile işlem tamamlanır.

6.5. Geliştirilen STKK Yazılımı Modülleri

Geliştirilen STKK yazılımı 4 modülden oluşmaktadır. Bunlar:

1. Şifreleme Operasyonları (kriptoOperations) modülü
2. Stego Opreasyonları (stegoOperations) modülü
3. Klasör Kilitleme Operasyonları (folderLockOperations) modülü
4. WAV Okuyucu (WAVEReader) modülü

Bu modülleri aşağıda kısaca açıklayalım.

Şifreleme Operasyonları modulünde 3 temel fonksiyon bulunmaktadır. Bunlar:

- AlgoritmaUret(..)
- Sifrele(..)
- Desifre(..)

AlgoritmaUret(..) fonksiyonu kullanıcı ara yüzü aracılığıyla girilen şifreyi ve seçilen şifreleme algoritması türünü kullanarak bir şifreleme algoritması üretir. Şifre SHA özetleme fonksiyonu kullanılarak byte dizisine dönüştürülür. Daha sonra bu byte dizisi kullanılarak belirlenen şifreleme algoritmasına ilişkin anahtar değeri belirlenir. Bu fonksiyon sonuçta şifreleme ve şifre çözme işlemlerinde kullanılmak üzere bir simetrik şifre algoritması objesi döndürmektedir.

Sifrele(..) fonksiyonu ikili haldeki bir veri kanalı ile bir simetrik şifreleme algoritmasını girdi olarak alıp bu algoritmanın *Encrypt(..)* fonksiyonunu kullanarak ikili veri kanalının şifrelenmiş halini sonuç olarak döndürmektedir.

Desifre(..) fonksiyonu da *Sifrele(..)* fonksiyonunun -doğal olarak- tam tersine, şifrelenmiş haldeki bir ikili veri kanalı objesi ile bir simetrik şifreleme algoritması objesini girdi olarak alarak bu algoritmanın *Decrypt(..)* fonksiyonu ile ikili veri kanalını normal haline dönüştürüp sonuç olarak döndürmektedir.

Stego Opreasyonları modülünde 4 temel fonksiyon bulunmaktadır. Bunlar:

- BitDegeriniGetir(..)
- ChangeBitValue(..)
- VeriDegistir(..)
- VeriGetir(..)

olarak sıralanabilir. Bu fonksiyonları kısaca açıklayalım.

BitDegeriniGetir(..) fonksiyonu belirtilen byte değişkenin belirtilen seviyesindeki bit değerini yani 1 mi veya 0 mı olduğunu döndürmektedir.

ChangeBitValue(..) fonksiyonu ise belirtilen byte değişkenin belirtilen seviyesindeki bit değerini yine belirtilen bit değerine eşitleyerek byte değişkeninin yeni değerini döndürmektedir.

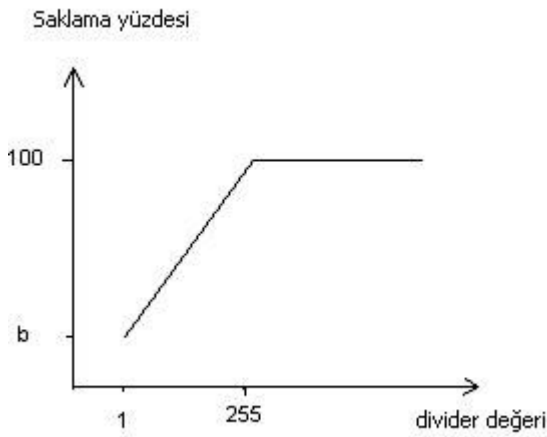
VeriDegistir(..) fonksiyonu, ikili haldeki bir veri kanalına ilgili değişkenleri girdi olarak alıp belirtilen veriyi saklayarak sonuçta oluşan stego veriyi döndüren fonksiyondur. Bu fonksiyonda saklamayı etkileyen değişkenler *Key*, *Divider*, *Level* ve *SampleByteSayisi* değişkenleridir. *Key* bir sonraki saklama pozisyonun byte olarak okunduğu bir ikili veri kanalıdır. *Key* değişkenindeki byte değerleri sırayla okunarak bir sonraki saklama pozisyonu belirlenir. *SampleByteSayisi* değişkeni ise taşıyıcı olarak kullanılan verinin en küçük birimini ifade eder. *Level* verinin hangi seviyedeki bit değerlerine saklanacağını belirler. *Divider* değişkeni saklama kapasitesini ayarlamaya yönelik bir değişkendir. Bu değişkenle oynayarak saklama kapasitesinin büyüklüğü ayarlanabilir. Bu fonksiyonun işleyişini bir örnekle açıklayalım. *Key* değişkeninden okunan sıradaki byte değeri 203 olsun. *Level* değişkeninin değerinin ise 2 olduğunu varsayalım. *Divider* değişkeninin değeri 5 *SampleByteSayisi* değişkeninin değeri de 2 olacak olursa, saklanacak olan verinin sıradaki biti şu anda bulunulan pozisyondan

$$(203/5 + 1) * 2 = 82$$

(6.12)

kadar ileri byte değerine saklanır. Byte değeri 0 ile 255 arasında olabileceğinden atanacak pozisyon sayısı 1 ile 52 *SampleBytesayisi* arasında değişecektir.

Divider değeri attıkça saklama kapasitesi artacaktır. *Divider* değerinin 255'den büyük olması durumunda ardışık saklama yapılmış olacaktır. Bu durumda tüm bitler saklama için kullanılabileceğinden saklama kapasitesi %100'e yükselecektir. Şekil 6.6'da *Divider* değeri ile saklama kapasitesi arasındaki ilişki verilmektedir.



Şekil 6.6. Saklama Yüzdesi & Divider değişkeni ilişkisi

Bu uygulamada PCM formatında stereo WAV dosyaları kullanılması durumunda *SampleByteSayisi* değeri 2, mono olması durumunda ise 1 olacaktır. *Level* değişkeni 0 olup LSB modifikasyonu yapılmaktadır.

VeriGetir(..) fonksiyonu ise *VeriDeğiştir(..)* fonksiyonundaki parametrelerle birlikte şifreleme algoritması objesini de girdi olarak alıp, saklama işleminin tam tersi olarak stego WAV dosyasından saklı olan veriyi çıkartmayı sağlamaktadır.

Klasör Kitleme Operasyonları modülünün bir çok fonksiyonu olmakla birlikte burada 3 önemli fonksiyonuna değinilecektir. Bunlar:

- *MakeBitStream(..)*
- *Capacities(..)*
- *VeriPaylastir(..)*

olarak verilebilir. Bu fonksiyonlar aşağıda kısaca açıklanmıştır.

MakeBitStream(..) fonksiyonu klasörler ve dosyalar dizilerini girdi olarak alıp saklanacak olan veriye ait meta bilgi oluşturarak tüm veri tek bir ikili sayısal veri kanalına dönüştürmektedir.

Capacities(..) fonksiyonu dosyaların boyutları ve anahtar değerini ve saklama kapasitesini etkileyen diğer parametreleri de girdi olarak alıp, her bir taşıyıcı dosyanın saklama kapasitelerini hesaplayıp dönmektedir.

VeriPaylastir(..) fonksiyonu ise *Capacities(..)* fonksiyonu ile *MakeBitStream(..)* fonksiyonlarından dönen sonuçları ve şifreleme algoritması objesini girdi olarak alarak her bir taşıyıcının kapasitesine göre saklanacak olan veriyi ikili veri kanalı parçalarına bölüp sonuçları bir dizi olarak dönmektedir.

WAV Okuyucu modülü de yine bir çok fonksiyona sahip olmakla birlikte önemli olan üç fonksiyonuna burada değinilecektir. Bunlar:

- *ReadHeader(..)*
- *SetData(..)*
- *ComposeNewWAVFile(..)*

olup aşağıda kısaca açıklanmıştır.

ReadHeader(..) fonksiyonu WAV dosyası verisinin başlık kısmını okuyarak dosyaya ait parametre değerlerini elde etmektedir.

SetData(..) fonksiyonu WAV dosyasının veri bölümünü girdi olarak alınan ikili veri kanalına eşitlemektedir.

ComposeNewWAVFile(..) fonksiyonu ise başlık parametreleri elde edilmiş ve veri bölümü stego veri kanalına eşitlenmiş yeni WAV dosyasına ait ikili veri kanalını yeniden oluşturmaktadır.

Geliştirmiş olduğumuz bu yazılımda ilk göze çarpan nokta eniyileme (optimization) yapılmamasıdır. Saklanacak olan veri taşıyıcı ses dosyalarına kapasiteleriyle orantılı olarak bölüştürülmektedir. Bunun yerine hangi dosyaya hangi verinin daha az değişimle saklanacağı veya yeterli sayıda taşıyıcı dosya varsa, hepsine mutlaka saklamak yerine veri saklanması en uygun olanlarının belirlenmesi gibi konuların dikkate alınıp eniyileme yapılması bu çalışmanın ileriki aşamasını oluşturmaktadır. “En şeffaf stego dosya içerisine hiç veri saklanmamış olan dosyadır!” ifadesinde belirtildiği gibi, steganografi uygulamalarında amaç en şeffaf şekilde veriyi saklayabilmek ve iletişimin gizliliğini sağlamaktır. Saklama işleminde değişim ne kadar az olursa şeffaflık o ölçüde artacaktır. Çeşitli saklama parametrelerinin varlığı eniyileme imkanı sağlamak ve genetik algoritma benzeri eniyileme algoritmaları ile daha şeffaf saklama işlemleri gerçekleştirilebilecektir.

6.6. STKK Yazılımının Mevcut Steganografik Yazılımlarla Karşılaştırılması

Bölüm 2.9’da tanıtılan steganografik yazılımlar ile bu tez çalışmasında geliştirilen STKK yazılımı taşıyıcı dosya türü, işletim sistemi, şifreleme imkanı, saklama yöntemi ve saklama kapasitesi yönünden karşılaştırılmıştır.

Tekrar hatırlatılacak olunursa, tanıtılan steganografik yazılımların çoğu resim içerisine veri saklayabilmektedir. Ses içerisine veri saklayan yazılım sayısı daha azdır. STKK yazılımı şu anda sadece WAV dosyalarını desteklemektedir. Ses içerisine veri saklayan herhangi bir Türkçe yazılım şu anda mevcut değildir. STKK yazılımının en önemli özelliklerinden birisi bu konudaki eksikliği gidermeye yönelik katkısıdır.

Steganografik yazılımları şifreleme seçeneği yönünden ele aldığımızda, şifreleme imkanı sunan yazılım sayısı çok fazla değildir. STKK yazılımı en etkin şifreleme algoritmalarından AES, DES ve 3DES ile şifreleme imkanı sunmaktadır.

Gizli bir anahtar yardımıyla saklama, rastgele saklama işlemi yapılması istatistiksel steganalize karşı sağlamlığı arttırmaktadır. STKK yazılımı da bir anahtar yardımıyla

rastgele saklama işlemi yapmakta ve böylelikle saklama işleminin güvenliği saklama algoritmasına bağlı olmayan yazılımla grubunda yer almaktadır.

Çizelge 6.1. STKK yazılımının mevcut yazılımlarla karşılaştırılması

Yazılım İsmi	Yazılım Özellikleri						
	Taşıyıcı dosya türü	Saklanan dosya türü	Dosya setine saklama	Şifreleme	Saklama kapasitesi	Klasör saklama özelliği	Dil
Snow	ascii txt	Metin	yok	Var	< TB	yok	İngilizce
Puff	Bmp, jpg, pcx, png, tga, aiff, mp3, wav	Hepsi	var	Var	256 mb.	var	İtalyanca
wbstego	Bmp, html, txt, pdf	Hepsi	yok	Var	< TB	yok	İngilizce
securengine	htm, bmp, gif, png	Hepsi	yok	Var	< TB	var	İngilizce
hermetic	Bmp, jpg, pcx, png, tga, aiff, mp3 ve wav	Hepsi	var	Var	yeteri kadar taşıyıcı varsa sınırsız	yok	İngilizce
stegoMagic 1.0	txt, bmp, wav	Hepsi	yok	Var	TB / 8	yok	İngilizce
Mp3stego	mp3	Metin	yok	Var	< TB	yok	İngilizce
stegHide	jpg, bmp, wav, au	Hepsi	yok	Var	< TB	yok	İngilizce
InfoStego	bmp	Hepsi	yok	Var	< TB	yok	İngilizce
gifshuffle	gif	Metin	yok	Var	< TB	yok	İngilizce
deogol	html	Metin	yok	Yok	< TB	yok	İngilizce
Publimark 0.1.2	ses	Metin	yok	Var	< TB	yok	İngilizce
Hydan	exe	Metin	yok	Var	TB / 110	yok	İngilizce
Jphs	jpg	Hepsi	yok	Var	< TB	yok	İngilizce
TürkSteg	bmp	metin	yok	var	< TB	yok	Türkçe
Verigizle.com	bmp	metin	yok	var	< TB	yok	Türkçe
STKK	wav	hepsi	var	Var	yeteri kadar taşıyıcı varsa sınırsız	var	Türkçe

Steganografik yazılımlardan bazıları taşıyıcı dosyanın %50'si kapasiteyle saklama yapabilmektedir ancak bunun steganalize ne kadar dayanıklı olacağı tartışmalıdır. STKK yazılımı sadece en az önemli bitlere rastgele saklama yaptığından bir WAV dosyasının çok küçük bir kısmını barındırıcı olarak kullanmaktadır. Ancak saklama işlemini WAV dosya seti içerisine yapan nadir yazılımlardan birisi olduğundan yeteri

kadar WAV dosyası olduğu müddetçe saklanacak verinin boyutunda bir sınırlama getirmemektedir. STKK yazılımını diğer yazılımlardan ayıran en önemli özelliği ise bir klasörü ve klasörler ve dosyalar setini tüm içeriği ve hiyerarşisiyle birlikte saklayabilmesi, yani stego tabanlı klasör kitleme yazılımı işlevi de görebilmesidir.

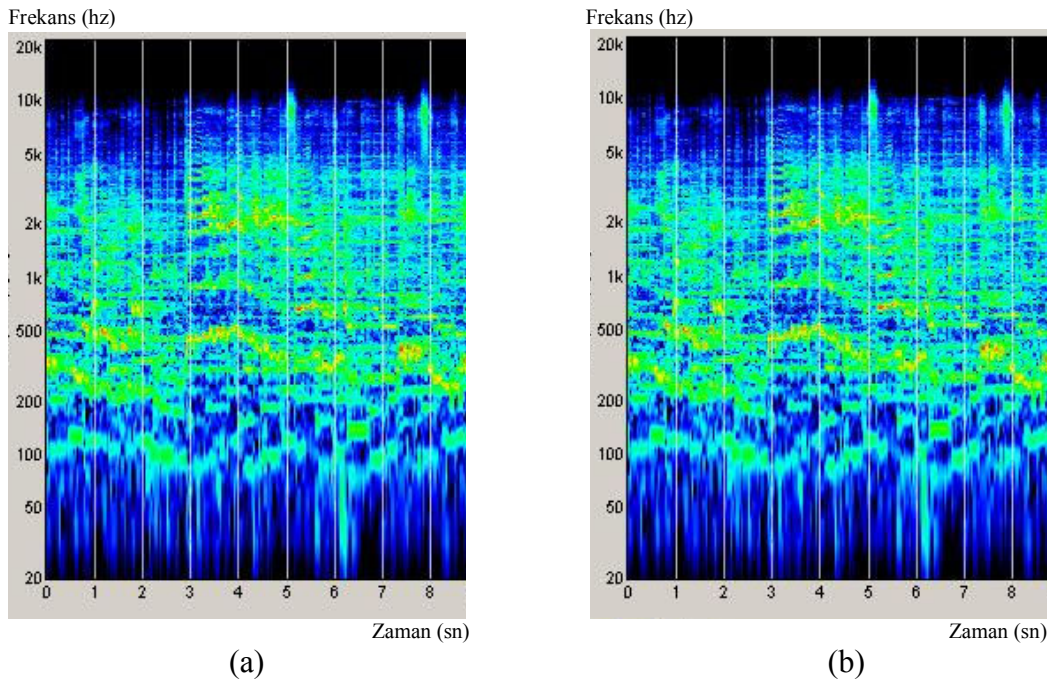
Çizelge 6.1’de, Bölüm 2.9’da anlatılan mevcut steganografik yazılımlarla STKK yazılımının özellikleri yer almaktadır. Bu çizelge incelediğinde, STKK yazılımının birçok yazılımdan üstün olduğu görülmektedir. Çizelge 6.1’de sunulan kriterlere göre, STKK yazılımı ses içerisine veri saklayan ilk Türkçe yazılımdır. Tek bir taşıyıcı dosya yerine, taşıyıcı dosyalar setine saklama yapabilen 3 yazılımdan birisi olmakla birlikte yine bir klasörü tüm hiyerarşisiyle birlikte saklayabilen 3 yazılımdan biridir. İtalyanca bir yazılım olan Puff ile benzer özellikler taşımaktadır. Puff daha çeşitli taşıyıcı dosya türleri içerisine saklama yapabilmekte ancak, taşıyıcı dosya boyutu olarak 4 GB, saklanacak veri boyutu olarak ise 256 MB sınırlama getirmektedir. STKK yazılımı taşıyıcı dosya olarak şimdilik sadece PCM biçimindeki WAV dosyalarını desteklemekte, ancak taşıyıcı dosya boyutu ve saklanacak veri boyutu açısından bir kısıtlama getirmemektedir. Özetlemek gerekirse, STKK yazılımı mevcut steganografik yazılımlardaki gelişmiş tüm özelliklere sahip bir yazılım olup, ses dosyası temelli klasör kitlemeye sahip, taşıyıcı veride ve saklanacak veride sınırı olmayan (yeteri kadar taşıyıcı dosya seçildiği müddetçe), gerektiğinde farklı şifreleme algoritmaları kullanılarak yüksek seviyede güvenlik sağlayan, tamamen tarafımızdan geliştirilmiş Türkçe bir yazılımdır.

6.7. Örnek Saklama İşlemi

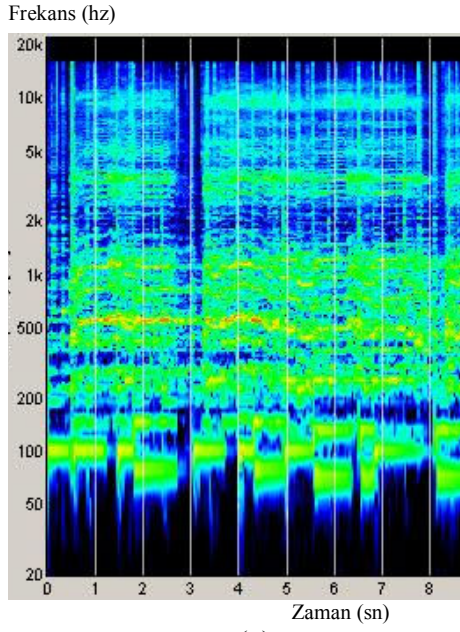
STKK yazılımıyla içerisinde bir alt klasör ile resim ve doküman yer alan bir klasör 6 adet wav dosyası içerisine, belirlenen bir anahtar yardımıyla, AES algoritmasıyla şifrelenerek rastgele saklanmıştır. Saklama işlemi sonucu oluşan stego ses dosyaları işitsel ve istatistiksel açıdan analiz edilmiş olup, işletim sistemi özellikleri yönüyle de taşıyıcı dosyalarla karşılaştırılmıştır.

Saklama sonucu oluşan tüm dosyalar işitsel olarak karşılık gelen orijinalinden farksızdır. Yani, taşıyıcı ses dosyasında değişen ses örnekleri işitsel olarak anlaşılabilir bir fark oluşturmamıştır.

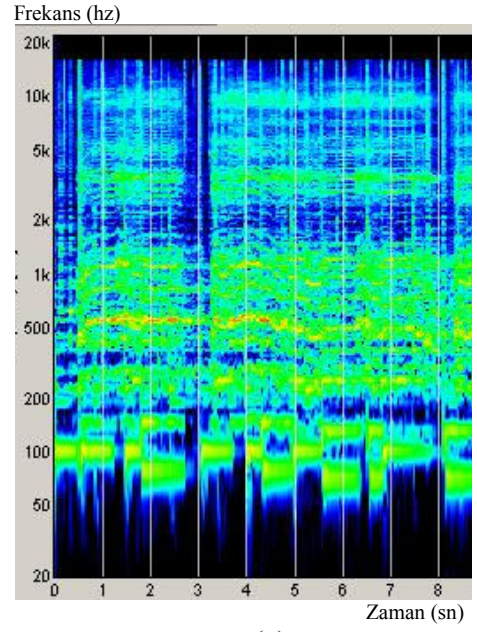
Bunun yanı sıra, stego dosyalar ile taşıyıcı dosyalara ait spektrogram grafikleri Şekil 6.7’de verilmiş olup, grafikler arasında da anlaşılabilir bir fark oluşmadığı gözlemlenmiştir. İşletim sistemi açısından da stego dosyalar ile karşılık gelen taşıyıcı dosyaları aynı fiziksel boyuta sahiptir. Şekil 6.7’ de verilen spektrogram grafiklerinin ayırt edilemez olması, oluşan stego dosyaların sadece işitsel steganaliz ataklarına karşı değil, aynı zamanda istatistiksel steganaliz ataklarına karşı da dayanıklı olacağını göstermektedir.



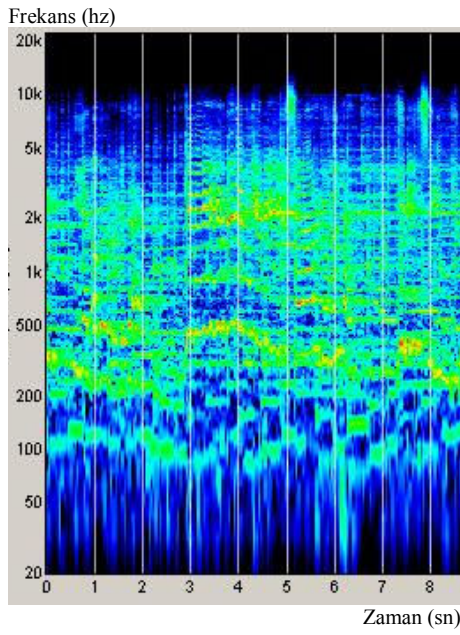
Şekil 6.7. Taşıyıcı ve stego ses dosyalarına ait spektrogram grafikleri a) 1. taşıyıcı ses dosyası b) 1. stego ses dosyası



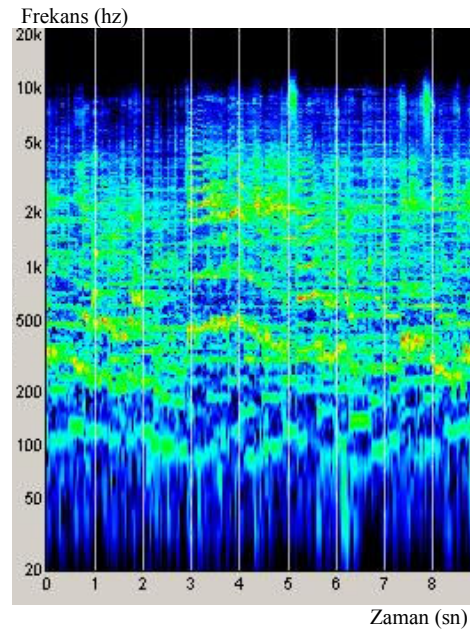
(c)



(ç)

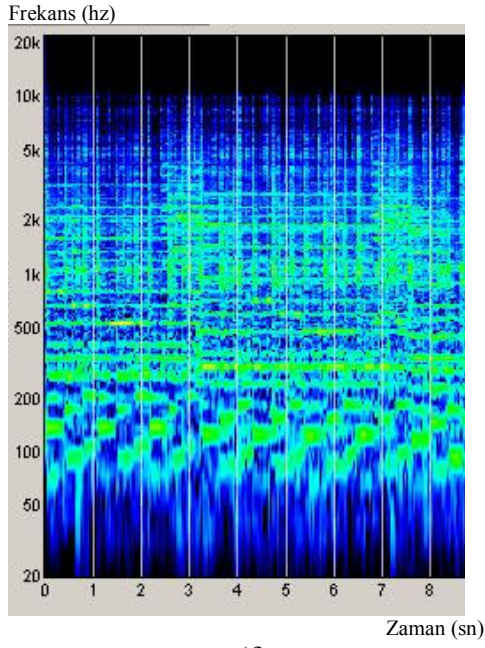


(d)

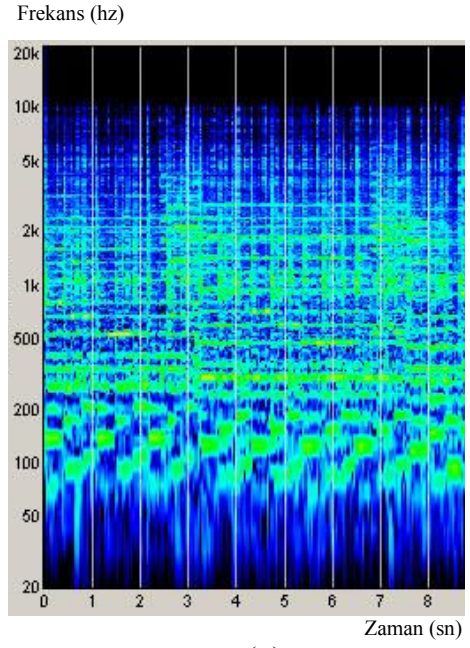


(e)

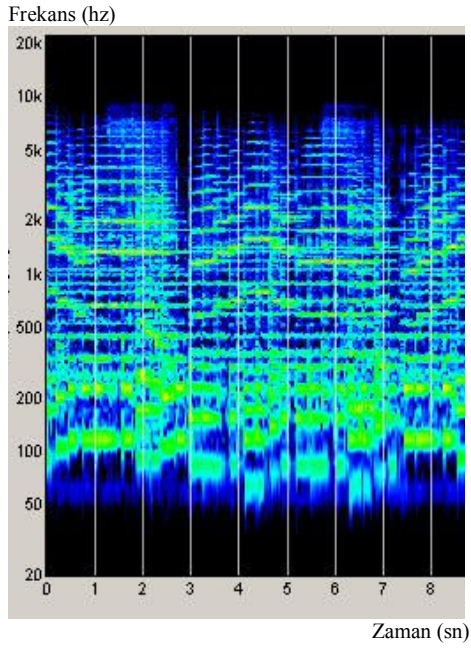
Şekil 6.7. (Devam) Taşıyıcı ve stego ses dosyalarına ait spectrogram grafikleri
 c) 2. taşıyıcı ses dosyası ç) 2. stego ses dosyası d) 3. taşıyıcı ses dosyası
 e) 3.stego ses dosyası



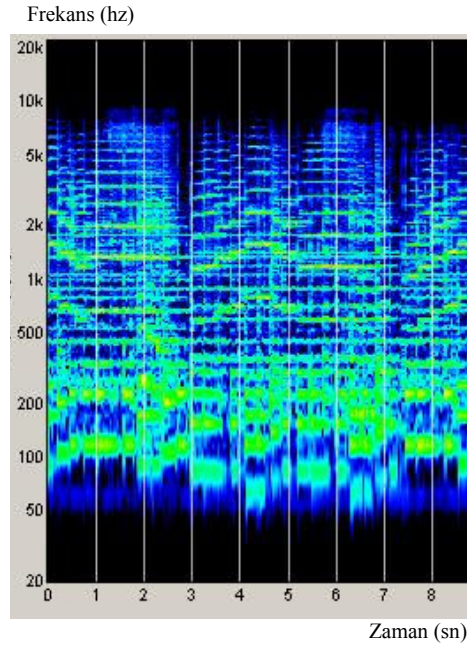
(f)



(g)

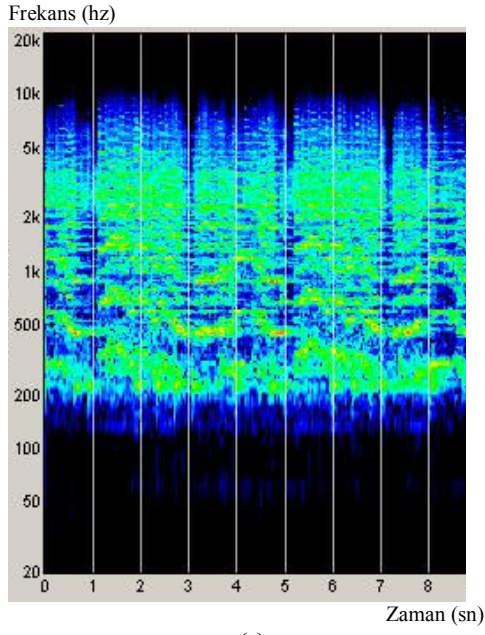


(ğ)

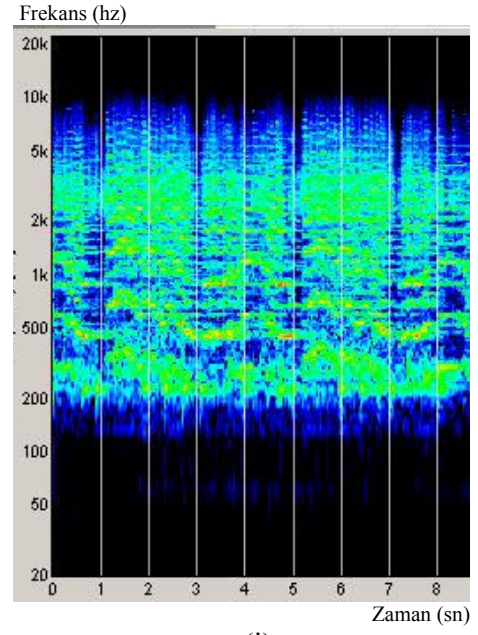


(h)

Şekil 6.7. (Devam) Taşıyıcı ve stego ses dosyalarına ait spectrogram grafikleri
 f) 4. taşıyıcı ses dosyası g) 4. stego ses dosyası ğ) 5. taşıyıcı ses dosyası
 h) 5. stego ses dosyası



(1)



(i)

Şekil 6.7. (Devam) Taşıyıcı ve stego ses dosyalarına ait spectrogram grafikleri
 1) 6. taşıyıcı ses dosyası i) 6. stego ses dosyası

7. SONUÇ VE ÖNERİLER

Bu tez çalışmasında LSB modifikasyonu yöntemiyle ses içerisine değişik bit seviyelerinde veri saklama işlemleri gerçekleştirilerek bunların sonuçları histogram çizelgeleri ve işitsel yöntemlerle analiz edilmiş ve bir stego tabanlı klasör gizleme yazılımı geliştirilmiştir. Ayrıca yapılan literatür araştırmasıyla veri saklama işleminin başarısını etkileyen ve steganaliz ataklarına karşı dayanıklılığı arttıran etkenler belirlenerek geliştirilen yazılıma bu etkenler mümkün olduğunca eklenmeye çalışılmıştır.

7.1. Sonuçlar

Bu çalışmada ulaşılan sonuçlar aşağıda özetlenmiştir.

- 16 bitlik ses örneklerinden oluşan WAV formatındaki ses dosyaları steganografide veri saklama amacıyla kullanılabilmekte ve yüksek saklama kapasitesi sunabilmektedir.
- 1. seviye ile 8. seviye arasındaki tüm bit seviyelerine yapılan saklama işlemi işitsel olarak fark edilir bir gürültü oluşturmamakta, şeffaf bir şekilde saklama yapılabilmektedir.
- Saklama işlemi sonucu şeffaflığın bozulması konusu taşıyıcı ses dosyasının ses kalitesiyle de ilgilidir. Kalitesi iyi olmayan ses dosyalarında yapılan saklamalarda 9. seviyede yapılan saklama işleminde de işitsel olarak hissedilir fark oluşmamaktadır.
- En az öneme sahip bit seviyesinde yani 1. seviyede yapılan saklama işleminde taşıyıcı bitlerin tamamı değişse bile işitsel olarak, oluşan stego ses dosyasının şeffaflığı bozulmamaktadır.
- Taşıyıcı olarak kullanılacak ses dosyasının orijinali kolay elde edilebilir dosyalar olmaması gerekir. İşitsel steganaliz ataklarına karşı bu gerekli olabilir.
- Saklama oranı, saklama kapasitesini düşürecek olmasına rağmen istatistiksel steganaliz karşısındaki dayanıklılığı arttırması açısından yeterince düşük tutulmalıdır.

- Saklama işlemi ardışık değil belirli bir anahtar değerine göre rastgele yapılmalıdır.

7.2. Öneriler

Tez çalışması kapsamında gerçekleştirilen stego tabanlı klasör gizleme yazılımı bir anahtar aracılığıyla WAV dosya seti içerisine rastgele saklama yapmakta ve saklama kapasitesi ile şeffaflık arasındaki ters orantılı ilişkiye dolaylı bir çözüm sunmaktadır. Rastgele saklama yapılmasının steganaliz ataklarına karşı dayanıklılığı arttırması beklenmekte, anahtar kullanılması ise saklama işlemini algoritmanın bilinmesinden bağımsız hale getirmektedir. Birden fazla WAV dosyası içerisine saklama yapması ise saklama kapasitesine dolaylı çözüm sağlamaktadır. Yazılımın eksik kısmı, saklama sonucu oluşan değişimi azaltmaya yani şeffaflığı ve steganaliz karşısındaki sağlamlığı arttırmaya yönelik eniyileme yapmamasıdır. İleriki çalışmalarda bir eniyileme algoritması olan genetik algoritmanın da yazılıma entegre edilmesiyle istatistiksel steganalize daha dayanıklı bir steganografi uygulaması gerçekleştirilebileceği düşünülmektedir.

Tüm dünyada olduğu gibi ülkemizde de steganografi konusu henüz tam olarak gereken ilgiyi görmemektedir. Genelde steganografi özelde ise ses içerisine veri saklama uygulamalarında henüz yeterli ilerleme sağlanamamıştır. Veri transferi teknikleri hızlandıkça resim dosyalarına göre daha büyük boyutlu olan ses dosyalarının da sayısal iletişimde çok yoğun bir şekilde kullanılacağı düşünülmektedir. Bu çalışmada en yaygın ses dosyası türlerinden birisi olan PCM biçimindeki WAV dosyaları kullanılmıştır ve gerçekleştirilen yazılımın ülkemizde gerçekleştirilecek ses tabanlı steganografik uygulamalara temel oluşturarak katkı sağlayacağı değerlendirilmektedir.

Kriptografi uygulamalarına benzer şekilde steganografi uygulamalarında da özgünlük çok önemlidir. İnternet üzerinde birçok ücretsiz steganografi yazılımı yer almasına veya ticari amaçlı steganografik uygulamalar bulunmasına rağmen bunları

kullanmak yerine özgün steganografik uygulamalar gerçekleştirilmesinin çok kritik olduğu düşünülmektedir.

İçinde yaşadığımız çağda bilgi çok kıymetli bir varlık haline gelmiştir. Gerek kurumlar gerekse kişiler için bilginin korunması yaşamsal öneme sahiptir. Bu tez çalışmasında gerçekleştirilen ses tabanlı yazılımın, özgün steganografik uygulamalar geliştirme konusunda ülkemizde duyulan kişisel ve kurumsal ihtiyacı karşılamaya yönelik öncül adımlardan birisi olacağı öngörülmektedir.

KAYNAKLAR

1. Adli, A.; Nakao, Z., "Three steganography algorithms for MIDI files," ***Machine Learning and Cybernetics, 2005. Proceedings of 2005 International Conference***, Guangzhou, China, 4 : 2401- 2404 (2005).
2. Akleyek, S. Nuriyev, U., "Steganography and new implementation of steganography", ***Signal Processing and Communications Applications Conference, 2005. Proceedings of the IEEE 13th***, Kayseri, TURKEY, 64-67 (2005).
3. Al-Hamami, A.H., Al-Ani, S.A., "A New Aproach for Authentication Technique", ***Journal Of Computer Science Science Publications***, 1 (1): 103-106 (2005).
4. Amin, M.M., Salleh, M., Ibrahim, S., Katmin, M.R., Shamsuddin, M.Z.I., "Information hiding using steganography," ***Telecommunication Technology, 2003. NCTT 2003 Proceedings. 4th National Conference***, Shah Alam, MALAYSIA, 21- 25 (2003).
5. Anderson,R.J., Petitcolas, F.A.P., "On The Limits of Steganography", ***IEEE Journal of Selected Areas in Communications***, 16(4):474-481 (1998).
6. Aura, T., "Practical Invisibility in Digital Communication", ***First Workshop of Information Hiding Proceedings***, Cambridge, pp 14 (1996).
7. Avcıbaşı, İ., "Audio Steganalysis With Content Independent Distortion Measures," ***IEEE Signal Processing Letters***, 13(2): 92-95 (2006).
8. Avcıbaşı, İ., M. Kharrazi, N. Memon, B. Sankur, "Image Steganalysis With Binary Similarity Measures," ***EURASIP Journal on Applied Signal Processing***, 2005(17): 2749-2757 (2005).
9. Avcıbaşı, İ., N. Memon, B. Sankur, "Steganalysis Using Image Quality Metrics," ***IEEE Transactions on Image Processing***, 12(2): 221-229 (2003).
10. Bender, W., Gruhl, D., Morimoto, N., and Lu, A. "Techniques for data hiding". ***IBM Syst. J.***, 35(3&4):313-336 (1996).
11. Brisbane, G., Safavi-Naini, R., Ogunbona, P., "High-capacity steganography using a shared colour palette," ***Vision, Image and Signal Processing, IEE Proceedings***, 152: 787- 792 (2005).
12. Caldwell, 2nd Lt. J., "Steganography", ***CROSSTALK The Journal of Defense Software Engineering***, 25-27 (2003).

13. Canbek, G., Sağiroğlu, Ş., “Bilgi ve Bilgisayar Güvenliği Casus Yazılımlar ve Korunma Yöntemleri”, **Grafiker**, Ankara, 1-50 (2006).
14. Chou, J., Ramchandran, K., Ortega, A., "High capacity audio data hiding for noisy channels," **Information Technology: Coding and Computing, 2001. Proceedings. International Conference**, Las Vegas, 108-112 (2001).
15. Cvejic, N., Seppanen, T., "Increasing the capacity of LSB-based audio steganography," **Multimedia Signal Processing, 2002 IEEE Workshop** , St. Thomas, Virgin Islands, USA , 336- 338 (2002).
16. Farid, H., “Detecting steganographic message in digital images”. **Technical Report TR2001-412, Dartmouth College**, 1-9 (2001).
17. Fridrich, J., Goljan, M., Du, R., "Reliable Detection of LSB Steganography in Color and Grayscale Images", **Proc. of the ACM Workshop on Multimedia and Security**, Ottawa, Canada, 27-30 (2001).
18. Fridrich, J., Goljan, M., “Practical steganalysis of digital images state of the art.”, **Proc. SPIE Photonics West**, 4675:1–13 (2002).
19. Fridrich, J., Goljan, M., Rui Du, "Detecting LSB steganography in color, and gray-scale images," **Multimedia, IEEE** , 8(4):22-28 (2001).
20. Fridrich, J., Du, R., Long, M., "Steganalysis of LSB encoding in color images," **Multimedia and Expo, 2000. ICME 2000. 2000 IEEE International Conference** , New York, USA , 3:1279-1282 (2000).
21. Fridrich, J., “Minimizing the embedding impact in steganography”, **Proceeding of the 8th Workshop on Multimedia and Security**, Geneva-Switzerland, 2-10 (2006).
22. Gopalan, K., "Audio steganography using bit modification," , **2003 International Conference on Multimedia and Expo**, Baltimore, Maryland, 629-632 (2003).
23. İnternet: Microsoft WAVE soundfile format
<http://ccrma.stanford.edu/CCRMA/Courses/422/projects/WaveFormat/>
24. İnternet: Chmura, J., Haller, B., “Steganography”, EECS 600 – 2004.
http://www.jchmura.com/presentations/jlc_bah_steganography.pdf
25. İnternet: Cummins, J., Diskin, P., Lau, S., Parlett, R., “ Steganography and Digital Watermarking”, 2004.
<http://www.cs.bham.ac.uk/~mdr/teaching/modules03/security/students/SS5/Steganography.pdf>

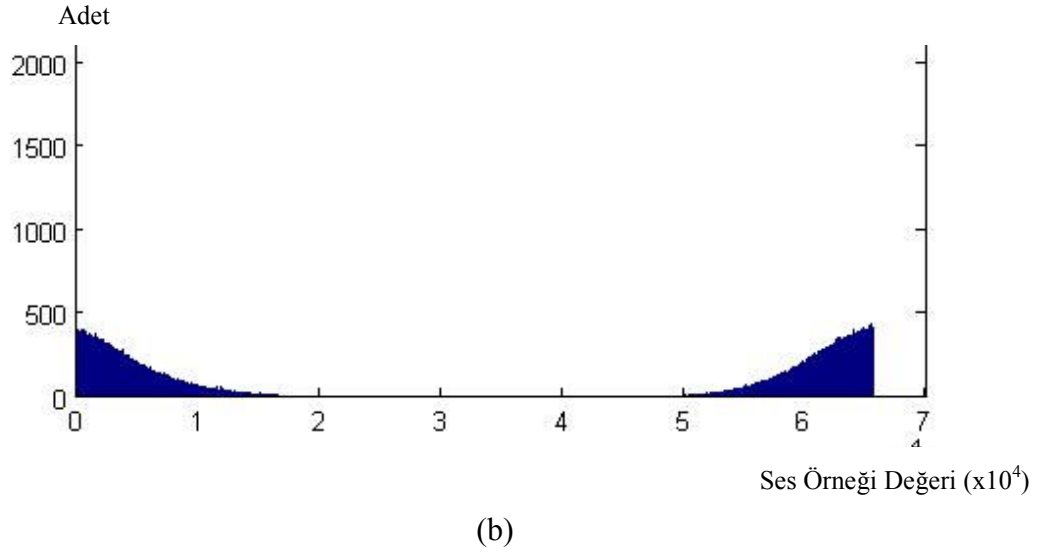
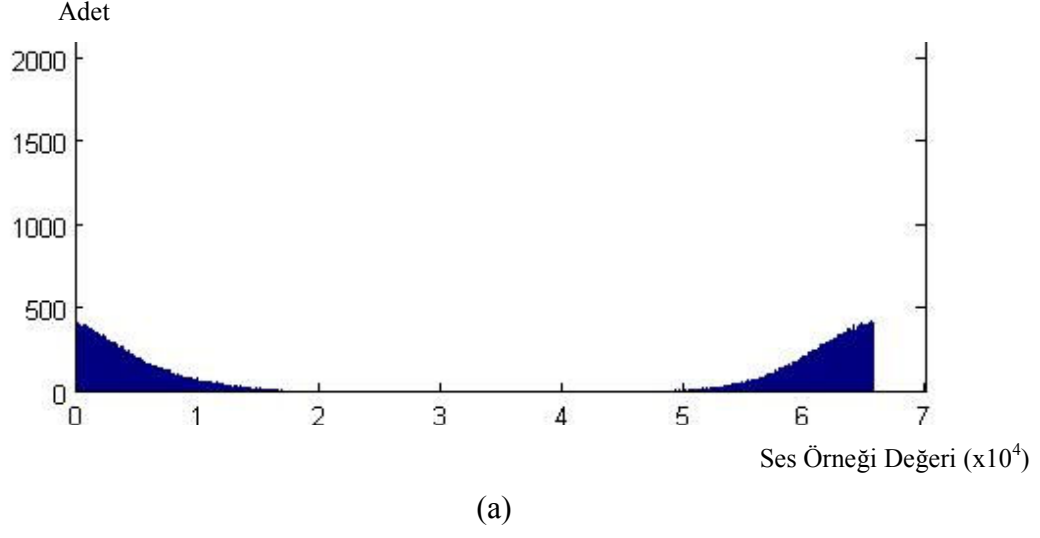
26. İnternet: Michaud, E., “Current Steganography Tools and Methods.”, GSEC Practical, Version 1.4b, 2003.
http://www.giac.org/certified_professionals/practicals/gsec/2760.php
27. İnternet: spammimic - hide a message in spam,
<https://www.spammimic.com/>
28. İnternet: XML Introduction - What is XML?,
http://www.w3schools.com/xml/xml_what.asp
29. İnternet: mp3stego, <http://www.petitcolas.net/fabien/steganography/mp3stego>
30. İnternet: El-Khalil, R., Keromytis, A.D., “Hydan: Hiding Information in Program Binaries” <http://www1.cs.columbia.edu/~angelos/Papers/hydan.pdf>
31. İnternet: Johnson, N.F., Jajodia, S., “Steganalysis of Images Created Using Current Steganography Software”, <http://www.jjtc.com/ihws98/jjgmu.html>
32. İnternet: “Veri Gizle” www.verigizle.com
33. İnternet: “Steganografik Kütüphane”, Erkin, Z., Örencik, B.,
http://www.emo.org.tr/resimler/etkinlikbildirileri/d8df73a3cfbf3c5_ek.pdf
34. İnternet: OutGuess - universal Steganography,
<http://www.outguess.org/>
35. İnternet: StegoArchive.Com - Steganography Information, Software, and News to Enhance Your Privacy, <http://www.stegoarchive.com/>
36. İnternet: The SNOW Home Page,
<http://www.darkside.com.au/snow/index.html>
37. İnternet: Freeware Italiano by Freeonline.it - Programmatore: Oliboni Cosimo, <http://www.freeonline.it/pr/pr-902/Puff>
38. İnternet: wbStego Steganography Tool, <http://wbstego.wbailer.com/>
39. İnternet: <http://secureengine.isecurelabs.com/>
40. İnternet: Hermetic Stego: Software to Hide a File in BMP Image Files,
<http://www.hermetic.ch/hst/hst.htm>
41. İnternet: The Gifshuffle Home Page
<http://www.darkside.com.au/gifshuffle/>
42. İnternet: Deogol - HTML Steganography Tool,
<http://wandership.ca/projects/deogol/intro.html#deogol>

43. Internet: Publimark 0.1.3, <http://gleguelv.free.fr/soft/publimark/>
44. Internet: Steghide, <http://steghide.sourceforge.net/>
45. Internet: SpyHunter <http://www.spy-hunter.com/stegspydownload.htm>
46. Internet: Stego Suite
<http://www.000.shoppingcartplus.com/catalog/item/4170630/4050552.htm>
47. Internet: WetStone Technologies Inc.
http://www.wetstonetech.com/f/faq_stego.html
48. Internet: OutGuess – Steganography Detection
<http://www.outguess.org/detection.php>
49. Jamil, T., "Steganography: the art of hiding information in plain sight," *Potentials, IEEE* , 18(1):10-12 (1999).
50. Johnson, N.F., Jajodia, S., "Exploring Steganography: Seeing the Unseen", *IEEE Computer*, 31(2):26-34 (1998).
51. Kaufman, C., Perlman, R., Speciner, M., "Network Security Private Communication in a Public World 2nd ed.", *Prentice Hall*, New Jersey, 62-65, 81-84, 109, 140-141 (2002).
52. Kessler, G.C., "An Overview of Steganography for the Computer Forensics Examiner", *Forensic Science Communications* , 6(3):1-29 (2004).
53. Lee, Y.K., Chen, L.H., "High capacity image steganographic model," *Vision, Image and Signal Processing, IEE Proceedings-* , 147(3):288-294 (2000).
54. Paulson, L.D., "New System Fights Steganography.", *IEEE Computer Society*, 39(8):25 (2006).
55. Niimi, M., Noda, H., Kawaguchi, E., Eason, R.O., "High capacity and secure digital steganography to palette-based images," *Image Processing 2002. Proceedings 2002 International Conference* , Rochester, New York, USA, 2: 917-920 (2002).
56. Noda, H., Spaulding, J., Shirazi, M.N., Kawaguchi, E., "Application of bit-plane decomposition steganography to JPEG2000 encoded images" *Signal Processing Letters, IEEE* 9(12):410-413 (2002).
57. Özer, H., Sankur, B., Memon, N., Avcıbaşı, İ., "Detection Of Audio Covert Channels Using Statistical Footprints Of Hidden Messages.", *Digital Signal Processing* 16 (4): 389-401 (2006).

58. Provos, N., Honeyman, P., "Detecting steganographic content on the internet," *Tech. Rep. CITI 01-1a, University of Michigan*, 1-14 (2001).
59. Rabah, K., "Steganography-The Art of Hiding Data.", *Information Technology Journal*, 3 (3):245-269 (2004).
60. Ru et al., "Audio Steganalysis Based On 'Negative Resonance Phenomenon' Caused By Steganographic Tools.", *Journal of Zhejiang University Science A*, 7(4):577-583 (2006).
61. Sağiroğlu, Ş., Alkan, M., "Her Yönüyle Elektronik İmza (e-İmza)", *Grafiker*, Ankara, 3, 5, 33 (2005).
62. Sağiroğlu, Ş., Tunçkanat, M., "A Secure Internet Communication Tool", *Turkish Journal of Telecommunications*, 1(1):40-46 (2002).
63. Sankur B., N. Memon, İ. Avcıbaş, "Automatic Detection of the Presence of Stegosignals and Watermarks in Images," *European Conference on Visual Perception*, Antalya, 38 (2001).
64. Shahreza, S., "Stealth steganography in SMS" *Wireless and Optical Communications Networks, 2006 IFIP International Conference*, Bangalore, 5- (2006).
65. Srinivasan, Y., Nutter, B., Mitra, S., Phillips, B., Ferris, D., 2004. "Secure Transmission of Medical Records Using High Capacity Steganography". *Proceedings of the 17th IEEE Symposium on Computer-Based Medical Systems (CBMS'04)*, Bethesda, Maryland, 122-127 (2004).
66. Sui, X.G., Luo, H., "A new steganography method based on hypertext" *Radio Science Conference, 2004. Proceedings. 2004 Asia-Pacific*, Qingdao, China , 181-184 (2004).
67. Swanson, M.D., Zhu, B., Tewfik, A.H., "Robust Data Hiding For Images", *In 7th Digital Signal Processing Workshop (DSP 96)*, Loen Norway, 37-40 (1996).
68. Tseng, H.W., Chang, C.C., "Steganography using JPEG-compressed images" *Computer and Information Technology, 2004. CIT '04. The Fourth International Conference* , Wuhan, China , 12- 17 (2004).
69. Westfeld, A., Pfitzmann, A., "Attacks On Steganographic Systems", *Proceedings of the Third International Workshop on Information Hiding*, Dresden, Germany, 61-76 (1999).

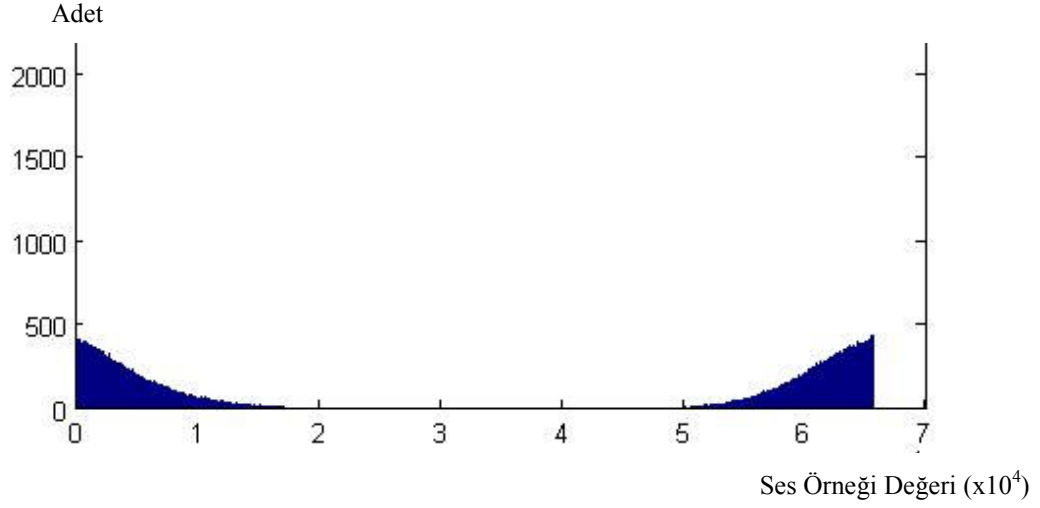
EKLER

EK-1 Stego ses dosyalarına ait histogram ve spectrogram grafikleri

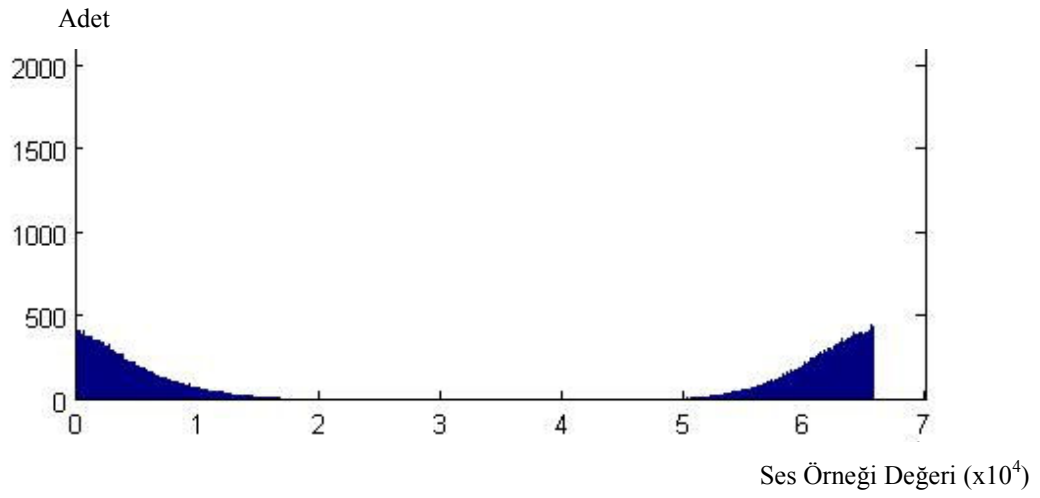


Şekil 1.1. Histogram çizelgeleri a) 2. seviyede veri saklı b) 3. seviyede veri saklı

EK-1 (Devam) Stego ses dosyalarına ait histogram ve spectrogram grafikleri



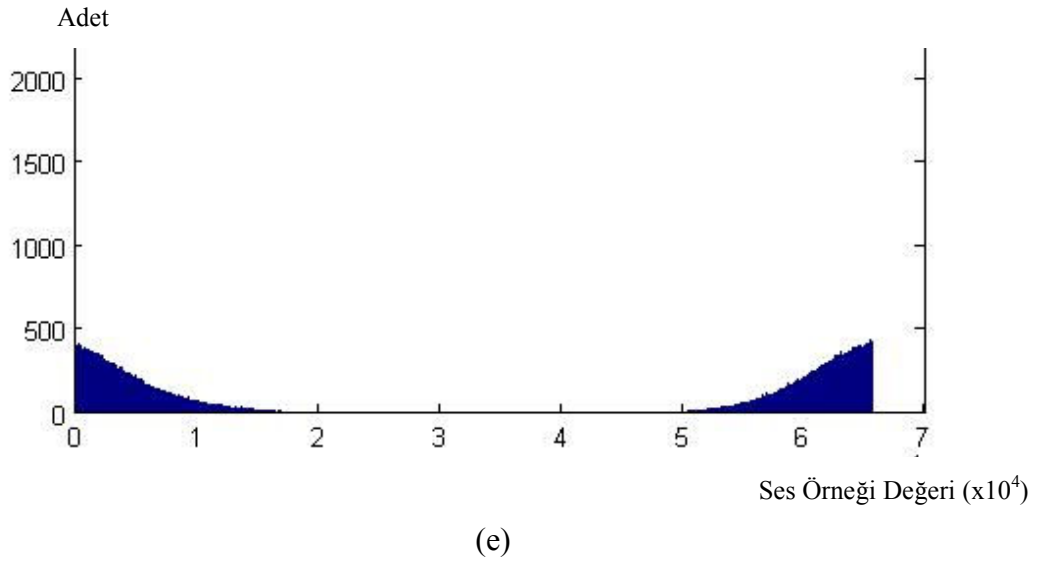
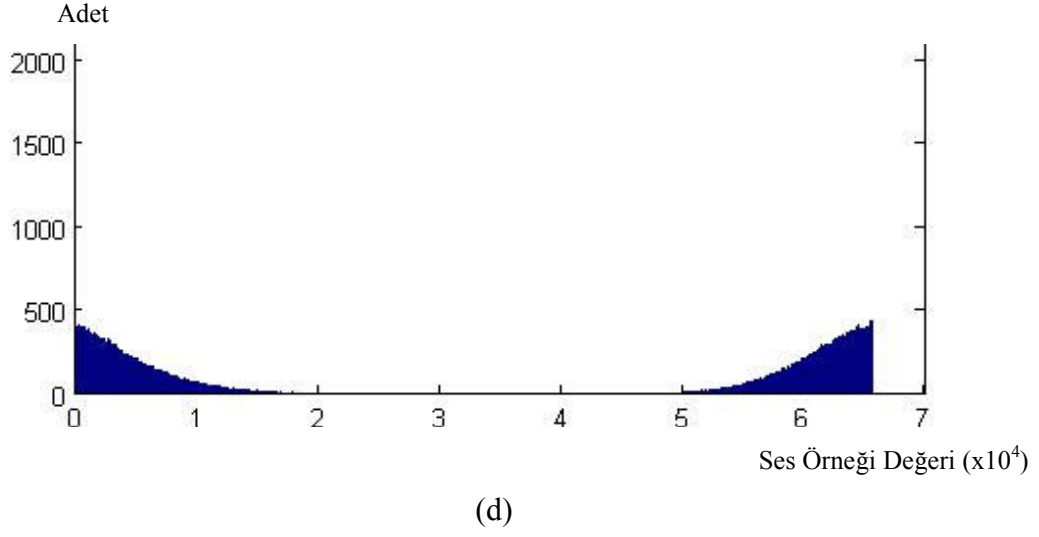
(c)



(ç)

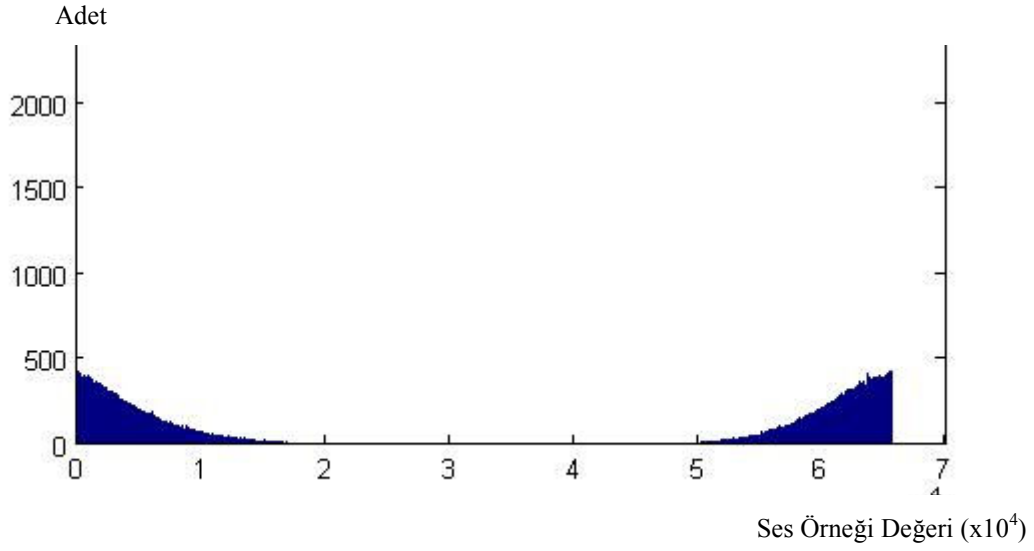
Şekil 1.1. (Devam) Histogram çizelgeleri c) 4. seviyede veri saklı ç) 5. seviyede veri saklı

EK-1 (Devam) Stego ses dosyalarına ait histogram ve spectrogram grafikleri

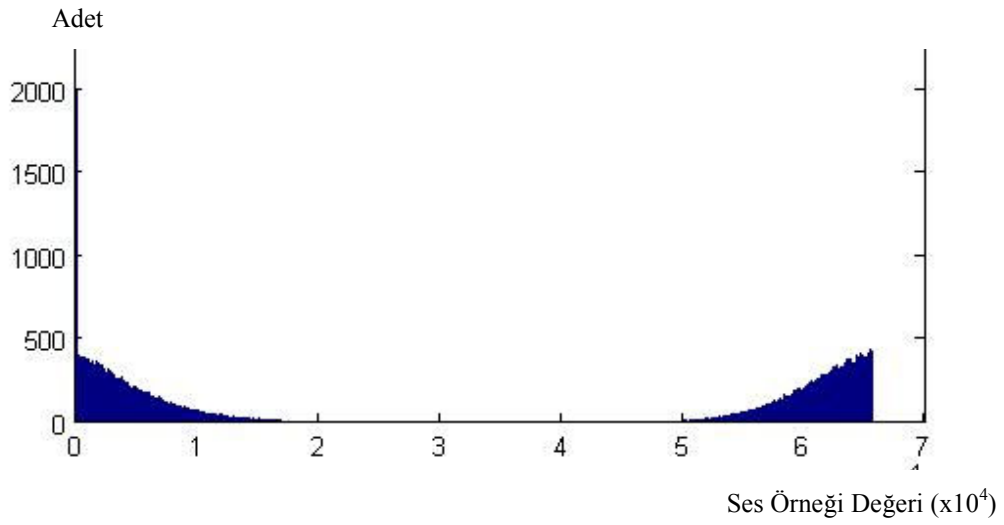


Şekil 1.1. (Devam) Histogram çizelgeleri d) 6. seviyede veri saklı e) 7. seviyede veri saklı

EK-1 (Devam) Stego ses dosyalarına ait histogram ve spectrogram grafikleri



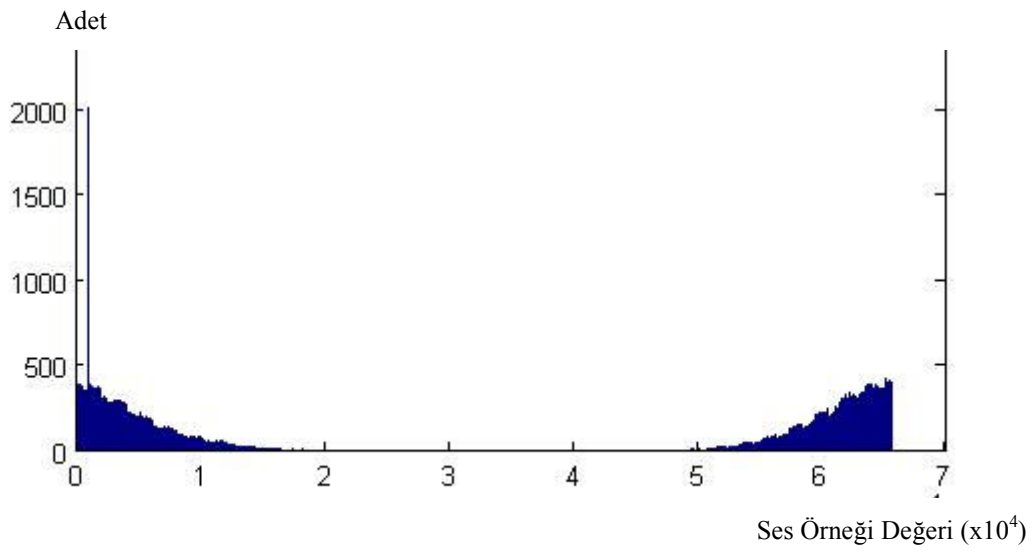
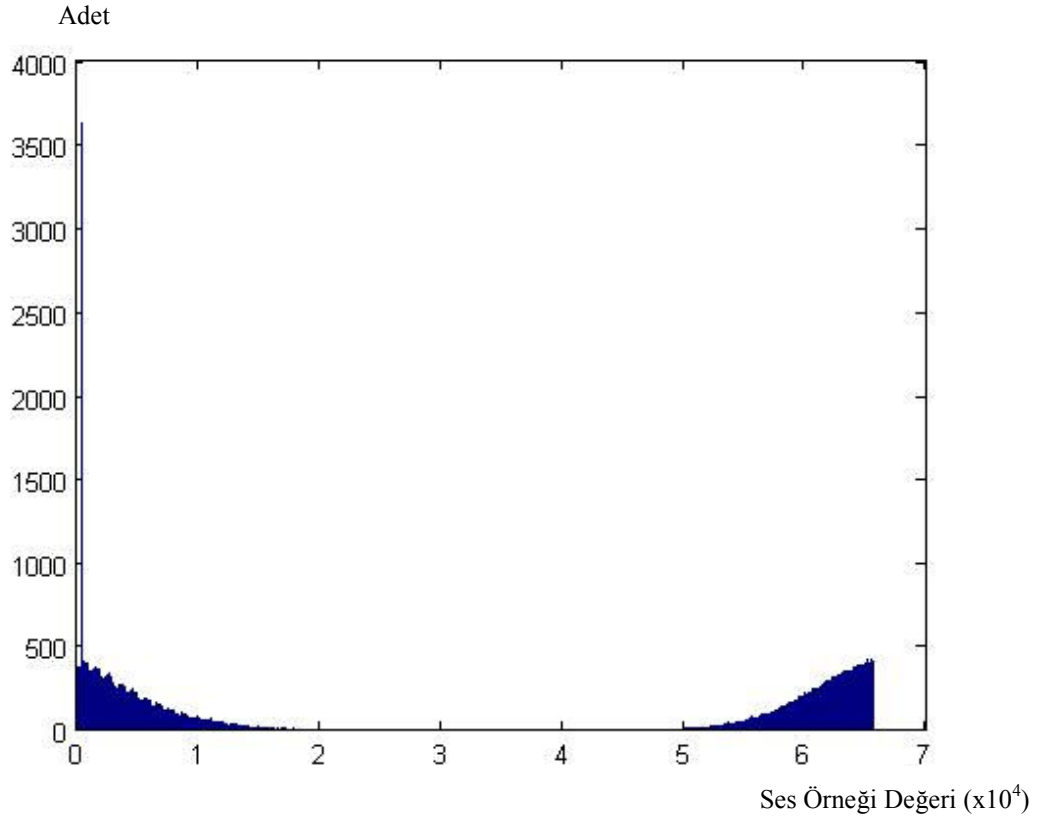
(f)



(g)

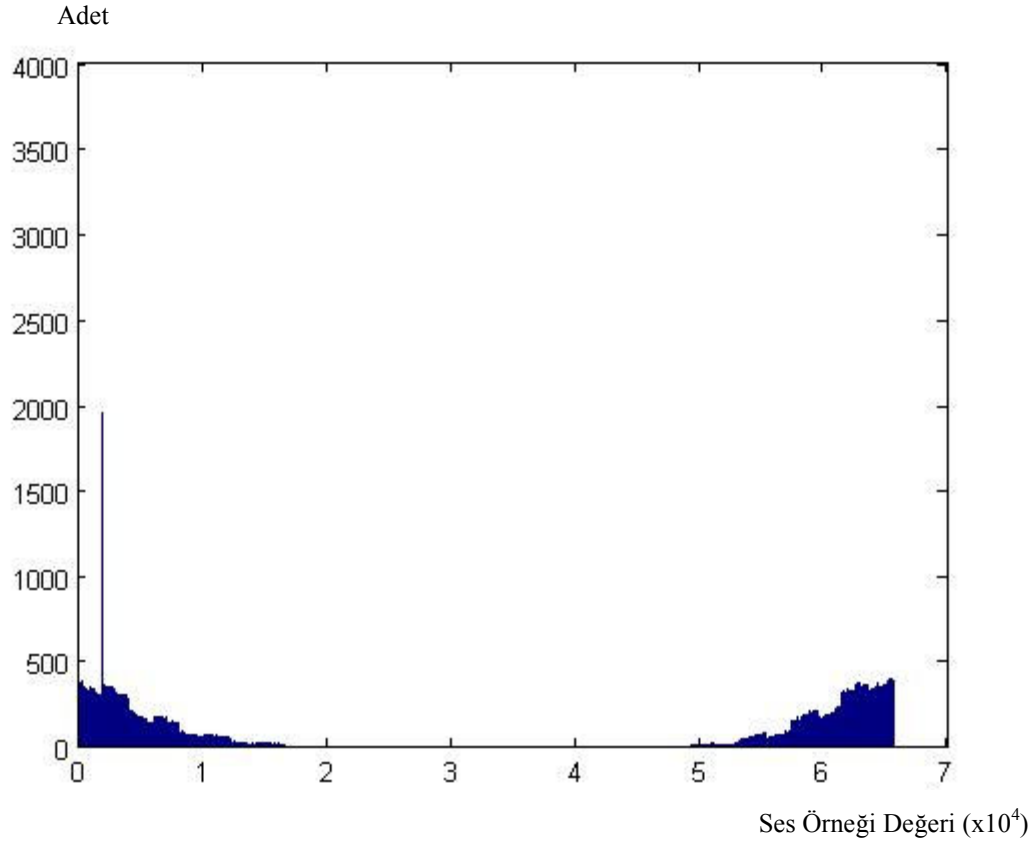
Şekil 1.1. (Devam) Histogram çizelgeleri f) 8. seviyede veri saklı g) 9. seviyede veri saklı

EK-1 (Devam) Stego ses dosyalarına ait histogram ve spectrogram grafikleri

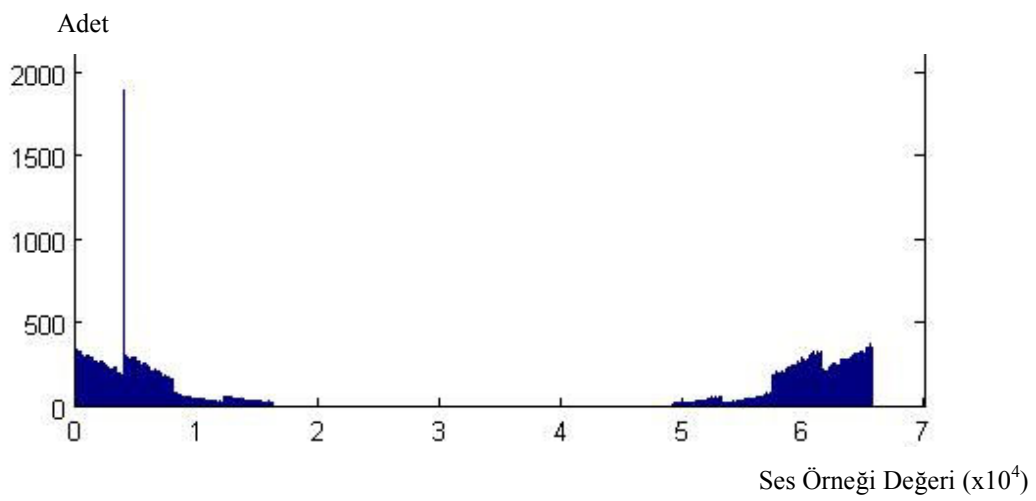


Şekil 1.1. (Devam) Histogram çizelgeleri ğ) 10. seviyede veri saklı h) 11. seviyede veri saklı

EK-1 (Devam) Stego ses dosyalarına ait histogram ve spectrogram grafikleri



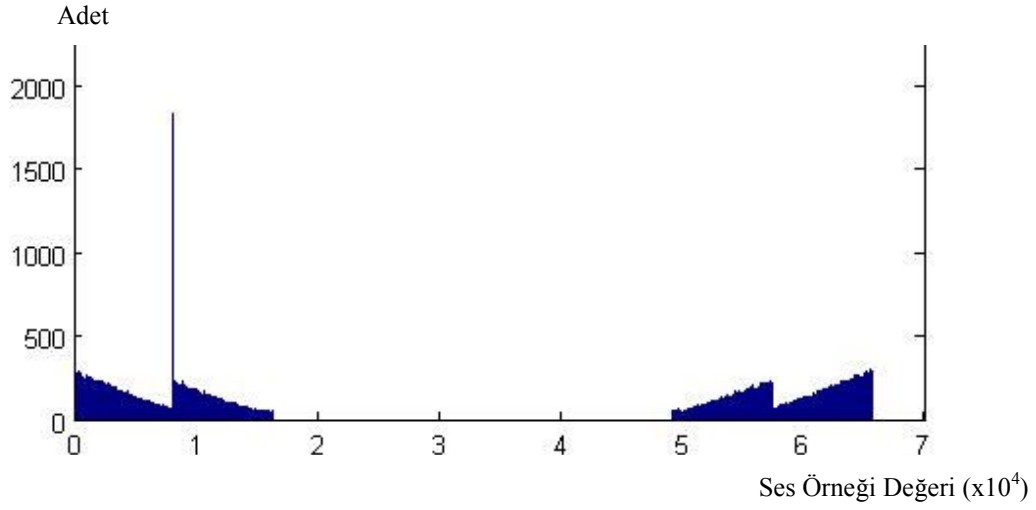
(i)



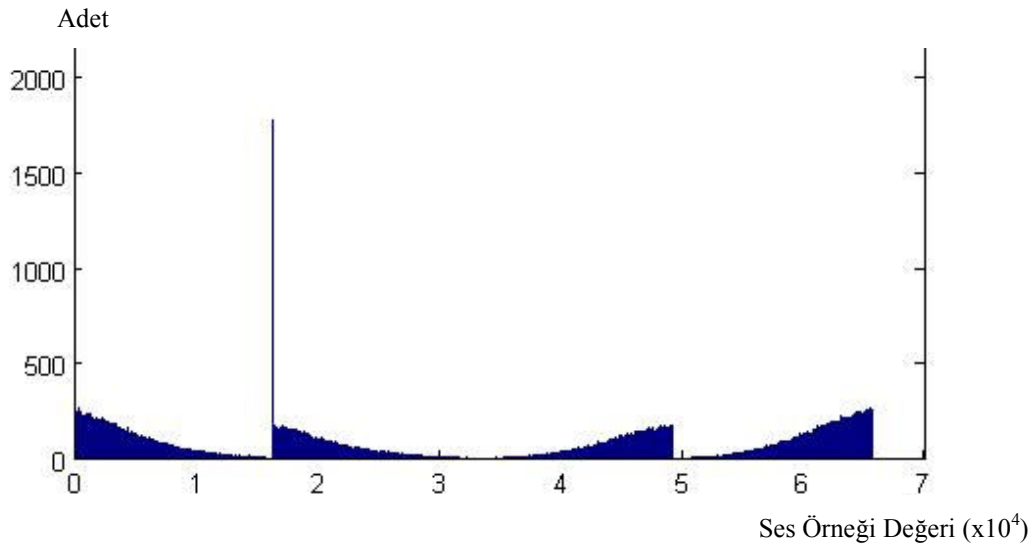
(i)

Şekil 1.1. (Devam) Histogram çizelgeleri i) 12. seviyede veri saklı i) 13. seviyede veri saklı

EK-1 (Devam) Stego ses dosyalarına ait histogram ve spectrogram grafikleri



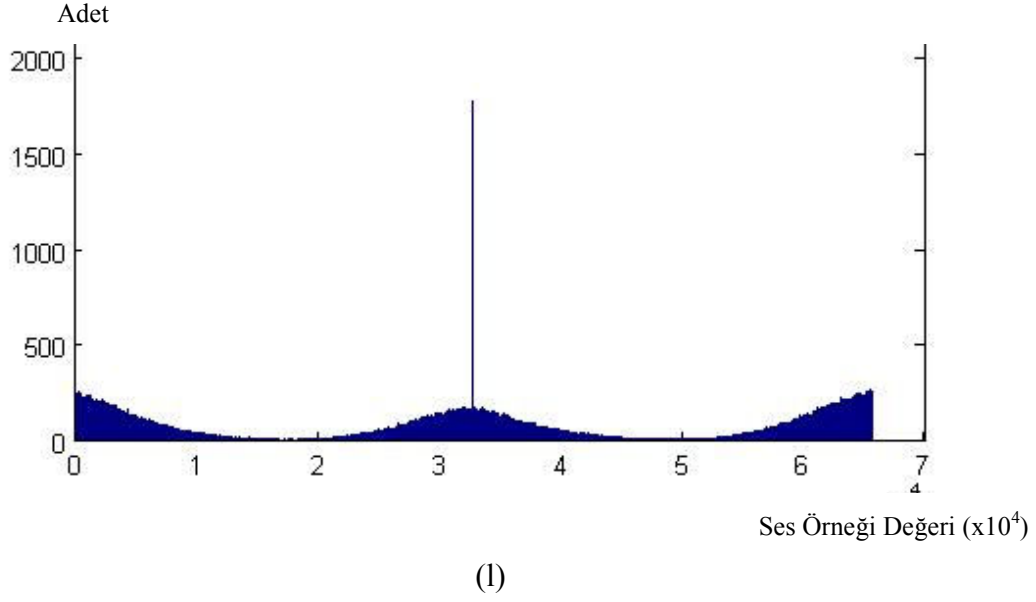
(j)



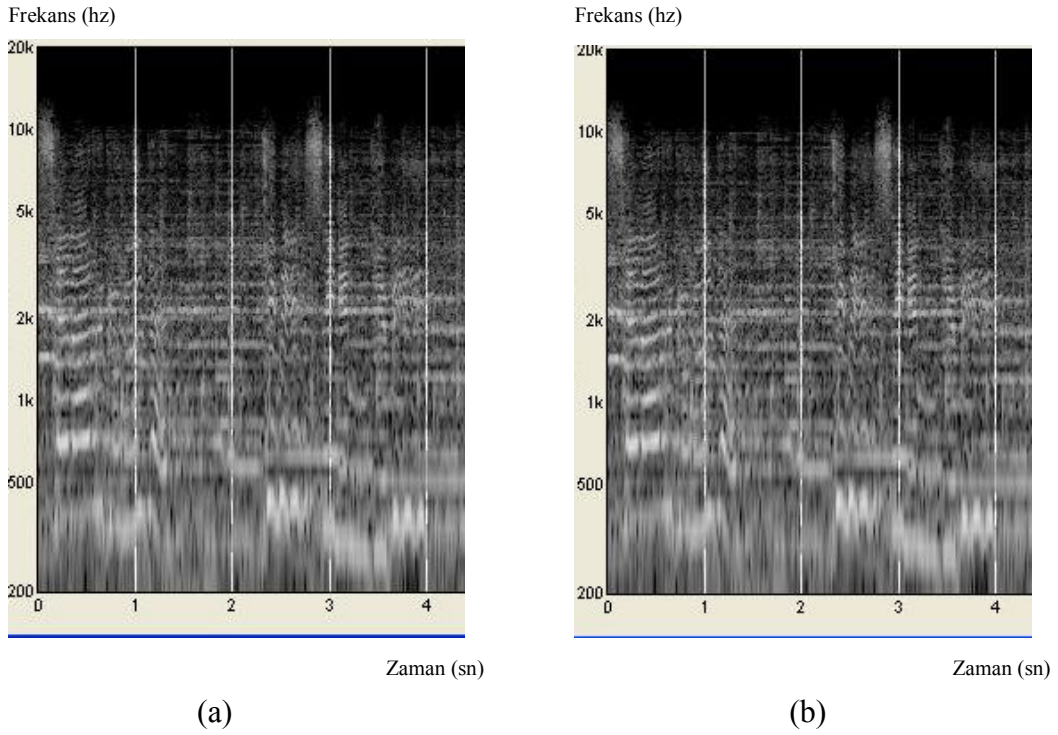
(k)

Şekil 1.1. (Devam) Histogram çizelgeleri j) 14. seviyede veri saklı k) 15. seviyede veri saklı

EK-1 (Devam) Stego ses dosyalarına ait histogram ve spectrogram grafikleri

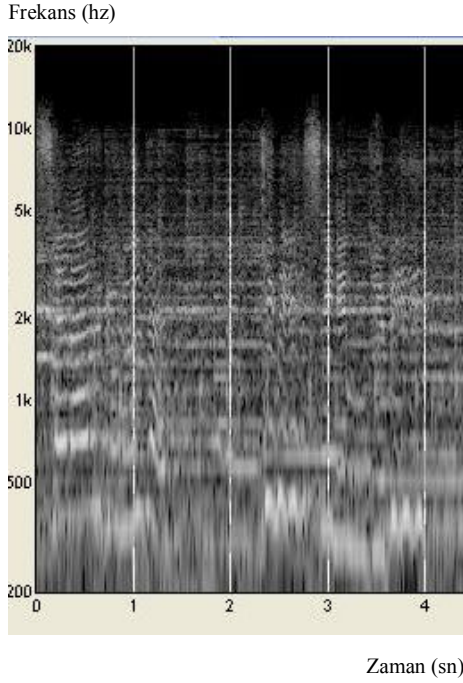


Şekil 1.1. (Devam) Histogram çizelgeleri I) 16. seviyede veri saklı

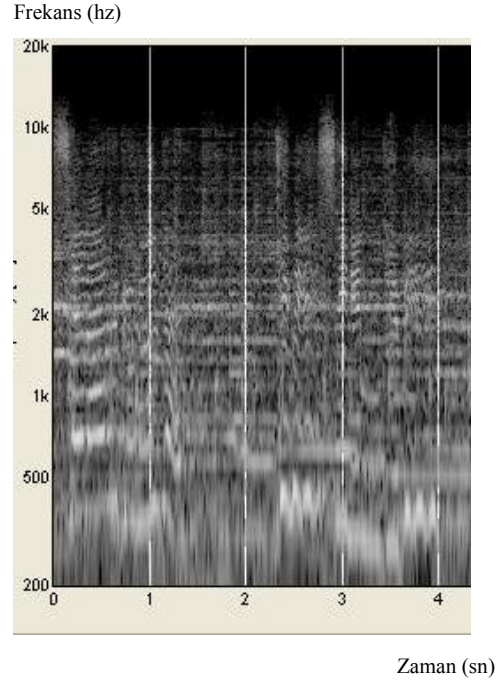


Şekil 1.2. Spectrogram grafikleri a-) 3. seviyede modifikasyon b-) 4. seviyede Modifikasyon

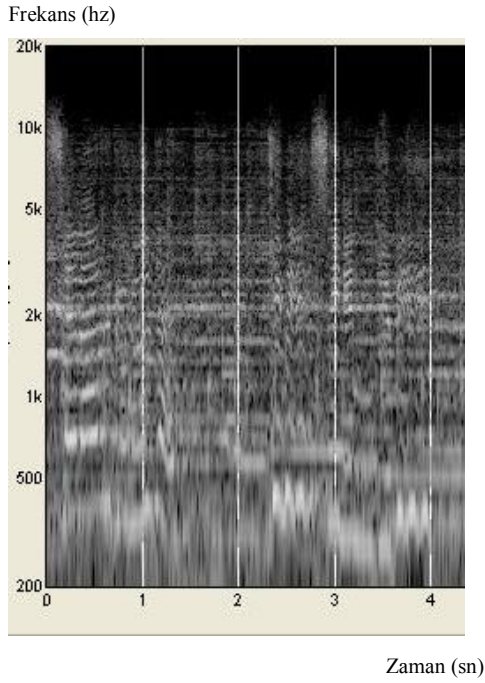
EK-1 (Devam) Stego ses dosyalarına ait histogram ve spectrogram grafikleri



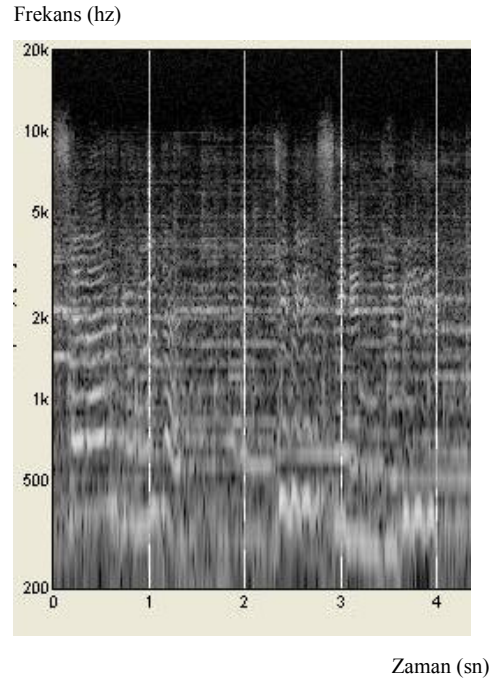
(c)



(ç)



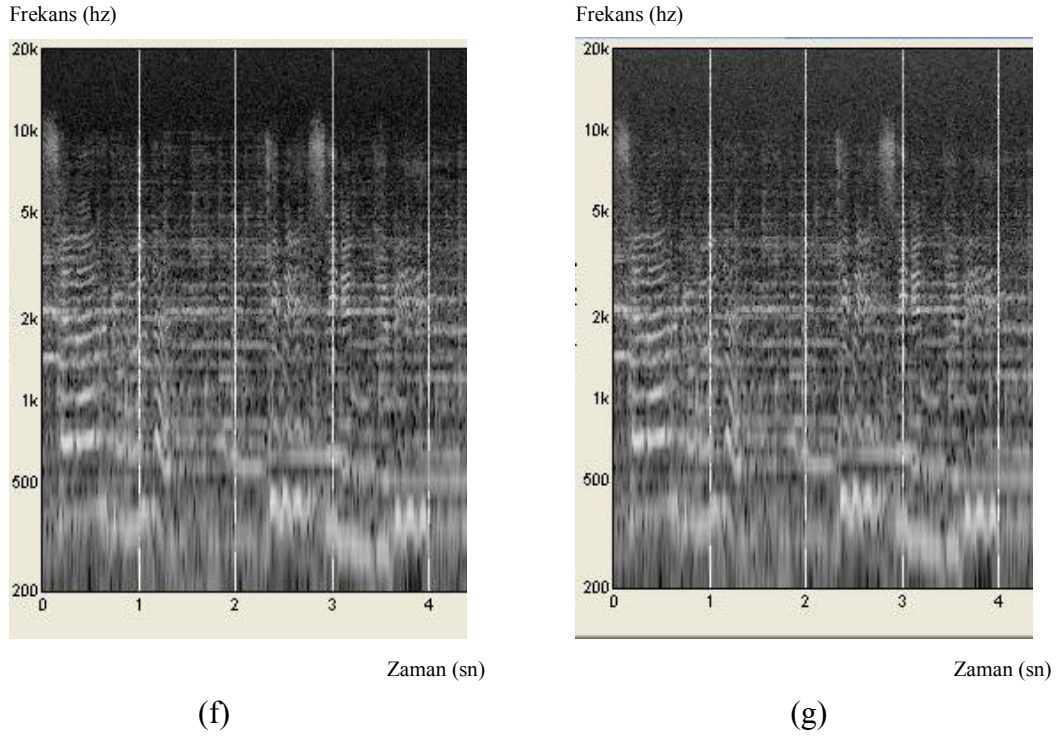
(d)



(e)

Şekil 1.2. (Devam) Spectrogram grafikleri c) 5. seviyede modifikasyon
ç) 6. seviyede modifikasyon d) 7. seviyede modifikasyon e) 8. seviyede modifikasyon

EK-1 (Devam) Stego ses dosyalarına ait histogram ve spectrogram grafikleri



Şekil 1.2. (Devam) Spectrogram grafikleri f) 9. seviyede modifikasyon
g) 10. seviyede modifikasyon

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, Adı : ATICI, Mehmet Ali.
Uyruğu : T.C.
Doğum Tarihi ve Yeri : 13.02.1980, Keşan
Medeni Hali : Bekar
Telefon : 0312 591 23 80
e-mail : ali.atici@nvi.gov.tr

Eğitim

Derece	Eğitim Birimi	Mezuniyet Yılı
Lisans	Bilkent Üni. Bilgisayar Müh.	2002
Lise	Gazi Çiftlik Lisesi	1997

İş Deneyimi

Yıl	Yer	Görev
2002-2002	İşlem Şirketler Grubu	Yazılım Mühendisi
2002-2004	M.E.B Konya	Bilgisayar Mühendisi
2004-	İçişleri B. NVİ GM.	Bilgisayar Mühendisi

Yabancı Dil

İngilizce

Hobiler

Masa Tenisi, Yürüyüş, Müzik