KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

KAOTİK HARİTALARIN STEGANOGRAFİ İLE BİRLİKTE KULLANIMI

YÜKSEK LİSANS TEZİ

Bilgisayar Mühendisi Esra ODABAŞ YILDIRIM

HAZİRAN 2014
TRABZON

KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

KAOTİK HARİTALARIN STEGANOGRAFİ İLE BİRLİKTE KULLANIMI

Bilgisayar Mühendisi Esra ODABAŞ YILDIRIM

Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsünce
"BİLGİSAYAR YÜKSEK MÜHENDİSİ"
Unvanı Verilmesi İçin Kabul Edilen Tezdir.

Tezin Enstitüye Verildiği Tarih : 23.05.2014
Tezin Savunma Tarihi : 27.06.2014

Tez Danışmanı : Doç. Dr. Mustafa ULUTAŞ

Trabzon 2014

Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsü

Bilgisayar Mühendisliği Anabilim Dalında

Esra ODABAŞ YILDIRIM tarafından hazırlanan

KAOTİK HARİTALARIN STEGANOGRAFİ İLE BİRLİKTE KULLANIMI

başlıklı bu çalışma, Enstitü Yönetim Kurulunun 03/06/2014 gün ve 1556 sayılı kararıyla oluşturulan jüri tarafından yapılan sınavda
YÜKSEK LİSANS TEZİ
olarak kabul edilmiştir.

Jüri Üyeleri

Başkan : Doç. Dr. Mustafa ULUTAŞ .....

Üye : Yrd. Doç. Dr. Hüseyin PEHLİVAN .....

Üye : Yrd. Doç. Dr. Önder AYDEMİR .....

Prof. Dr. Sadettin KORKMAZ

Enstitü Müdürü

ÖNSÖZ

Günümüzde internet birçok bilgisayar sisteminin birbirine bağlı olduğu dünya çapında yaygın olan ve sürekli büyüyen bir iletişim ağı haline gelmiştir. İnternetin bu kadar talep görmesinin nedenlerinden birisi bilgiyi saklamak, paylaşmak ve ona kolayca ulaşma isteğidir. İnternet sayesinde insanlar birbirleriyle iletişim kurabilme imkanı bulabilmektedir. Bu iletişim kurulurken bilgi sistemlerinin İnternet'e açılması beraberinde güvenlik problemini getirecektir. Bu güvenlik problemi için alınan önlemlerden birisi de bilginin şifrelenerek kötü niyetli kişilerin dikkatini çekmesi yerine gizli bilginin anlamlı hale getirilerek iletilmesi, bilginin başka bir ortamda gizlenmesi işlemi; steganografidir. Bu tezde Kaotik haritaların steganografi ile birlikte kullanımı gerçekleştirilmiştir.

Çalışmalarımda danışmanlığımı üstlenip benden ilgisini, bilgisini ve yardımlarını esirgemeyen danışman hocam sayın Doç. Dr. Mustafa Ulutaş'a sonsuz teşekkürü bir borç bilirim. Yüksek lisans süresince fikirlerine başvurduğum, beni her zaman tatlı tebessümüyle karşılayan hocam Yrd. Doç. Dr. Güzin Ulutaş'a da ayrıca teşekkür ederim. Tüm eğitim-öğretim hayatımda benden desteklerini esirgemeyen aileme, kardeşlerime ve her daim yanımda olan sevgili eşime teşekkür ederim.

Esra ODABAŞ YILDIRIM
Trabzon, 2014

TEZ BEYANNAMESİ

Yüksek Lisans Tezi olarak sunduğum "Kaotik Haritaların Steganografi ile Birlikte Kullanımı" başlıklı bu çalışmayı baştan sona kadar danışmanım Doç. Dr. Mustafa ULUTAŞ'ın sorumluluğunda tamamladığımı, verileri kendim topladığımı, analizleri ilgili laboratuvarlarda yaptığımı, başka kaynaklardan aldığım bilgileri metinde ve kaynakçada eksiksiz olarak gösterdiğimi, çalışma sürecinde bilimsel araştırma ve etik kurallara uygun olarak davrandığımı ve aksinin ortaya çıkması durumunda her türlü yasal sonucu kabul ettiğimi beyan ederim. 23/05/2014

Esra ODABAŞ YILDIRIM

İÇİNDEKİLER

Sayfa No

Yüksek Lisans Tezi

ÖZET

KAOTİK HARİTALARIN STEGANOGRAFİ İLE BİRLİKTE KULLANILMASI

Esra ODABAŞ YILDIRIM

Karadeniz Teknik Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı
Danışman: Doç. Dr. Mustafa ULUTAŞ
2014, 69 Sayfa, 6 Ek Sayfa

Teknolojinin gelişmesine bağlı olarak internet üzerinde sayısal resimler sıklıkla kullanılmaktadır. Sayısal resimler hem dağıtımının hem de çoğaltılmasının kolay olması, veri gizlemek için yeterince gürültüye sahip olmasından dolayı steganografide en yaygın kullanılan ortamlardan biri olmuştur.

Kaos, günümüzde karmaşıklık, düzensizlik, bilinememezlik olarak düşünülse de aslında kaos kavramı, agnostik bir kavram değildir ve kaosu bir sistemin gelecekteki davranışının bilinememesi olarak tanımlamak yanlış bir tanı olacaktır. Kaos bir "bilme" durumudur, bir sistemin gelecekteki davranışının öngörülemeyeceğinin bilinmesi durumu. Bu özelliğinden dolayı kaotik haritalar rasgele sayı üretiminde popüler hale gelmiş ve son zamanlarda yaygın olarak kullanılmaya başlanmıştır.

Bu çalışmada kaotik haritalarla rastgele sayılar üretilmiş ve üretilen sayılar NIST (National Institute of Standards and Technology) testlerinden geçirilmiştir. NIST testleri başarı ile sonuçlanan bu sayılar klasik LSB yönteminin sıralılık zaafiyetini gidermek için kullanılmıştır. Elde edilen stego görüntüler yine rasgele sıralı bir yöntem olan ayrık logaritma fonksiyonunu kullanan steganografi yöntemi ile karşılaştırılmıştır. Koas tabanlı steganografi, kaotik haritaların başlangıç koşullarına hassas bağımlılığı dolayısıyla, gizlediği sır bilgisinin geri elde edilmesini son derece güçleştirmiştir.

Anahtar Kelimeler: Kaos, Lojistik Harita, Steganografi, Dinamik Sistemler

Master Thesis

SUMMARY

USING CHAOTIC MAPS WITH STEGANOGRAPHY

Esra ODABAŞ YILDIRIM

Karadeniz Technical University
The Graduate School of Natural and Applied Sciences
Computer Engineering
Supervisor: Assoc. Prof. Musafa ULUTAŞ
2014, 69 Pages, 6 Pages Appendix

On the behalf of the technologic development digital images are frequently used on the internet. Digital images have been one of the most widely used medium in steganography because of distribution of digital images as well as be easy to replicate, to hide data due to having enough noise.

Although, chaos is considered as complexity, disorder, actually it is not an agnostic concept, and defining chaos as unknown future behavior of the system will be an incorrect diagnosis. Chaos is a "knowing" state, knowing that the future behavior of a system is unpredictable. Because of this property, chaotic maps have become popular in the random number generation, and recently has begun to be widely used.

In this study, random numbers was generated with chaotic maps, and NIST (National Institute of Standards and Technology) tests are performed on these produced numbers. The numbers resulted in the achievement with NIST tests, used to resolve the weakness of sequentially classic LSB methods. Obtained stego images are compared by steganography uses discrete logarithm function which is in random sequence. Chaos based steganography complicates to regeneration of secret information, owing to sensitive connectivity of starting conditions of chaotic maps.

Key Words: Chaos, Logistic Map, Steganography, Dynamic Systems

ŞEKİLLER DİZİNİ

Sayfa No

TABLOLAR DİZİNİ

Sayfa No

SEMBOLLER DİZİNİ

DCT : Ayrık kosinüs dönüşümü (Discrete Cosine Transform)

DFT : Ayrık fourier dönüşümü (Discrete Fourier Transform)

LSB : En az anlamlı bit (Least Significant Bit)

NIST : Ulusal Standart ve teknoloji enstitüsü (National institute of standarts and technology)

PSNR : Tepe sinyal gürültü oranı (Peak to signal noise ratio)

1. GENEL BİLGİLER

1.1. Çalışmanın Amacı

Günümüzde internet birçok bilgisayar sisteminin birbirine bağlı olduğu dünya çapında yaygın olan ve sürekli büyüyen bir iletişim ağı haline gelmiştir. İnternetin bu kadar talep görmesinin nedenlerinden birisi bilgiyi saklamak, paylaşmak ve ona kolayca ulaşma isteğidir. İnternet sayesinde insanlar birbirleriyle iletişim kurabilme imkanı bulabilmektedir. Bu iletişim kurulurken bilgi sistemlerinin internet'e açılması beraberinde güvenlik problemini getirecektir. Bu güvenlik problemi için alınan önlemlerden birisi de bilginin şifrelenerek kötü niyetli kişilerin dikkatini çekmesi yerine gizli bilginin anlamlı hale getirilerek iletilmesi; bilginin başka bir ortamda gizlenmesi işlemi; steganografidir.

Görüntü dosyalarının paylaşımının kolay olması ve içine veri gizlemek için yeteri kadar gürültüye sahip olması nedeni ile görüntü dosyaları steganografide yaygın olarak kullanılmıştır.

Kaotik haritalar da başlangıç koşullarına hassas bağımlılığı ve doğrusal olmayan sistemler olması dolayısı ile literatürde yaygın kullanılmıştır. Rastgele sayı üretiminde kaotik haritalar yaygın olarak kullanılmıştır.

Bu çalışmada steganografide verinin gizleneceği piksel pozisyonunun rastgele belirlenmesi amacı ile kaotik haritalar kullanılmış, üretilen stego görüntü içerisinde steg analizlerle veri gizlendiği belirlense bile, bu verinin görüntü içerisinden geri oluşturma işlemi kaotik haritaların başlangıç koşullarına hassas duyarlılığı sayesinde neredeyse imkansız hale gelmiştir.

1.2. Steganografi

1.2.1. Steganografinin Tarihçesi

Steganografi, içinde gizli mesaj veya bilgiler bulunan bir veriyi, alıcıdan başka kimsenin fark edemeyeceği bir biçimde gönderme sanatıdır. Steganografi kelimesi eski Yunan alfabesinden türemiş bir kelime olup örtülü yazı anlamına gelmektedir [1].

Steganografi'nin amacı gizli mesaj ya da bilginin varlığını saklamaktır. Taşınmak istenen mesaj bir başka masum görünüşlü ortamda saklanarak, üçüncü şahısların iletilen mesajın varlığından haberdar olması engellenir. Steganografi resim, ses, metin veya uygun formattaki bir sayısal dosyanın içerisine veri saklamada kullanılan bir bilgi güvenliği yaklaşımıdır.

Tarihte steganografi, hem şifreleme öncesi dönemde hem de sonrasında (ilgi çekmeme avantajından dolayı) kullanılmıştır. Steganografinin tarihsel gelişimi şu şekildedir;

MÖ 440'de Antik Yunan'da ulakların saçları kazınıp, saç derisine mesaj yazılmış, ulağın saçları uzadıktan sonra varacağı yere gitmiş ve saçları tekrar kazınmıştır.

Antik Çağlar'da, Antik Yunan'da mesajlar tahtaya yazılır, üzeri balmumu ile kaplanırdı. Böylece cisim kullanılmamış bir tablete benzerdi öte yandan mumun eritilmesiyle birlikte içindeki gizli mesaj okunabilirdi.

Steganografi hakkında yazılan ilk kitap Johannes Trithemus (1462–1516) tarafından yazılmış olan Steganographia isimli kitaptır. 1600'lü yıllarda yaşamış olan Gaspar Schott (1608–1666) tarafından yazılmış olan Schola Steganographica isimli kitapta ise müzik notalarının bilgi gizlemek için nasıl kullanıldığı anlatılmıştır. Bu yöntem birçok bilgi gizleme yöntemine de temel oluşturmuştur.

Daha sonraki yıllarda steganografi, görünmez mürekkep, metin belgelerindeki harf frekanslarını kullanma, I. ve II. Dünya Savaşlarında kullanılan mors kodları gibi uygulamalarla karşımıza çıkmaktadır. İkinci dünya savaşı esnasında, Alman casusların gizli bilgileri kimyevi bir madde ile beyaz bir mendile yazdıkları ortaya çıkartılmıştır. Casus, gizli mesaj içeren bu mendili daha önce belirlenen noktalarda çöpe atmakta; alıcı ise yine kimyevi maddeler kullanarak bu yazıyı okumaktadır.

Yine ikinci dünya savaşı döneminde Almanlar "mikrofilm" teknolojisi kullanarak "mikro noktalar" (microdot) kullanmıştırlar. Bu yöntemde A4 büyüklüğündeki herhangi bir belge veya çizim bir dizi işlem sonrasında daktilo yazısında kullanılan bir nokta kadar küçültülmektedirler. Bu yöntem kullanılarak masum içerikli bir sayfa düz metindeki i ve j harflerinin noktalarına oldukça büyük miktarda veri saklamak mümkün olmuştur.

Amerika'da yaşayan bir Japon ajanının, oyuncak bebek siparişi gibi görünen mesajlarla diğer ajanlarla ve hükümetiyle gizli bir şekilde mesajlaşması, Fransızların görünmez mürekkep kullanarak gönderilen postaların üzerine bir takım notlar saklaması, mektup pullarının arka yüzeylerine yazılan bir takım notlar da steganografinin 20.

yüzyıldaki yaygın kullanımına örnek olarak gösterilebilir. Bir diğer örnekse ikinci dünya savaşı sırasında Alman bir casus tarafından gönderilen bir telgraftır;

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

Bu yazıda her kelimenin ikinci harfleri alınıp yan yana koyulduğu zaman Alman casusun göndermek istediği gizli mesaj elde edilmektedir;

*"*Pershing sails from NY June 1*"*

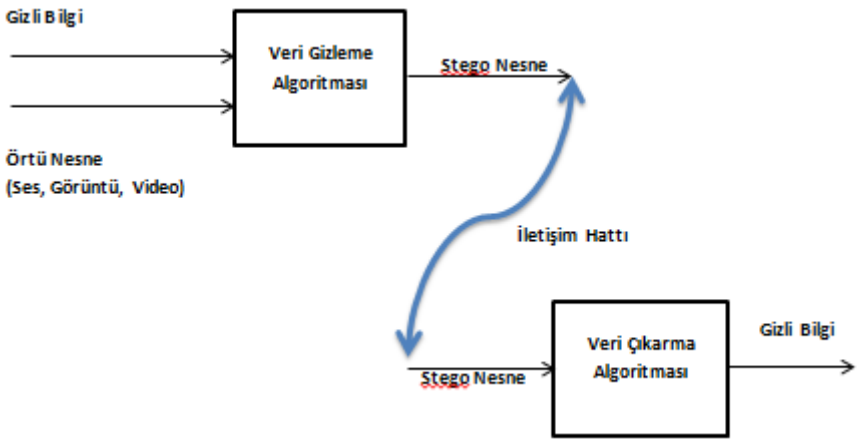
Yukarıdaki mesaj bir başka gazetecinin yazdığı bir yazıyla da elde edilebilmektedir;

*"*President's Embargo Rulıng Should Have Immediate Notice. Grave Situation Affecting International Law. Statement Foreshadows Ruın Of Many Neutrals. Yellow Journals Unifying National Excitement Immensely.*"*Bu yazıda da her kelimenin ilk harfleri alındığında, Alman casusun gönderdiği bilgiyle aynı sonuca ulaşılmaktadır. Bu da bize steganografinin, aynı mesajı vermek için birden fazla örtü verisini desteklemesi, dolayısıyla ne kadar güvenli olduğu konusunda gayet açık bir fikir sunmaktadır.

Steganografi gizli bir iletişim sağlamaktadır. Amacı iki kişi arasındaki iletişimin bir üçüncü şahıs tarafından fark edilememesidir. Bilimsel ortamda Steganografi çalışmaları 1984 yılında Simmons tarafından "Prisoner Problem"'in tanımlanması ile başlamaktadır. Bu problemde Alice ve Bob hapishanededir ve hapishaneden kaçmak için planlar yapmaktadırlar. Fakat bu planların gardiyan Willie'ye fark ettirilmeden yapılması gerekmektedir. Eğer Willie bunu fark ederse kaçma planları suya düşecektir. Bu nedenle de çeşitli gizli haberleşme yöntemleri geliştirilmesi gerekmektedir [2].

Steganografi'de kendisine bilgi gönderilen kişi bile ancak anahtar bilgisini bilmesi durumunda gizli veriyi elde edebilir. Kriptografiden farklı olarak steganografide bilginin şifrelenmesi önemli değildir. Kriptografi, sağlamlığını şifreleme algoritmasından alan ve iletilmek istenen verinin varlığının bilinmesinden çekinmeyen bir bilimdir. Dolayısıyla, kriptografide verinin hangi kanalla taşınacağı önemsenen bir durum değildir, fakat ne kadar sağlam olursa olsun hiçbir algoritma çözülemez değildir. Bundan farklı olarak steganografide, verinin nasıl taşınacağı saklanmak zorundadır. Steganografide önemsenen nokta, verinin varlığının gizlenmesidir. Verinin varlığı ne kadar iyi gizlenebilirse, taşınacak veri o kadar güvendedir. Sıradan bir resim veya müzik dosyası kimsede kuşku uyandırmazken, içinde çok gizli bilgiler taşıyor olabilir. Alıcı dahil hiç kimse anahtar bilgi olmadan, verinin nerede saklandığını bilmeden, bundan kuşkulanmaz. Bu yüzden

steganografide, alıcının elinde, verinin hangi kanalla taşınacağı ve mesajın nasıl çözülebileceği bilgileri olmadan verinin karşı tarafa iletilmesi hiçbir önem taşımaz. Şekil 1.1'de veri gömme ve çıkarma prosedürleri görülmektedir.



Şekil 1.1. Steganografi'nin çalışma mantığı

Steganografi hakkında literatürde çeşitli tanımlar yapılmaktadır. Bir tanıma göre steganografi gizli mesajın varlığının tespit edilemediği bir iletişim bilimidir [1]. Baska bir tanıma göre ise görünüşte zararsız bir mesajın içerisine veri saklama sanatıdır [3]. Bu bilim dalı askeri literatürde ise kısaca TRANSEC olarak adlandırılmaktadır [4]. Tanımlar farklı olsa da steganografide temel amaç iletişimin gizliliğinin sağlanmasıdır. Yani iki nokta arasında gerçekleştirilen iletişimde gizli bir veri aktarımının anlaşılmamasının 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 iletisimin gizliliğinin gerçekleştirilmesi yönüyle yüksek seviyede bir bilgi güvenliğinin oluşturulmasına katkılar sağlamaktadır.

Sayısal ortamların yaygın olarak kullanıldığı günümüzde, sayısal nesneler üzerinde steganografi uygulamaları yapılmaktadır ve gelişen teknoloji nedeniyle, verilerimizi korumak amacıyla son yıllarda sıklıkla kullanılmaya başlanmıştır. Gizli veri yine masum içeriğe sahip olan bir dizi dosyanın içinde saklanabilmektedir. Bunlardan en ilgi çekicileri, vermiş oldukları olanaklardan dolayı, resim, ses ve video dosyalarıdır. Benzer bir şekilde düz metin dosyaları, sabit disklerdeki kullanılmayan alanlar, IP (Internet Protocol) paketlerinin ileride kullanmak üzere ayrılmış bölümleri gizli verinin saklanması için

kullanılabilmektedir. Html dosyaları, exe dosyaları vb. gibi dosyalar da içlerine veri saklamada kullanılabilmektedir.

1.2.2. Text (Metin) Steganografi

Metin steganografi bilgi gizlenecek ortamın metin (text) olduğu steganografi koludur. Metin steganografinin uygulanabilmesi için çeşitli yöntemler vardır. Bunlar şu şekilde sınıflandırılabilir;

- Açık Alan Yöntemleri (Open Space Methods)
  - o Satır Kaydırma Kodlaması
  - o Kelime Kaydırma Kodlaması
  - o Gelecek Kodlaması
- Yazımsal Yöntemler (Syntactic Methods)
- Anlamsal Yöntemler (Semantic Methods)

Metin steganografi en eski steganografik yöntemlerden biridir. Tarihimizden buna örnek olarak cümlelerin içerisine tarih saklama yöntemi olan ebcet hesabı verilebilir. Metin steganografi, steganografinin en zor uygulandığı alandır [6]. Bu ses ve görüntü gibi diğer ortamlara nazaran tekrar eden lüzumsuz verinin çok daha az olmasından kaynaklanmaktadır. Bu yöntemi Tayland dilinde, İngilizce, Japonca, Korece, Çince, Farsça ve Arapça'da başarıyla uygulayan çalışmalar bulunmaktadır [5]. Shirali-Shahreza bir çalışmasında [6] Arapça ve Farsça'daki harflerin çoğunda noktalama işaretlerinin bulunduğundan bahsetmiş ve bu işaretlerin dikey yüksekliklerinin saklama amacıyla kullanıldığı bir yöntem önermiştir.

Meral H. M., Sankur B., Özsoy A. S., 2006'da Türkçe için yapılan bir çalışmada, Türkçe metinlere gizli bilgi yükleme olanakları yapısal anlamsal ve işaretsel değişiklikler açısından incelenmiştir [7].

Shirali-Shahreza, konuşma seslerindeki sessizlik aralıklarının uzunluklarını değiştirerek gerçek zamanlı saklama yapan ve MP3 sıkıştırmaya dayanıklı yeni bir yöntem önermiştir [10].

1.2.3. Ses Steganografi

Yaygın dağıtımı ve içerdiği bilginin steganografiye uygun olması nedeni ile ses dosyaları da steganografide kullanılmaktadır. Ses steganografide kullanılan yöntemlerden bazıları;

- Düşük bit şifrelemesi (Low bit encoding)
- Faz kodlaması (Phase coding)
- Tayfa yayılım (Spread spectrum)
- Yankı veri gizleme (Echo data hiding)

yöntemleridir.

Pakistan'da üç akademisyen çalışmalarında LSB (Least Significant Bit-En az anlamlı bit) yönteminin gelişmiş bir versiyonunu önermektedirler [8]. Bu yöntemde saklama işleminde 8 bitlik ses örnekleri kullanılmakta, örnek ve saklama yapılacak bit seçimi bir haritaya göre yapılmaktadır. Böylece üçüncü şahıslara saklamanın rastgele yapıldığı izlenimi verilmektedir. Bu da istatistiksel saldırılara karşı güvenlik seviyesinin artmasını sağlanmaktadır.

Diqun Yan ve Rangding Wang 2009 yılında yayınlanan makalelerinde MP3 sıkıştırma standardının özelliklerinden biri olan Huffman kodlaması işleminde tablo seçimi yaparak steganografi sağlamışlardır [9].

1.2.3.1. Düşük Bit Şifrelemesi

Dayanıksız bir yapısı vardır. Resim steganografide bahsedilecek olan LSB ekleme yöntemiyle aynı şekilde gerçekleştirilir. Ses dosyasındaki verinin her baytının son bitine gizlenecek bilginin bir biti yazılır. Sonuçta oluşan değişiklik ses dosyasında gürültüye neden olmaktadır. Tekrar örnekleme veya kanalda oluşabilecek gürültü ile mesaj zarar görebilir veya yok edilebilir.

1.2.3.2. Faz Kodlaması

Faz kodlaması yöntemi de resim dosyalarında uygulanan JPEG algoritmasına benzer bir yöntemdir. Veriyi gizleme işleminde ses dosyası küçük parçalara bölünür ve her parçaya ait faz gizlenecek veriye ait faz referansı ile değiştirilir.

1.2.3.3. Tayfa Yayılım

Gizleme işlemini ses sinyalinin kullandığı frekans tayfı üzerinde yapmaktadır. Güçlü bir yapısı olmakla birlikte seste gürültü meydana getirmektedir.

1.2.3.4. Yankı Veri Gizleme

Bilginin gizlenmesi taşıyıcı ses sinyali üzerine bir yankı eklenmesi ile sağlanmaktadır. Bilgi, yankının gecikme miktarı, zayıflama oranı veya büyüklüğü gibi değerler kullanılarak gizlenir. Her bitin kodlanması için sinyal parçalara bölünür.

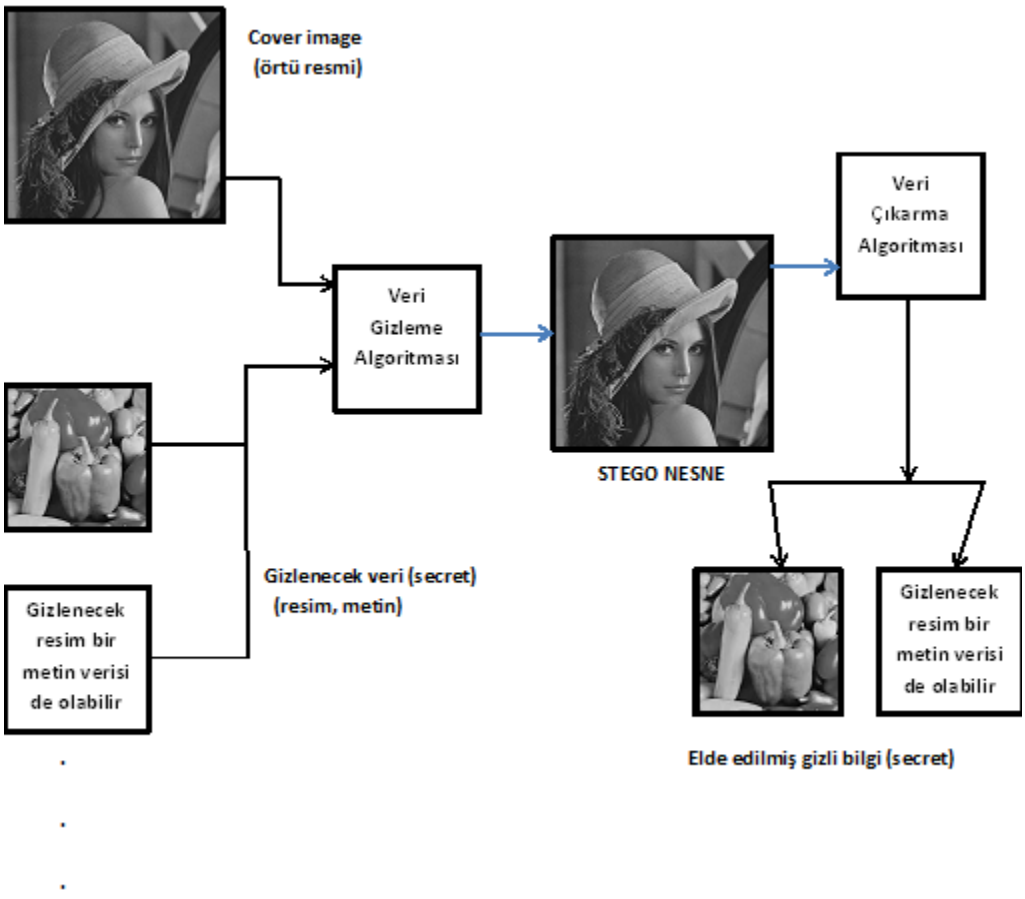
Yankı veri gizleme yöntemi herhangi bir gürültüye neden olmamakta veya kayıplı bir kodlama kullanmamaktadır.

1.2.4. Görüntü (Resim) Steganografi

Sayısal resimler dağıtımı kolay ve internette yaygın paylaşılan dosyalardır. Bu nedenle görüntü dosyaları steganografide yaygın bir çalışma alanı oluşturmuştur. Görüntü dosyalarının içerisine yine bir görüntü dosyası gizlenebileceği gibi, bir metin dosyası da gizlenebilir.

Gizli bilgiyi bir resme gömme (ya da gizleme) işleminde iki dosya söz konusudur. Kapak resim, taşıyıcı ya da örtü verisi (cover image) olarak adlandırılan ilk dosya, gizli bilgiyi saklayacak resim dosyasıdır. İkinci dosya ise gizlenecek bilgi olan mesajdır. Bu mesaj da sır görüntü (secret image) olarak adlandırılır. Gömme (saklama) işlemi sonucunda oluşan, orjinalinden ayırt edilemeyen medyaya ise Stego Image (Stego görüntü) denilmektedir. Stego, saklanan veriyle taşıyıcı medyanın bileştirilmiş haline denilmektedir.

Şekil 1.2'de basit bir steganografik sistem gösterilmiştir. Resim dosyası içine başka bir resim dosyası gizlenebileceği gibi bir metin de gizlenebilir.



Şekil 1.2. Steganografi şeması

Görüntü steganografide, bilgilerin resim dosyaları içine saklanmasının çeşitli yöntemleri vardır. Şekil 1.2'de Gizleme Fonksiyonu olarak gösterilen ve en çok kullanılan yöntemler aşağıdaki verilmiştir;

- En önemsiz bite ekleme
- Maskeleme ve filtreleme
- Algoritmalar ve dönüşümler

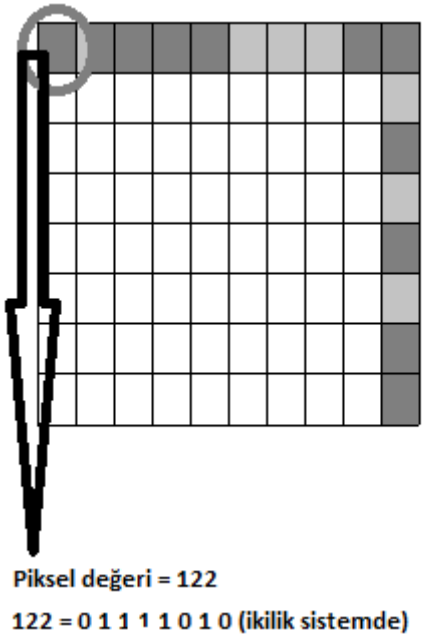
1.3. Görüntü (Image) Steganografi

1.3.1. Sayısal Resmin Yapısı

Sayısal (dijital) resim N satır ve M sütunluk bir dizi ile temsil edilir. Genellikle satır ve sütun indeksleri y ve x veya r ve c olarak gösterilebilir.

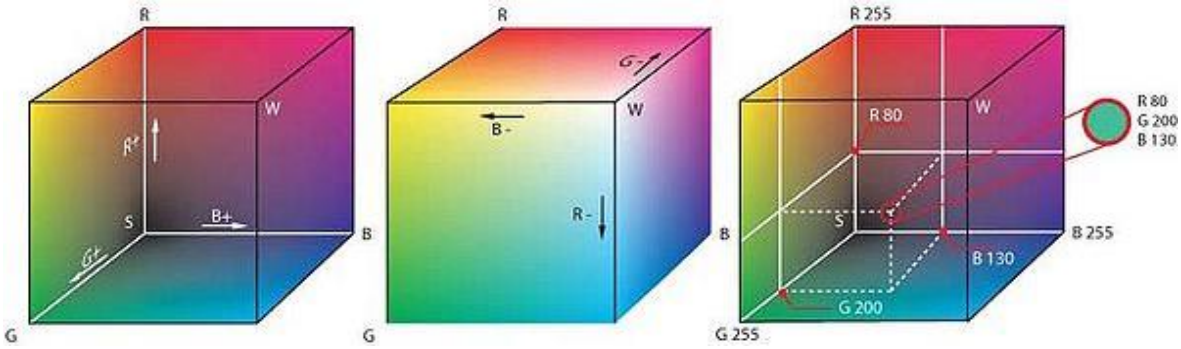
Bir resim dizisinin elemanlarına piksel denir. En basit durumda pikseller 0 veya 1 değerini alırlar. Bu piksellerden oluşan resimlere ikili (binary) resim denir.

Gri-seviye resimlerde her piksel, 0 (siyah) ile 255 (beyaz) arasında tam sayı değer alabilen 1 byte ile temsil edilmektedir. 0–255 arasındaki değerler gri'dir ve bundan dolayı bir resme ait tam sayı "gri ton seviye" (gray level) olarak isimlendirilmektedir. Şekil 1.3'te gri seviye sayısal bir resmin yapısı gösterilmiştir.



Şekil 1.3. Gri seviye sayısal resmin yapısı

24 bitlik piksellerden oluşan resimler en fazla veri saklayabilen resimlerdir [11]. 24 bit resimler bir piksel başına 3 byte kullanmaktadır. Bütün renk değerleri kırmızı, yeşil ve mavi (RGB) renklerinin tonlarının karışımından elde edilir [11]. 256 farklı renk tonuna elde edilir. Şekil 1.4'te renk küpü gösterilmektedir.



Şekil 1. 4. Renk küpü

1.3.2. Resim Dosyalarının Sıkıştırılması

Resim dosyalarında “kayıplı ve “kayıpsız” olarak iki sıkıştırma yöntemi vardır. Her iki yöntem de dosya büyüklüğünü azaltmaktadır. Ancak gömülü bilginin etkilenmesi açısından farklı sonuçlar vermektedirler.

Kayıpsız sıkıştırma, orijinal mesajın doğru olarak elde edilmesini sağlamaktadır. Bu nedenle orijinal bilginin eksiksiz elde edilmesi gereken durumlarda tercih edilmektedir. GIF (Graphic Interchange Format) ve BMP (BitMaP) formatları bu yapıdadır.

Kayıplı sıkıştırma, dosya büyüklüğünü büyük ölçüde azaltmakta ancak resmin bütünlüğünü korumamaktadır. JPEG resimler bu tip sıkıştırma kullanan resimlerdir. Sıkıştırma sonrasında gizlenen bilgide kayıplar meydana gelebilmektedir. Kullanılan kayıplı sıkıştırma algoritmasına bağlı olarak JPEG formatı, yüksek kaliteli sayısal resimlere yakın sonuç vermektedir.

BMP formatındaki resimler içinse herhangi bir sıkıştırma işlemi uygulanmamaktadır. Bu nedenle resim boyutu fazla olmakta lakin veri kaybı olmamaktadır.

1.3.3. Veri Gizleme Yöntemleri

Günümüze kadar resim içerisine veri saklama konusunda birçok çalışma yapılmıştır [11-18]. Bu çalışmalarda jpeg formatındaki resimler içerisine [18], gri seviyeli bitmap resimler içerisine [11], siyah-beyaz resimler içerisine [16], LSB modifikasyonu, dönüştürme tekniği, maskeleme ve filtreleme gibi yaklaşımlarla [19,20] veri saklama bunlara verilebilecek örneklerdir.

Görüntü dosyalarında bilgi gizleme temel olarak iki ana başlıkta toplanabilir. Bunlar;

1. Uzaysal / Görüntü Kümesi (Spatial / Image Domain) Tekniği
2. Frekans / Dönüşüm Kümesi (Frequency / Transform Domain) Tekniği

Uzaysal küme (görüntü kümesi) olarak adlandırılan teknik, gizleme işleminde resim dosyasındaki veriyi doğrudan kullanır. Bu tekniğe örnek olarak ileride anlatılacak olan ve yaygın olarak kullanılan en az anlamlı bite ekleme yöntemi gösterilebilir.

Frekans kümesi (dönüşüm kümesi) olarak bilinen teknik ise kapak verideki değişimler üzerinde gizleme işlemini uygular. Dönüşüm kümesi tekniğine örnek olarak JPEG formatlı resim dosyalarına veri gizleme işleminde kullanılan algoritmalar verilebilir.

1.3.4. Resim Dosyalarında Steganografik Yöntemler

Resim dosyalarında bilgiyi gizlemek için kullanılan çeşitli yöntemler vardır. Bu yöntemler 3 gruba ayrılabilir [21];

Değiştirmeye dayalı yöntemler: Bu grup yöntemlere LSB Yöntemi ya da Genlik Modülasyonu kullanılarak bilgi gizleme verilebilir. Burada bilgi gizlemek için renk değerleriyle oynanabilir ya da palet değiştirilebilir.

Renk değerleriyle oynama en basit yöntemdir. Renk değerlerinin düşük anlamlı bitleri ile gizli verinin bitleri değiştirilir. Değişim, insan gözü tarafından algılanmaz. Gizli veri, "gürültü (noise)" olarak resme eklenir. Bu yöntem yüksek oranda veri gömme şansı verir, fakat resim üzerinde yapılacak değişimlere karşı oldukça hassastır.

Yer değiştirmeye dayalı yöntemlerde, yer değişimi için kullanılacak bitlerin sayısı bir bitten daha fazla da olabilir. Ancak karar vermede, kullanılan resmin özelliği belirleyici unsurdur. Resim dosyasında düz alanlar fazlaysa yapılan yer değiştirme işlemi çıplak gözle bile fark edilebilir.

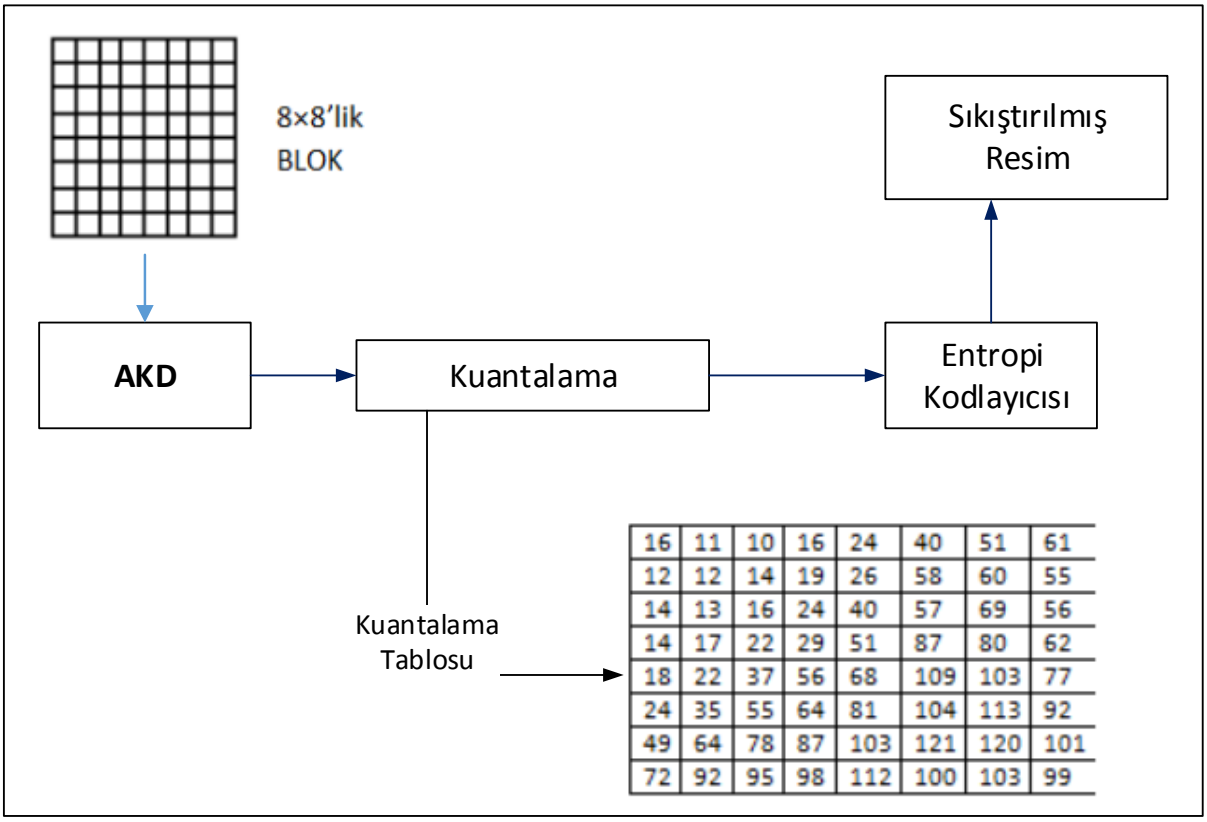
Palet ile oynamada ise renk bilgilerinin palet üzerinde tutulduğu resim dosyaları kullanılır. Paletteki sıralama değiştirilir. Bunun sonucunda resim bozulabilir. Ayrıca resim türü değiştirildiğinde tüm yapılanlar yok edilir

İşaret işlemeye dayalı yöntemler: Bu yöntemler çeşitli dönüşümlerin kullanıldığı yöntemlerdir. DCT, DFT gibi dönüşümler kullanılabilir.

Kaliteden belli ölçülerde ödün vererek, grafik dosyalarının büyüklüğü ile ciddi miktarda oynamak mümkündür. Internet kullanıcıları var olan resim sıkıştırma algoritmaları ile yüksek oranda verim alabilmektedir. Bu sıkıştırma algoritmaları içinde

JPEG, GİF, PNG algoritmaları önemli yer tutar. Tüm sıkıştırma algoritmaları insan gözünün filtreleme özelliğini kullanır. İnsan gözü belli bir frekansa kadar renk değişimlerini algılayabilir. Dolaysıyla, asıl resimdeki renk değerleri frekans düzlemine taşınabilir. Frekans düzleminde kullandığımız dönüşüme göre karşımıza çıkan çeşitli katsayılar kullanılarak yapılan ters işlem sonrası tekrar asıl resmi elde etmek mümkündür. Asıl resmi elde etmek için belirli bir kaybı göz önüne alarak sonsuz sayıda frekans bileşeninden yararlanmaktan kaçınılabilir ve bu vazgeçilen yüksek frekans katsayısı, tekrar elde edilen resmin kalitesini ortaya koyar.

Sıkıştırma algoritmaları içerinde yaygın olarak kullanılan JPEG sıkıştırma algoritması da benzer bir ilkeyle çalışmaktadır (Şekil 1.5).



Şekil 1.5. JPEG sıkıştırma algoritma şeması

Şekil 1.5'te de gösterildiği gibi JPEG sıkıştırması, ilk önce resmi 8x8 renk değerinden oluşan parçalara böler. Parçaların her biri üzerinde frekans düzlemine geçmek üzere ayrık kosinüs dönüşümü uygular. Bu dönüşümün tercih edilmesinin sebebi katsayıların karmaşık değil reel sayılar olmasıdır. Elde edilen sayılar, seçilen kalite oranına

göre belirlenen bir tablo kullanılarak kuantalanır. Kuantalama tablosu kaliteye göre yüksek frekans katsayılarından ne kadarının yok edileceğini belirler. Kuantalama işlemi sonucunda ayrık kosinüs dönüşümü sonrası elde edilen 64 adet frekans katsayılarından bir kısmı sıfır değerini alacaktır. Bu katsayıların Huffman kodlaması ile sıkıştırılması ciddi oranda yer kazanımı sağlar.

Resim tekrar elde edilmek istenildiğinde sıkıştırma işleminin tersi uygulanır. Öncelikle Huffman kodu çözülür ve içerisinde sıfır katsayılarının da olduğu 64 adet katsayıdan oluşan blok ile yine aynı kuantalama tablosu ile çarpılır. Birçok katsayı asıl değerine yakın değerlere ulaşacaktır ancak sıfır ile çarpılanlar sıfır kalacaktır. İşte bu, ters kosinüs dönüşümü sonrası elde edilen 8x8 renk değerlerindeki kayıbı belirleyecektir. Sıkıştırma işleminde kullanılan yöntem, steganografik bir algoritma oluştururken bize yol göstermektedir. Her şeyden önce 8x8 renk bloklarının dikkate alınması gerekir. Bunun yanı sıra, sıkıştırma işlemi kayıplı olacaktır. Bu işlemlere dayanıklı bir algoritma ayrık kosinüs dönüşümünü dikkate almalıdır. Önerilen birçok yöntemde, algoritmalar ayrık kosinüs dönüşümünü kendileri yapmaktadırlar. Daha sonra yok edilecek frekans katsayılarını dikkate alarak her bir blokta 1 bit saklarlar. Burada önemli olan, düşük frekans değişimlerinin resmi bozacağının kesin olmasıdır. Öte yandan, yüksek frekans katsayıları da muhtemelen yok edileceklerdir. Dolayısıyla resmi değiştirmeyecek kadar yüksek, yok edilmeyecek kadar düşük frekans katsayılarının bulunması gerekir. Ayrık kosinüs dönüşümü kullanarak gizli iletişimi sağlayan yöntemlerde ortak sorun, katsayıların belirlenmesidir. Katsayılar resimden resime fark gösterecektir fakat resimde bozulma olup olmayacağı ancak işlem sonrası anlaşılabilir. Ayrıca, her 64 adetlik bloklara 1 bit gömülebilmesi, gizlenebilen veri miktarını önemli oranda düşürmektedir.

Spektrum yayılmasına dayalı yöntemler: Tayf (Spektrum) yayılmasına dayalı yöntemler son yıllarda yapılan çalışmaların en dikkat çekenidir, oldukça fazla kullanılmaya başlanmıştır. Tayf yayılması askeri iletişimde çok eskiden beri kullanılan bir metottur. Bu yöntemde gönderilmek istenen mesaj ihtiyaç duyduğu frekans bandından çok daha fazlasına dağıtılır. Üçüncü bir kişi araya girip bir ya da birden fazla frekans bandında bozulmalara neden olsa bile, alıcı geri kalan frekans bantlarındaki bilgiler ile asıl mesajı elde edebilmektedir. Gizli mesaj birden fazla bantta yayılarak resme gürültü olarak eklenebilir.

Yukarıdaki yöntemleri kullanarak bilgi gizleyen steganografik algoritmalardan en yaygın olarak kullanılanları şunlardır;

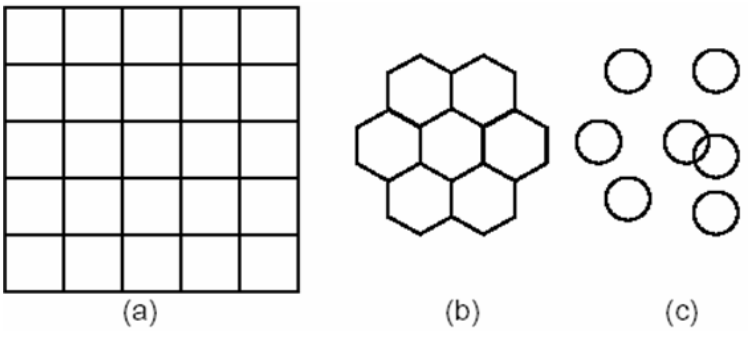
− Patchwork Algoritması
− Amplitude (genlik) Modülasyonu kullanılarak bilgi gizleme
− Toplamsallık Algoritması
− SSIS (Spread Spectrum Image Steganography) Yöntemi
− Frekans Domeni İçine Veri Saklanması
− Son Bite Saklama yöntemi

1.3.4.1. Patchwork (Yama) Algoritması

Bender tarafından ortaya atılan ve halen sıklıkla kullanılan algoritmadır [22]. Bu algoritma, bilgiyi Gauss dağılımı gösteren bir istatistiğe sahip örtü verisinin içine gizlemeyi amaçlayan bir istatistiksel yönteme dayanır. Bu algoritma genelde filigran (watermarking) uygulamalarında kullanılmaktadır. Bu algoritma genelde 256 bir gri-seviye resimler için kullanılmaktadır. Algoritmada iki ana adım vardır:

- Sahte rastgele yöntemle iki yama seç
- İlk yamaya d sabit sayısını ekle, ikinci yamadan d sabit sayısını çıkar.

Parlaklık değerleri üzerinde uygulanan arttırma ve azaltma aynı miktarda yapılır. Bu sebeple ortalama parlaklık değeri değişmez. Burada yama şekilleri de oldukça önem kazanmaktadır. Olası üç adet tek boyutlu yama şekli vardır. Eğer keskin kenar içeren küçük yamalar seçersek, yamanın enerjisi görüntü analizinin yüksek frekanslı kısmı içerisinde yoğunlaşacaktır. Fark edilmesi oldukça zordur fakat filtreleme sonucunda kolaylıkla elde edilebilir. Diğer bir olasılık yumuşatılmış kenarlar içeren yamalar kullanılmasıdır. Bu durumda bilgiler düşük-frekans analizi içinde kalacaktır. Üçüncü olasılık ise ilk iki olasılığın birleştirilmesidir. Bu şekilde yamanın enerjisi dağıtılmaktadır.



Şekil 1.6. Yama çeşitleri

Şekil 1.6 (a)'da doğrusal bir kafes biçimi kullanılmıştır. Fakat bu pek tercih edilen bir yöntem değildir. (b) a'ya alternatif olarak gösterilmiştir. Fakat en tercih edilir çözüm (c)'de verilmektedir. Buradaki yamalar rastgele olarak dağılmakta ve seçilebilmektedir. Akıllıca bir dağılımla her türlü çözünürlükte iyi sonuç verebilecektir.

1.3.4.2. Amplitude (Genlik) Modülasyonu Kullanılarak Bilgi Gizleme

Amplitude modülasyonu kullanılarak işaret bitleri mavi kanaldaki piksel değerleri değiştirilerek gömülür. Bu değişimler ışığın oranına ve bitin değerine bağlı olarak arttırma ya da eksiltme yoluyla yapılmaktadır.

s ; saklanacak tek bir bit;

I ={R,B,G} görüntü;

p(i, j) ; I görüntüsü içerisinde sahte rastgele (pseudo random) olarak seçilmiş bir pozisyon;

K'da üretilen anahtar olarak tanımlanmaktadır.

Gizlenecek bit L = 0.299R + 0.587G + 0.114B olmak üzere B mavi kanal (blue channel)'ın değiştirilmesiyle p pozisyonuna saklanır.

$$B_{ij} = B_{ij} + (2s - 1)L_{ij}q \tag{1.1}$$

Eşitlik (1.1)'deki q, imzanın gücüne göre belirlenmiş olan bir sabittir. q 'nun değeri, veri saklamanın amacına bağlı olarak seçilmektedir. q 'nun değeri değiştirilerek veri gizleme ya da doküman işaretleme (document marking) işlemleri yapılabilmektedir.

1.3.4.3. Toplamsallık Algoritması

Toplamsallık algoritması Pitas ve Nikolaidis'in 1996'da yaptığı çalışmalarında, daha çok filigran uygulamaları için kullanılmaktadır. NxM çözünürlüğe sahip bir görüntü eşitlik (1.2)'deki ifade edilmektedir [23].

$$\text{I} = \{X_{mn}, n \in \{0, \dots, N-1\}, m \in \{0, \dots, M-1\}\} \tag{1.2}$$

Buradaki $X_{mn}, n \in \{0, \ldots., N-1\}$, (n,m) pikselinin koyuluk seviyesini belirlemektedir. Gizlenecek bitler, 1 ya da 0 değeri alabilen aynı büyüklükteki ikili çiftler olarak gösterilebilir.

$$S=S_{mn}, n \in \{0, \ldots., N-1\}, m \in \{0, \ldots., M-1\}, S_{mn} \in \{0,1\} \quad (1.3)$$

Bu aşamadan sonra I , S kullanılarak iki eş büyüklükte alt kümeye bölünebilir (1.3).

$$\begin{aligned} &A = X_{mn} \in I, S_{mn} = 1 \\ &B = X_{mn} \in I, S_{mn} = 0 \\ &|A| = |B| = \frac{|I|}{2} = \frac{|NxM|}{2} = P \\ &I = A \cup B \end{aligned} \quad (1.4)$$

Filigran (işaret), görüntü üzerinde şu şekilde ilave edilir. $C = \{X_{mn} \otimes k, X_{mn} \in A\}$. Buradaki $\otimes$ işlemi Toplamsallık kuralı olarak bilinmektedir. İşaretlenmiş görüntü (1.5)'teki gibi verilmektedir.

$$I_s = C \cup B \quad (1.5)$$

1.3.4.4. SSIS (Spread Spectrum Image Steganography) Yöntemi

Yayılmış Spektrum iletişim dar banttaki sinyalin geniş bant sinyal üzerinde yayılarak gönderilmesidir. Bu yayılma işleminden sonra dar bant sinyali tespit etmek zorlaşır.

Lisa M. Marvel ve arkadaşları 1999 yılında SSIS (Spread Spectrum Image Steganography) yöntemini tanıtmışlardır [24]. Bu yöntem gizli bilgiyi gürültüiçerisine saklamayı temel alır. Gizli bilgiyi başarıyla çıkarabilmek için restorasyon teknikleri kullanılmış. Restorasyon ile orijinal görüntü elde edilebilirse eklenen gürültü de elde edilecektir. Bu tekniklerle bilgi hatasız olarak elde edilemediği için saklamadan önce hata kontrol kodlaması uygulanmıştır. Ayrıca saklanan verinin görüntünün iyi seçilmiş parametrelerle sıkıştırılmasına dayanabildiği tespit edilmiştir.

K.Satish ve arkadaşları 2004 yılında Lisa'nın yöntemine kaotik şifreleme ve kaotik modülasyon ekleyen ve böylece daha yüksek güvenlik sağlayan bir çalışma yapmışlardır

[25]. 2008'deki çalışmada R.S.Youail ve arkadaşları ise pikseldeki renk bayt'ının son biti yerine orta bitleri rastgele bir şekilde saklama amacıyla kullanmış ve böylece sıkıştırma ve gürültü gibi etkilere daha dayanıklı ve daha güvenli hale getirilmiştir [26]. SSIS frekans alanında uygulandığında ise görsel olarak çok daha iyi performans göstermiş ve orta ve yüksek gürültü seviyelerinde daha da iyi sonuç vermiştir [27].

1.3.4.5. Frekans Domeni İçine Veri Saklanması

Filigran (watermark) teknolojileri için kullanılan bu yöntem görüntülerin dönüştürülmesi temeline dayanmaktadır. Görüntü dönüştürülmesi için genellikle ayrık kosinüs dönüşümü (DCT) kullanılmaktadır. Bunun dışında kullanılan dönüşüm algoritmaları ise; Ayrık Fourier Dönüşümü (DFT), Walsh dönüşümü ya da Wavelet (dalgacık) dönüşümüdür.

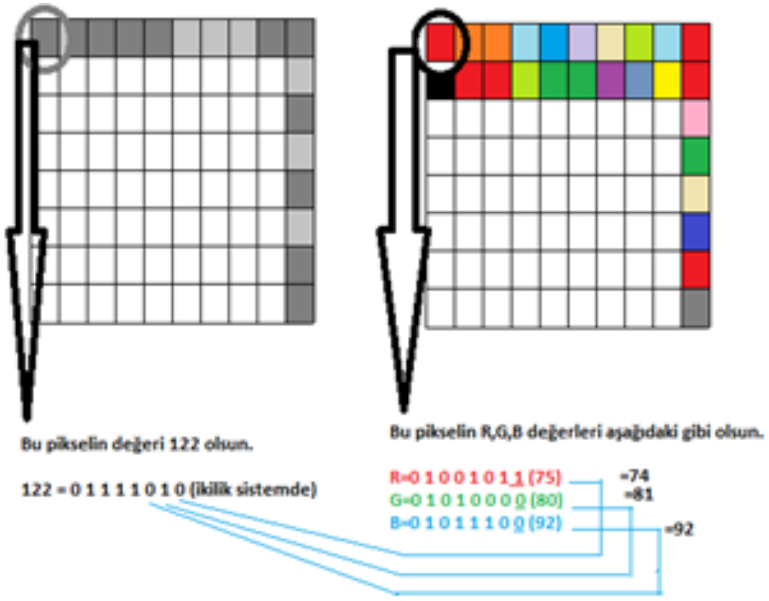
DCT steganografisinin en çok kullanıldığı görüntü dosyaları JPEG'dir. JPEG dosyalarında bir blok üzerinde DCT uygulanıp kuantalama ve yuvarlama yapıldıktan sonra elde edilen sonuç blok üzerinde LSB yöntemini uygularız. Bu işlem yukarıda "İşaret İşlemeye Dayalı Yöntemler" kısmında Şekil 1.5'te gösterilmiştir. Burada hangi değerlere bilgi saklanacağını iyi seçmek son derece önemlidir, çünkü bir değer üzerinde yapılan değişiklik tüm bloğu etkileyecektir.

1.3.4.6. En Az Anlamlı Bite Saklama Yöntemi

En önemsiz bite ekleme yöntemi (Least Significant Bit Insertion Methods) yaygın olarak kullanılan ve uygulaması basit bir yöntemdir. Fakat yöntemin dikkatsizce uygulanması durumunda veri kayıpları ortaya çıkmaktadır. Bu yöntemde; resmi oluşturan her pikselin her byte'nın en önemsiz biti olan son biti değiştirilerek o bitin yerine gizlenmesini istediğimiz verinin bitleri sırasıyla verinin başlangıcından itibaren birer birer yerleştirilmektedir. Burada her sekiz bitin en fazla bir biti değişikliğe uğratıldığından ve eğer değişiklik olmuşsa da değişiklik yapılan bitin byte'ın en az anlamlı biti olmasından dolayı, ortaya çıkan stego nesnesindeki (= örtü verisi + gömülü veri) değişimler insan tarafından algılanamaz boyutta olmaktadır. Burada 8 bitlik veriyi 8 piksele gömmüş olacağımızdan saklayabileceğimiz veri kapasitesi resim boyutunun 1/8'i kadar olmaktadır.

Her bir pikselde kullandığımız en önemsiz bit sayısını artırdıkça saklayacağımız veri kapasitesini aynı oranda artırmış oluruz ama burda resimde gözle fark edilebilecek değişimlerden kaçınmak gerekir.

Resim içine bir metin gizleyebileceğimiz gibi başka bir resim de gizleyebiliriz. Şekil 1.7'de gri bir görüntü dosyasının renkli görüntü dosyasının RGB kanalları kullanarak nasıl gizlendiği basitçe gösterilmiştir.



Şekil 1.7. Görüntü dosyaları üzerinde LSB yönteminin gerçeklenmesi

Yukarıda bahsedildiği üzere Şekil 1.7'de renkli resimin her pikseline 3 bit gömdüğümüzden burada veri saklama kapasitemiz resim boyutunun 3/8'i kadar olmaktadır. Örneğin 512x512 boyutlarına sahip 24 bit derinliğine sahip (renkli) bir resimde toplam piksel sayısı 262144, toplam gizlenebilecek veri miktari 98304 piksellik bir resimdir.

LSB yönteminde genellikle gri seviye veya 24 bitlik renkli görüntüler tercih edilmektedir. 8 bitlik görüntülerde piksel başına 1 byte kullanılmaktadır. 8 bitlik görüntüler renk sınırlaması yüzünden çok iyi bir sonuç vermemektedir. Saklanacak bilgi, saklama ortamını çok fazla değiştirmeyecek şekilde dikkatlice seçilmelidir. Orijinal görüntüde son bite saklama işlemi yapıldığında, renk girişi göstergeleri değişmektedir. 8 bitlik görüntülerde 4 basit renk (WRBG) kullanılmaktadır. Bunlar; beyaz (W), kırmızı (R), mavi (B) ve yeşildir (G).

Bu renklerin renk paletinde karşılık gelen girişleri ise sırasıyla 0 (00), 1 (01), 2 (10), 3 (11) şeklindedir.

Örnek olarak verilen orijinal görüntü pikselleri "beyaz, beyaz, mavi, mavi" (00 00 10 10) ise 10 sayısının ikilik tabandaki karşılığı olan 1010 değeri bu piksellere gizlendiğinde, yapılan değişiklikler sonucunda görüntünün yeni piksel değerleri aşağıdaki gibi elde edilmektedir.

0**1** 00 1**1** 10

Bu değerler de renk paletinde sırasıyla kırmızı, beyaz, yeşil ve mavi değerlerine karşılık gelmektedir. Piksellerin renk değerleri oldukça değiştiğinden, gözle fark edilebilecektir ve bu kabul edilemez bir durumdur. Veri gizleme uzmanları bu nedenle 8 bitlik renkli görüntüler yerine gri-seviye görüntülerin kullanılmasını daha uygun bulmaktadırlar.

1.4. Ayrık Dinamik Sistemler

Dinamik sistemler, sistemin durumunu belirten bir noktanın geometrik uzayda zamana olan bağımlılığı ile devamlı gelişen sistemlerdir. Herhangi bir zamanda, dinamik bir sistem gerçek sayılardan oluşan bir duruma sahiptir. Bulunduğu durum uzayında o andaki durumu bir noktayla gösterebilir. Dinamik sistemlerin mevcut durumdan nasıl başka bir duruma geçeceklerini açıklayan gelişim kuralları vardır. Bu gelişim kuralları belirlenebilirdir.

Ayrık zamanlı dinamik bir sistem iteratif bir haritalamadır. Tam sayı kümesinin elemanı olan iterasyon sayısı t, sonraki iterasyonlara da atanabilir bir değerdir. Bir X metrik uzayının (X, f) çiftin de f: X→X fonksiyonu, dinamik sistemin ayrık t zamanında durumunu belirlemeyi işaret eder. Burada t doğal bir sayıdır ve ayrık-zamanlı dinamik bir sistemin zamanını gösterir. Dinamik bir sistemin başlangıç koşulundan itibaren durum uzayında izlediği noktalar, o dinamik sistemin yörüngesidir.

1.4.1. Kaos

Kaos kuramcılarına göre kaos; bir durumun değil bir sürecin bilimi [28], bir varoluşun değil bir oluşumun bilimi, sistemlerin global doğasının bilimidir ve karmaşıklığın evrensel davranış biçimini sorgular.

Bu düşünceyi ilk ortaya atanlardan biri olan Feigenbaum'a göre; sistemlerin doğrusal olmayan davranış biçimi çözülmeliydi. Bu konuyla ilgili diğer fizikçiler de Einstein'ın izafiyet (görecelik) kuramının bilinmesinin gerektiğini düşünüyorlardı. Bu teorinin katkısıyla; incelenen doğrusal olmayan gelişmelerin ve türbülansların farklı ölçek ve boyutlarda değerlendirilebilmesi yaklaşımı ve "fraktal geometri" geliştirilmiştir.

J. Gleick kaos için, "Kaosun başladığı noktada klasik bilim durur. Fizik bilimi var olduğundan beri doğanın yasalar; araştırılıp sorgulanmış ancak fizikçiler atmosferde, çalkantılı denizlerde, yabani popülasyonların dalgalanmalarında, kalp ve beynin titreşimlerinde varolan düzensizlik konusuna gelip dayandıklarında dünyayı bu konuların cahili olmaktan kurtaramamışlardır. Doğanın kural dışı olan devamsızlık ve düzensizlik gösteren yüzü, bilim için bir bilmece olarak kalmıştır." yazmıştır.

Mesela bir adada yaşayan belli bir canlı türünün populasyonu, bu canlının üreme hızına, adadaki besin miktarına, bu canlıyla beslenen diğer türlerin adadaki etkinliği vs. gibi birtakım faktörlere belli oranda bağlıdır. Kolaylık olması için bütün bu faktörlerin zamanla değişmediği kabul edilerek modellenen sistemlerde beklenmeyen sonuçlar ortaya çıkmaktadır. Mesela, bir borudan akan sürtünmeli bir sıvının akış şekilleri incelendiğinde önceden tahmin edilemediği görülmektedir. Akışkanlar için kullanılan dinamik denklemler Newton kanunlarından çıkartılıp düzgün akışlara uygulanabilse de, akış hızı belli bir değeri aştığında girdaplar oluşmakta ve akış tamamen kaotik hale gelmektedir.

1.4.2. Kaotik Sistemler

Kaotik sistem, kararlı olarak belli sabit kanunlara göre gelişen bir sistemde beklenmedik şekilde düzensiz davranabilen sistemdir. Zira kaos terimi, günlük yaşamda kullanıldığından farklı olarak, kısmen hesaplanabilen bir karmaşıklık anlamında kullanılmakta. Her sistem veya her hadise, şu veya bu şekilde bir yerlerinde kaotik bileşenler içerir.

Kaotik sistemlerin önemli özelliklerini şöyle sıralayabiliriz [29]:

- Hesaplanamaz olmak: Kaotik sistemlerin belli bir zaman sonra nasıl davranacaklarını tam olarak kestirebilmek imkansızdır. Bunun en bildik örneği, hava durumu tahminleridir. Bir iki gün için yapılan hava tahminlerinde genelde büyük bir sapma olmamasına rağman, hala daha bir aylık ya da yıllık hava tahmini yapmak mümkün değildir. Lorenz'in de hava tahminlerinden görülebileceği gibi sisteme girdiği değerlerde milyonda birlik bir değişim bile tahmin edilemeyen sonuçlar doğurmuştur.
- Başlangıç koşullarına hassas bağlılık: Kelebek etkisi olarak bilinen bu özellik, yukarıda bahsedildiği gibi, Lorenz'in yuvarlayarak bilgisayara girdiği milyonda birlik bir ondalık sayı değişiminin, sistemin davranışında büyük değişikliklere neden olması, sistemin başlangıç koşullarına hassas bağlılığın bir sonucudur. Birçok kaotik sistem, başlangıç koşullarındaki küçük değişimlere çok farklı tepkiler vermeleri ile doğrusal sistemlerden ayrılır. Herhangi bir anda sistem üzerine etki eden tüm etkenlerin bilinmesi imkansız olduğundan, kaotik sistemlerin uzun süreli davranışlarının tahmin edilmesi mümkün olmamaktadır.

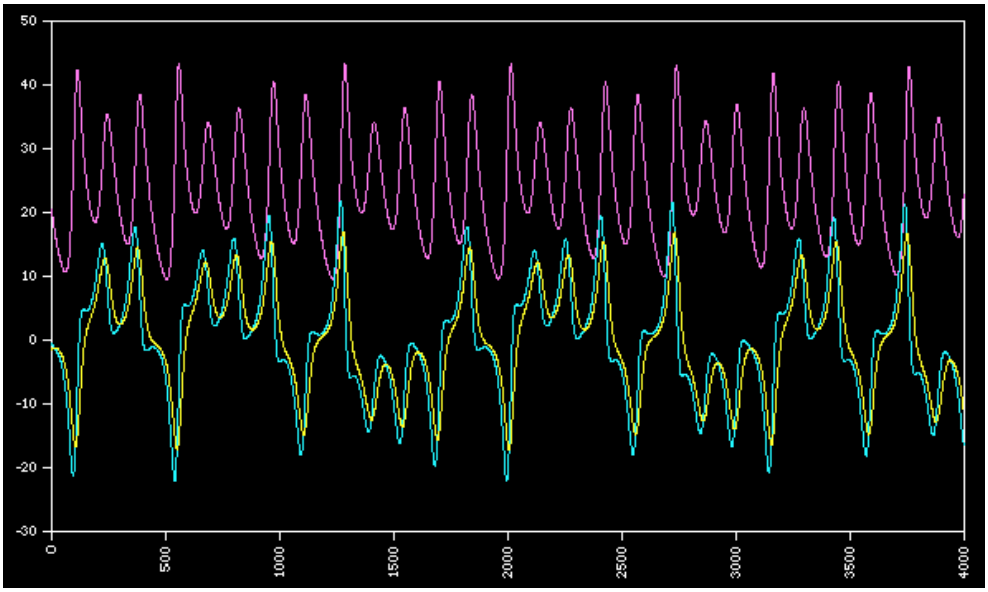
1.4.3. Kaotik Sistemlerin Başlangıç Durumuna Hassas Bağlılığı

Kaosun 1960'lı yıllarda Lorenz ile başladığı söylenebilir. Lorenz kaosun en önemli kavramlarından birini, başlangıç durumlarına hassas bağımlılığını bulmuştur. Ayrıca Lorenz'in çekicisi (Lorenz Attractor) de uzun yıllar kaosu tanımlamıştı.

II. Dünya Savaşı sonunda Massachusetis Teknoloji Enstitüsünde hava tahminleri üzerine bilgisayar destekli araştırmalarda bulunan E. Lorenz, 1960'ta icat ettiği minyatür meteoroloji modeliyle meslektaşlarını şaşırtmıştır. Lorenz ilkel bilgisayarını kullanarak havayı en basit şekilde ifade edilebilen bir hale indirgemiştir. Bilgisayarı, havadaki ısı-basınç ilişkilerini, rüzgarın yönünü, siklon gruplaşmalarını sayısal olarak, her gün 12 denklem yardımıyla kaydediyordu. Rüzgarlar ve hava sıcaklıkları, Lorenz' in yazıcısından satır satır dökülürken dünyadaki gerçekleşme biçimiyle aynı davranışı gösteriyordu. Dahası, bu listelenen değerlerden hareketle tahminlerde de bulunabiliyordu. Biraz uğraşıdan sonra sayıları grafiğe de dökmeyi başardı.

1961'in kış aylarında, bu işi kestirme yoldan yapmak için, makineye önceki rapor değerlerini klavyeden girdi. Ancak rapordaki değerlerin sıfırdan sonraki üç rakamını

yuvarlayarak. Şaşkınlıkla gördü ki, hava durumu, bir önceki dökümde yer alan şekilden umulmadık derecede sapmıştı. Bir süre sonra da dökümle hiç alakasız bir hava raporu grafiği elde etti.



Şekil 1.8. Lorenz'in hava tahminlerini yapmak için kullandığı formüllerin sonucunda elde edilen iki ayrı grafik görülmektedir

Şekil 1.8'de görüldüğü gibi başlangıç değerlerinde yapılan küçük bir değişiklik sonuçta birbirinden farklı grafiklerin oluşmasına sebep olmaktadır. Bu olay, Lorenz'e en küçük detayların bile ileride çok büyük sonuçlara neden olabileceğini hiç bir zaman uzun süreli bir hava tahmini yapılamayacağını göstermişti. Sistem, ilk çıkış noktasıyla çok hassas olarak ilişkiliydi. Bu olayı J.H.Poincare şöyle ifade ediyordu: "Başlangıç şartlarındaki küçücük bir hata son olguda muazzam bir hataya neden olacaktır. Bu durumda, olacağı öngörmek olanaklı değildir."

Kaotik sistemlerde, sistemin zaman içindeki gelişimini tam olarak belirleyebilmek için başlangıç değerlerini sonsuz hassasiyetle bilmek gerekmektedir. Herhangi bir dinamik sistem için de bu geçerlidir, fakat doğrusal (lineer) sistemlerde hata zamanla doğrusal artmasına rağmen kaotik sistemlerde üsse bağlı olarak artmaktadır. Yani doğrusal bir sistemde, mesela bir gezegenin Güneş etrafındaki hareketinde, başlangıçtaki gözlemde yaptığımız hata 1 birim ise, zamanla 2,3,4,... şeklinde artacak; fakat kaotik bir sistemde, mesela atmosferde 10, 100, 1000, 10000,... şeklinde korkunç bir süratle artacaktır. Bunun sonucunda da uzun süreli hava tahminleri kesinlikle mümkün olmayacaktır.

Başlangıç koşullarına hassas bağımlılık fenomeni için en çok kullanılan "Kelebek etkisi" tabiri "Çin'de bir kelebek kanat çırparsa Teksas'ta kasırga çıkar" deyiminden gelmektedir.

1.4.4. Bifurkasyon (Çatallanma-Dallanma) Teorisi

Dinamik sistemler teorisi, temelleri geçen yüzyılda atılmış olan modern bir alandır. Bugün bu alan üzerinde, teorik ve uygulamalı olmak üzere yoğun çalışmalar yapılmakta ve her geçen gün daha da gelişmektedir. Dinamik sistemler teorisini bu kadar çekici kılan mekanikten ekonomiye kadar olan her alanda incelenen sistemlerin matematiksel modellerinin geliştirilmesi ve bu modellerin incelenmesinde dinamik sistemler teorisi kullanılarak, sistemin nitel davranışları hakkında sonuçlar elde edilmesidir. Bifurkasyon teorisi, dinamik sistemler teorisinin bir alanıdır.

Bifurkasyon (Dallanma) olayı doğal bilimlerde çok önemli bir rol oynamaktadır. Doğal bilimlerde nitel hareketi değişen birçok sistem vardır. Nitel hareketin değişimi, sistemin denge noktalarının sayısının veya sistemin karalılığının değişimini ifade etmektedir [30-31]. Bu değişimler sistemin içerdiği parametrelerin kritik değerlerinde meydana gelir. Sistemde değişime neden olan kritik parametre değerlerine bifurkasyon noktaları denir.

Bifurkasyon, sistemin nitel hareketini değiştirmesi olayıdır. Nitel harekete göre bifurkasyonu sınıflandırabiliriz. Sonuç olarak doğal bilimlerde, sistemler genelde bir veya birden fazla parametre içerdiğinden, bifurkasyon problemini içeren birçok olayla karşılaşırız. Örneğin, bir eksen etrafında dönen bir akışkanı düşünelim. Bu akışkanının belli açısal hızlarda daha önceden var olan denge durumunu değiştirerek yeni denge durumları oluşturduğunu, yani hareketini değiştirdiğini gözlemleriz. Burada akışkanın açısal hızını sistemin parametresi olarak düşünebiliriz. Bu durumda akışkanın denge durumunu değiştiren açısal hız değerleri de bu sistemin bifurkasyon noktalarıdır. Bu olay bifurkasyon olayını açıklamak için kullanılan iyi bir örnek olup, bu teorinin oluşturulduğu ilk yıllarda çok incelenmiş olan bir olaydır. Bir çubuğun deformasyonuna neden olan kritik kuvvetlerin analizi de bifurkasyon problemini içeren bir diğer örnektir. Ayrıca kimyasal reaksiyonlar birçok bifurkasyon olayını içermektedir. Buna örnek olarak ani renk değişimlerini verebiliriz. Özetlersek doğal bilimlerde birçok sistemin matematiksel

modelini analiz ettiğimiz zaman bifurkasyon problemi ile karşılaştığımızı görürüz. Sistemlerin matematiksel modelleri genelde bir veya daha fazla parametreye bağlıdır.

Nitel davranıştaki değişikliğe göre bifurkasyon, "pitchfork" bifurkasyonu, "transcritical" bifurkasyonu, "saddle-node" bifurkasyonu gibi sınıflara ayrılmıştır, yani her bir bifurkasyon sınıfında sistemin kritik parametre değerlerinde gösterdiği davranış bifurkasyonun sınıfını belirlemektedir. Sistemin periyodik davranışlarının belli bir parametre değerinde yok olması veya var olması da bir bifurkasyondur. Hopf bifurkasyonu sistemin sabit bir noktasından bifurke eden periyodik çözümlerini inceler [32 - 35].

Dinamik sistemlerde genelde incelenen matematiksel modeller çok büyük veya sonsuz boyutlu uzaylarda olduğunda böyle sistemlerin incelenmesi zor olur. Matematiksel modellerin yerel analizinde, yani belli bir denge noktasının komşuluğunda incelenmesinde, indirgeme yöntemleri çok önemli rol oynamaktadırlar [36]. İndirgeme metodu ile matematiksel model sonlu bir boyuta indirgenmekte veya modelin davranışına katkısı olmayan terimler yok edilmekte ve bunun sonucunda analizi daha kolay olan bir sistem elde edilmektedir. Merkez Manifold (Center Manifold) indirgeme metodu, Normal Form teorisi ve Lyapunov-Schmidt indirgeme metodu dinamik sistemlerde çok kullanılan indirgeme metotlarıdır.

Bir bifurkasyonun Normal Formu ile çözümlerdeki nitel değişimi sergileyecek olan en basit sistem kast edilmektedir [37]. Sistemin normal formuna bakarak sistemin genel nitel davranışı hakkında bilgi edinilebilmektedir. Bu çok büyük boyutlu sistemlerin analizini kolaylaştırmaktadır.

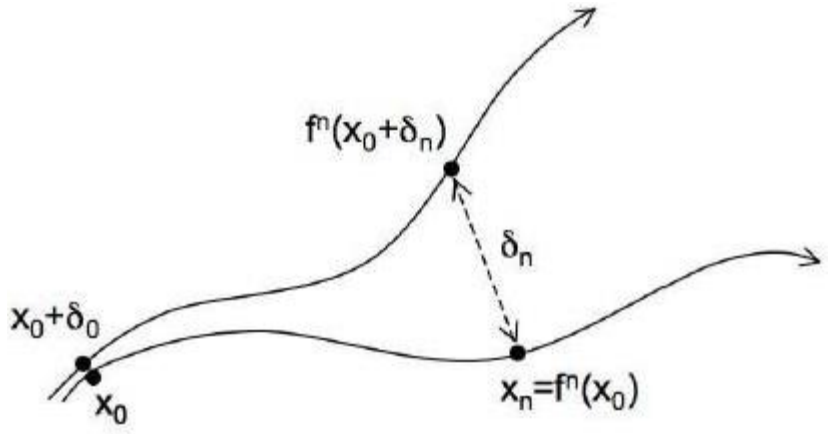
1.4.5. Lyapunov Üstelleri

Bir sistemde belirlenebilir (deterministik) kaos olması onun gelişigüzel görünümüne rağmen, bir kural ya da bağıntıya uyması, yani belirlenebilir olması demektir. Kaotik sistemlerin en büyük özelliği başlangıç koşullarına aşırı derecede bağımlı olmalarıdır. Belirlenebilir olmalarına ve hiçbir gelişigüzellik içermemelerine rağmen gelişigüzel gözükebilmelerinin de sebebi budur.

Kaotik dinamik sistemler başlangıç koşullarına duyarlı sistemlerdir, dinamik sistemlerin başlangıç koşullarına olan duyarlılıklarının ölçümü Lyapunov üstelleri ile ifade edilebilir. Bir dinamik analizci olan Wolf, bu tür sistemlerde başlangıçtaki bilginin üstel bir hızla kaybını ve kestirilebilirliğin ortadan kalktığını, sistemin Lyapunov üstellerden

hareketle ortaya konmuştur. Kaotik bir çözüm veren başlangıç koşullarına çok yakın bir başka grup için çözüm, bir önceki çözümden üstel olarak farklı zaman aralıklarında uzaklaşır. Bu uzaklaşmanın ölçüsü Lyapunov üsteli olup pozitif bir Lyapunov üsteli kaotik, negatifi ise düzgün davranışı ortaya koyar [38].

Lyapunov üstelleri dinamik sistemlerde, komşu yörüngelerin birbirlerine yakınlaşma veya uzaklaşma durumlarını inceleyerek, başlangıç koşullarından uzaklaşma oranlarını Şekil 1.9'daki gibi verir. Faz uzayının boyutu kadar Lyapunov üsteli olup her bir üstel o yöndeki açılma veya büzülmenin ölçüsünü gösterir. Sistemi temsil eden diferansiyel denklem sistemlerinden hareketle değişim hesabı kullanılarak sisteme ait Lyapunov üstellerinin elde edilmesi 1985 yılında Wolf tarafından gerçekleştirilmiştir. İki başlangıç noktası arasındaki uzaklık do olmak üzere, daha sonraki bir zamanda bu uzaklık (1.6)'da verilmiştir. Eğer sistem n boyutlu ise n adet Lyapunov üsteli vardır.

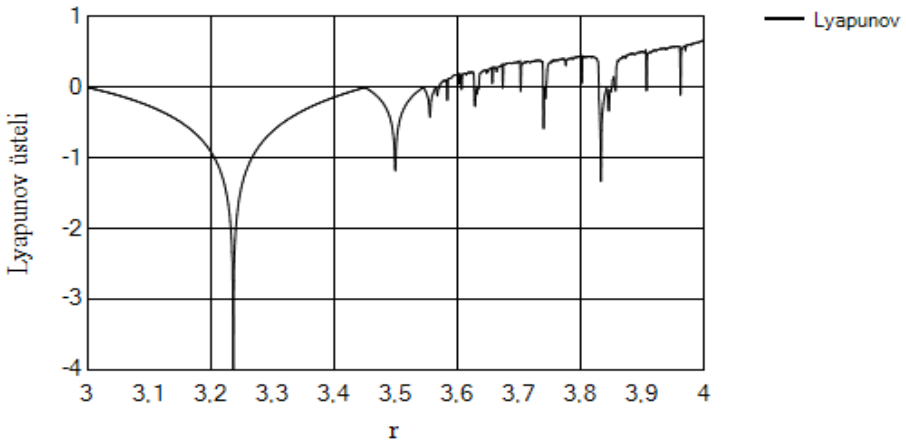


Şekil 1.9. Kaotik bir sistemin yakın noktalarına ait yörüngelerin zamanla uzaklaşması

R gerçek sayılar kümesinde tanımlı f: R->R dinamik sisteminde X_0 ve Y_0 farklı birer nokta ve aralarındaki uzaklık $\delta = |Y_0 - X_0|$ olsun. Bir iterasyondan sonra yeni uzaklık $\delta_1 = |Y_1 - X_1|$ ve Y_1=f(Y_0), X_1=f(X_0), olarak hesaplanır. Eğer Λ'yi $\delta_1 = e^{\delta\Lambda}$ olarak tanımlarsak Λ bir iterasyon sonucunda δ uzaklığından δ_1 uzaklığına olan büyümenin üstel oranını ölçer ve n iterasyon sonunda δ_n n uzaklığı (1.6)'daki gibi hesaplanır.

$$\delta_n = |f^n(x_0) - f^n(y_0)| = \delta e^{n\Lambda} \tag{1.6}$$

Kaotik ortamda Lyapunov üsteli pozitif değere sahiptir. Lyapunov üstelinin değerindeki pozitif artış başlangıç koşullarına olan duyarlılığın daha fazla olduğunu göstermektedir. Lyapunov üstelinin negatif olduğu durumlarda, sistem sabit bir noktaya veya periyodik bir yörüngeye çekilir. Lyapunov üstelinin sıfıra eşit olduğu durumlar ise sistemin kararlı olduğunun göstergesidir. Tüm çok boyutlu kaotik sistemlerin en az bir pozitif Lyapunov üsteli ve çekiciyi sınırlandırmak içinde en az bir negatif Lyapunov üsteli olmalıdır. Çok boyutlu kaotik sistemlerin Lyapunov üstellerinin toplamı negatiftir. Şekil 1.10'da 1 boyutlu lojistik haritanın, farklı r kontrol parametre değerlerine göre Lyapunov üstellerinin grafiği gösterilmektedir. Bu grafikten lojistik haritanın kaotik davranış gösterdiği parametre değerleri gözlemlenebilir. Lyapunov üstelinin pozitif değeri arttıkça kaotik özellikte artmaktadır. Yani sistemin en yüksek Lyapunov üsteli değerine sahip olduğu durum, en fazla kaotik özellik gösterdiği durumdur.

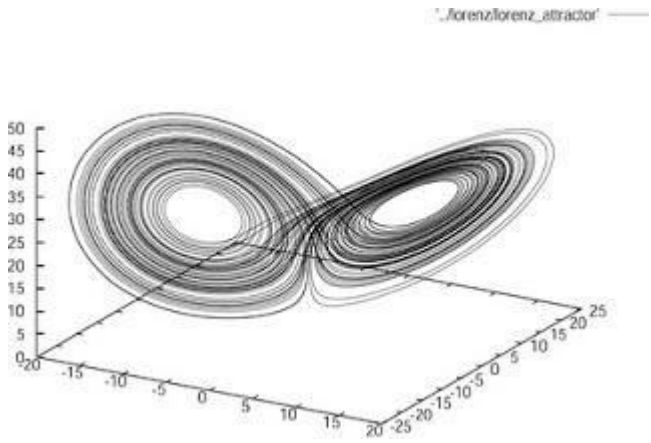


Şekil 1.10. Lojistik haritanın Lyapunov üstellerinin r =[3,4] durumundaki grafiği

1.4.6. Kaotik Haritalar, Garip Çekerler

Garip çekerler (strange attractors), tamamen rastlantısal davranıyormuş gibi gözüken kaotik sistemlerin davranışlarının uzun süreli seyirlerini incelemekte kullanılan özel bir grafik yöntemi olan 'faz uzayı' diyagramlarıdır. Burada ortaya çıkan karmaşık ama düzenli hareket desenleri, halen bu bilim dalı ile yeni tanışanları şaşırtmaya devam ediyor. Çekerler, incelenen sistemin gerçek dünyada doğrudan gözlenemeyen bazı değişkenlerinin zamana karşı nasıl bir dönüşüm geçirdiğini gösteren grafiklerdir. İncelenen sistem "kaotik" özelliklere sahip olduğunda, ortaya çıkan çekerler de "garip" özelliklerinden dolayı "garip

çekerler" olarak adlandırılırlar. Örneğin, Lorenz'in hava durumuna ilişkin ortaya koyduğu matematiksel model, böyle bir grafikle görselleştirildiği takdirde, karşımıza, üç boyutlu uzayda, kanatları yarı açılmış bir kelebeği andıran bir görüntü çıkar (Şekil 1.11).



Şekil 1.11. Üç boyutlu faz uzayında Lorenz'in dinamik sisteminin davranışlarını gösteren Lorenz çekeri

Şekil 1.11'deki desenin anlamı özetle şudur: Hava koşulları tamamen rastgele bileşenlerin etkisi ile oluşuyor gibi görünse de, aslında belli bir sınır dahilinde ve karmaşık dinamik kurallarla hareket eden değişkenlerden oluşur. Siz, herhangi bir anda, hava koşullarının ne olacağını tam olarak kestiremezsiniz; fakat hava koşullarının bu grafiğin izin verdiği şartların dışına taşmayacağını bilirsiniz. Çünkü bu bir davranış grafiğidir ve sistemin belli bir zaman aralığında ve verilen başlangıç koşullarıyla gösterebileceği tüm durumları bir arada temsil eder. Sistem buradaki sınırlar dışına taşamaz. Taşsa bile, bu deseni oluşturan iç kuvvetler o denli güçlüdür ki, sistem yine kendisini bu sınır döngü içine 'çeker'. İşte bundan dolayı, faz uzayı diyagramlarında ortaya çıkan bu tip görüntülere 'çeker' (attractor) adı verilir. Kaotik çekiciler, zaman içinde asla kendini aynen tekrar etmeyen, fraktal karmaşıklığa sahip eşsiz biçimlerdir.

Farklı sistemler, incelenen değişkenlerine göre farklı çeker biçimleri ile karşımıza çıkarlar. Örneğin, düzenli salınan ve sönümlenmeye uğramayan (enerjiyle beslenen) basit bir sarkacı 'sistem' olarak alırsak, sarkacın anlık hızını anlık konumuna göre bir grafiğe döktüğümüzde, ortaya bir 'daire çeker' çıkacaktır. Çünkü bu sistem, belli bir anda, salınımın iki aşırı ucu arasında bir yerde olmak zorundadır ve bu da faz uzayında kapalı bir "daire" ile temsil edilir.

Daha karmaşık sistemlerde ise, daha karmaşık desenler görülür. Örneklerine bolca rastlayabildiğimiz Lorenz, Duffing, Henon veya Rössler çekerleri, böyle karmaşık ve özel şekillerdir.

Garip grafik biçimler olarak "çeker"ler sadece matematiksel soyutlamalardan türetilmiş sistemlerle ilgili değildir. İnsan iradesinin de dahil olduğu gündelik bir çok olayın uzun sürelerle izlenmesi sonucu kaotik bir davranışa sahip olduğu ve belli "çekerler" ile uyumlu davranışlar sergilediği gösterilebilmektedir. Bunların arasında en meşhur örnekler, vahşi hayvan topluluklarının birey sayılarında yıllara göre meydana gelen değişiklikler, yıllar boyu tutulmuş kayıtlardan yola çıkılarak incelenen Nil nehrinin yükselme ve alçalma davranışları, veri aktarım hatlarında meydana gelen gürültülerin periyotları (son ikisi bizzat fraktal geometrinin kurucusu Benoit Mandelbrot tarafından incelenmiştir) ve alınıp satılan kağıtların değerlerinin sürekli dalgalandığı menkul kıymetler borsaları gibi sistemlerin davranışlarıdır [29].

1.4.6.1. Lorenz Çekeri

Lorenz çekeri, 1963 yılında Edward N. Lorenz tarafından, atmosferik olayları açıklamak amacıyla bulunan doğrusal olmayan bir dinamik sistemdir [39]. Bazı parametreler için kaotik yapı sergileyen bu çeker denklemleri, atmosfer çalışmaları ile ortaya çıkmıştır.

Edward N. Lorenz, hava durumunu bilgisayarında modelleyerek, sayısal bir hava durumu tahmin sistemi üzerinde çalışması ile başlayan süreç *Kaotik Sistemlerin Başlangıç Durumuna Hassas Bağlılığı* bölümünde ayrıntılı anlatılmıştır.

Kaotik Lorenz sistemi iki boyutlu akışkan davranışı için ortaya atılan, bilinen en meşhur kaotik sistemdir.

Kaotik Lorenz sistemi (1.7) ile verilmiştir.

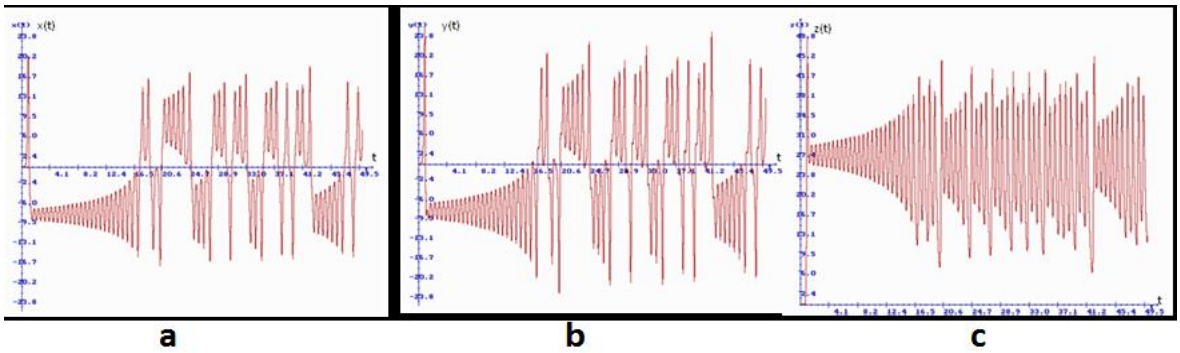
$$dx/dt = -ax + ay$$

$$dy/dt = cx - y - xz \tag{1.7}$$

$$dz/dt = -bz + xy$$

Burada a, b ve c sistem parametreleri, x, y ve z dinamik değişkenlerdir.

Sistemin karakteristik özelliği, spektrumu geniş bir frekans bölgesine yayılmış periyodik olmayan salınımlar üretmesidir. Bu salınımlar gürültüye benzediği ve tahmini zor bir şekilde başlangıç koşullarına bağlı oldukları için gizli haberleşmede kullanılabileceği fark edilmiştir. Lorenz sistemi için x(t), y(t) ve z(t) kaotik dinamiklerinin zamana karşı çizimleri sırasıyla Şekil 1.12'de verilmiştir [40].



Şekil 1.12. Lorenz sistemi kaotik dinamiklerinin gösterimi a) x(t), b) y(t), c) z(t)

1.4.6.2. Lojistik Harita

Lojistik harita, sınırlı bir çevredeki popülasyonu modellemek için geliştirilen 2. dereceden lineer olmayan bir sistemdir. Biyolog Robert May tarafından çok basit bir denklemden kaotik davranış ortaya çıkabileceğinin gösterilmesi amacıyla ortaya atılmıştır[41]. Daha sonra matematikçi Pierre François Verhulst May'in oluşturduğu denkleme benzer ayrık zamanlı lojistik harita denklemini oluşturdu. Matematiksel olarak lojistik harita (1.8)'deki gibi ifade edilir.

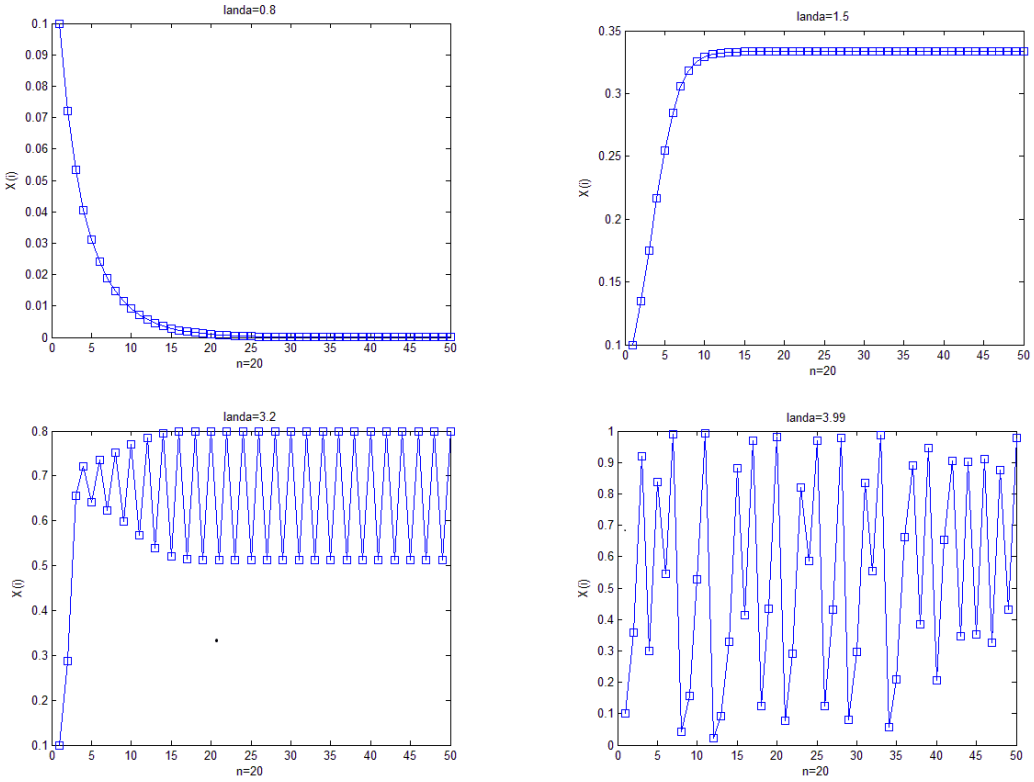
$$X_{n+1} = \lambda X_n(1 - X_n) \tag{1.8}$$

X_n var olan popülasyonun maksimum potansiyel popülasyona oranını gösteren 0 ile 1 arasında bir sayıdır ve mevcut nüfusun bir yıldaki maksimum en yüksek nüfus oranını temsil eder. X_0 başlangıç popülasyonunun maksimum potansiyel popülasyona oranın gösterir. λ reprodüksiyon ve beslenme yetersizliğinin birleşik oranını temsil eden pozitif bir sayıdır. Bu doğrusal olmayan fark denklemi bu iki etki nedeniyle değişim gösterir. Reprodüksiyon, popülasyonun hangi oranda artacağını gösterir. Beslenme yetersizliği

(fazla yoğunluktan ölüm), popülasyonun teorik kapasitesinin aşıldığında ölüm oranını simgeler.

Bu fonksiyonda $X_n \in (0,1)$ başlangıç değeri ve $0 < \lambda < 4$, sistem kontrol değişkeni, n ise yineleme (iterasyon) sayısıdır. Böylece, başlangıç değerleri olarak X_0 ve kontrol değişkeni olarak λ alınarak, $\{X_n\}_{n=0}^{\infty}$ serisi hesaplanır.

Şekil 1.13'te lojistik haritanın kontrol değişkeni olan λ değerine duyarlılığı gösterilmiştir.



Şekil 1.13. a) λ=0.8, b) λ=1.5, c) λ=3.2, d) λ=3.99 değerleri ile üretilen lojistik harita

1.4.6.3. Çadır Haritası

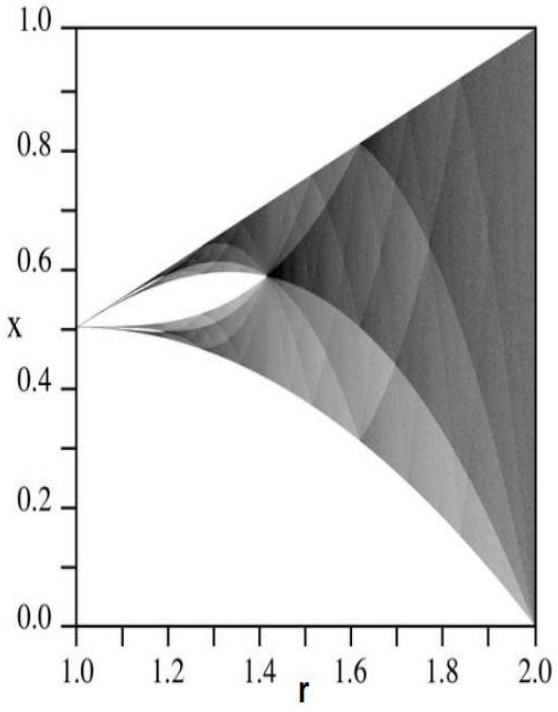
En basit kaotik süreç Çadır Haritası örneğidir. 0 ile 1 arasında bir x0 sayısı seçilir. Daha sonra (1.9)' a göre X_n serisi elde edilir [42].

$$X_n = 2rX_{n-1}, \qquad X_{n-1} \leq 0.5$$

$$X_n = 2r(1 - X_{n-1}), \quad X_{n-1} > 0.5 \tag{1.9}$$

Bu şekilde verilen non-lineer $X_n = f(X_{n-1})$ fonksiyonunun grafiği bir çadırı andırdığından bu isim verilmiştir. Tekrar sayısı arttığında süreç 0-1 aralığını doldurmaktadır [42].

r<1 olduğu durumlarda x değişkeni iterasyonlar sonucunda 0'a yakınsamaktadır. r=1 olduğunda x<=0.5'e olmaktadır. r>1olduğunda, bifurkasyon başlamaktadır. 1 ile 1.5 değerleri arasındaki değerleri için x diyagramı ikiye ayrılmakta, 1.5'ten sonra 4'e ayrılma görünmektedir. r=2 için sistem tam kararsız hal almakta, sadece x değerlerinin maksimum ve minimum noktaları kesin olarak bilinmektedir. Bu durum Şekil 1.14'te verilmiştir.



Şekil 1.14. Çadır haritası'nın bifurkasyon diyagramı

Şekil 1.14.'te görüldüğü gibi çadır haritası topolojik olarak lojistik haritası ile eşleniktir. Lojistik haritasının λ =4 durumu ile çadır haritasının r=2 durumu birbirlerine dönüşebilir.

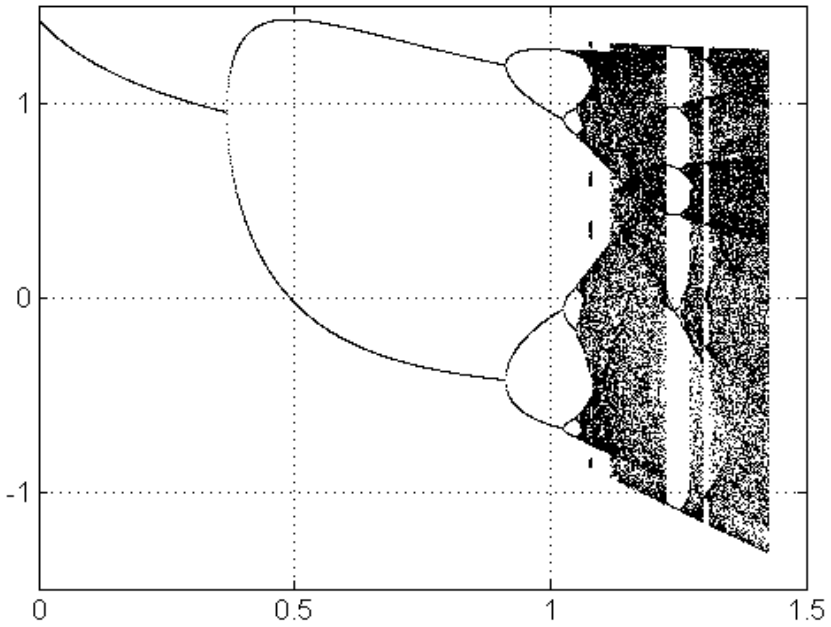
1.4.6.4. Henon Haritası

Michel Henon iki boyutlu bir haritanın kaotik olabileceğini göstermiştir [43]. Denklem (1.10)'da verilen denkleme göre Henon haritası iki boyutlu bir düzlemde $(X_{n,}Y_n)$ noktalarını doğrusal olmayan iki fonksiyon ile yeni bir noktaya taşır.

$$X_{n+1} = Y_{n+1} - aX_n{}^2$$

$$Y_{n+1} = bX_n \tag{1.10}$$

Lojistik haritasındaki gibi Henon haritasında da kaotik olan ve olmayan bölgeler vardır. Sistemin parametreleri olan a ve b'nin bazı değerleri için sistem kaotik yapı göstermektedir. Yapılan analizle sayesinde a=1,4 ve b=0,3 noktalarının kararsız olduğunu belirlenmiştir [43].



Şekil 1.15. Henon haritası bifurkasyon diyagramı

Şekil 1.15'te b=0,3 için a parametresi 0'dan 1,5'a kadar değiştirilmiştir. Henon haritasındaki x değişkeninin aldığı değerler gösterilmektedir. Henon bifurkasyon diyagramında yaklaşık a=0,37 değeri için ikiye ayrılma görülmektedir. Bu değere kadar

iterasyonlar sonucunda x'in alacağı değerlerin yaklaşacağı rakamlar belirlidir. a=0,37 noktasından sonra x ya üst kırılıma ya da alt kırılıma yakınsayacaktır. Kararsız olarak bilinen b=0,3 ve a=1,4 değerlerinde x'in iterasyonlar sonrasında alabileceği değerler kaotiktir.

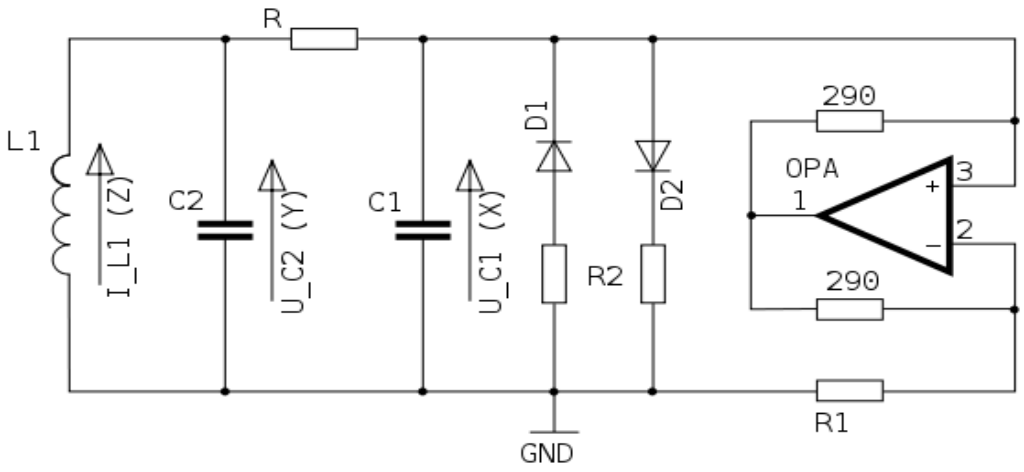
1.4.6.5. Chua'nın Devresi

Laboratuvar koşullarında kaotik davranış oluşturmaya yatkın Lorenz sistemine benzer ilk gerçek fiziksel dinamik sistem 1987 yılında Chua tarafından tanıtılmıştır [44]. Chua, birçok dinamik davranış sergileyen üçüncü dereceden basit bir otonom devre oluşturmuştur. Kaotik Chua devresi, Lorenz denklemlerine göre bazı avantajlara sahiptir. Chua devresi, Lorenz denklemine göre sadece bir değişkenli bir nonlineerlik içerir ve laboratuvar ortamında kolaylıkla oluşturulabilir.

Standard bileşenlerin (dirençler, kapasitörleri indüktörler) oluşturduğu bu devre, kaotik davranış sergileyebilmesi için aşağıdaki kriterlere sahip olmalıdır [45].

- Bir ya da birden fazla doğrusal olmayan element.
- Bir ya da birden fazla yerel aktif direnç.
- Üç ya da üçten fazla enerji depolayan element.

Chua'nın devresi bu kriterlere sahip olan en basit devredir. Şekil 1.16'da Chua'nın devresinin bir örneği ve çekeri görülmektedir. Daha ayrıntılı bilgi için [46]'te verilen Chua ve arkadaşlarının çalışması incelenebilir.

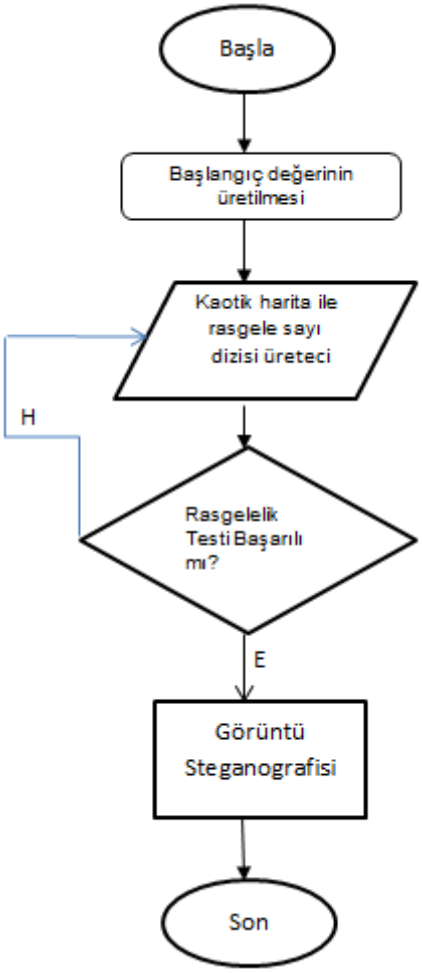


Şekil 1.16. Chua Devresi'nin bir örneği

2. YAPILAN ÇALIŞMALAR

Bu çalışmada steganografinin kaotik haritalarla birlikte kullanımı gerçekleştirilmiştir. Öncelikle lojistik haritanın başlangıç değerinin üretilmesi ve bu başlangıç değerinin karşılıklı taraflar arasında güvenli bir şekilde iletilmesi için asimetrik şifreleme yöntemi olan eliptik eğri kriptografisi kullanılmış, anahtar değişimi Diffie-Hellman protokolü ile sağlanmıştır. Daha sonra bu anahtarın ASCII kodları ile bir başlangıç değeri (2.5)'de verilen denklem ile üretilmiştir. Bu başlangıç değeri ile birlikte sayı üreteci olarak lojistik harita kullanılmıştır (Şekil 2.1). Üretilen sayılar rastgelelik testleri (NIST)'e tabi tutulmuş ve sonuçları verilmiştir. Çalışmanın ikinci kısmında görüntü steganografisi üzerinde çalışılmış, steganografi'nin yer değiştirmeye dayalı yöntemi olarak LSB yöntemi kullanılmış, gizlenecek verinin örtü nesnesi (taşıyıcı)'daki pozisyon seçiminde rastgeleliği kullanmak için lojistik harita kullanılmıştır (Şekil 2.6). Aynı veriler üzerinde sıralı LSB, ayrık logaritma fonksiyonu kullanan LSB yöntemleri de uygulanmış ve bu üç yöntem sonucu elde edilen stego görüntüler (örtü görüntüsü+sır görüntü) taşıyıcıdaki değişim açısından değerlendirilmiştir. Stego görüntüler PSNR değerleri ve histogram korelasyon ve entropy analizlerine tabi tutulmuştur.

Lojistik harita ile rastgele sayı üretiminde tekrarsız üretilen sayılar kısıtlı oluşu (Tablo 2.1) örtü görüntüsü ve saklanacak sır görüntüsünün boyutunu kısıtlamaktadır. Bu probleme çözüm olarak iki farklı kaotik harita kullanımı önerilmiştir.



Şekil 2.1. Lojistik harita ile rastgele sayı üreteci

2.1. Literatürde Kaotik Haritalarla Rastgele Sayı Üretimi

Kaotik birinci mertebeden fark denklemlerinden faydalanılarak ilk Pseudo-random rastgele sayı üreteci 1982 yılında Oishi ve Inoue [47] tarafından önerilmiştir.

Uzun bir aradan sonra, 1993 yılında Lin ve Chua [48] ikinci sıra dijital filtre kullanarak sahte bir rastgele sayı üreteci tasarlanmış ve dijital donanım üzerinde gerçekleşmiştir.

1996 yılında Andrecut [49] sonsuz ve periyodik olmayan bir lojistik rastgele sayı üreteci ile periyodik ve rastgele üreteci tasarımı için bir yöntem önerdi.

1999 yılında Gonzalez ve Pino [50] lojistik haritayı genelleştirilmiş ve rastgele sayı üretimine yardımcı, rastgele sayı üretiminde davranışı kestirilemeyen bir fonksiyon önermiştir.

2001 yılında Kolesov vd., [51] ayrık kaotik - sinyaline dayalı bir dijital rastgele sayı üretici geliştirdi. Önerilen dijital üretec chaotic sinynal sentezinde matris metodu uygulanmıştır.

Ayrıca, Kocarev [52] ve Stojanovski vd., [53], bir kaotik parçalı (piecewise) lineer tek boyutlu harita kullanarak rastgele sayı üreteci uygulamasını analiz etmiştir.

Li ve arkadaşları [54], parçalı lineer kaotik haritaların, uzun döngü boyutu, belirli bir aralıkta denge, yüksek lieer karmaşıklık, iyi korelasyon özellikleri gibi mükemmel kaotik özelliğe sahip olduğuna dair bir teorik analiz yaptı. Ayrıca, tek bir kaotik sistem üzerinden oluşturulan bit dizilerinin potansiyel güvensiz olduğuna, çıkışın (output) kaotik sistemi hakkında bazı bilgiler sızdırabileceğine dikkat çektiler. Bunu aşmak için bağımsız iterasyonlu ve bit dizileri ile bu kaotik haritaların çıkışlarını karşılaştıran ayrık lineer kaotik harita çifti tabanlı bir yöntem önerdiler.

2003 yılında Kocarev ve Jakimoski [55] kaotik haritalar kullanarak rastgele sayı üretiminde farklı olasılıkları tartışmış olasılıkları tartışmış ve aynı zamanda kaos tabanlı rastgele bir bit üreteci önermiştir.

2004 yılında Fu ve ark [56] ayrık kaotik haritayı kullanarak kaos tabanlı rastgele sayı üreteci önerdi.

2005 Li vd., [57] bir boyutlu-lineer harita tabanlı bir rastgele sayı üreteci tasarlayıp analiz yaptı.

Liu [58] tarafından lojistik haritayı modifiye ederek yeni bir, yeni bir yalancı rastgele sayı üretesi (New Pseudo-Random Number Generator-PRNG) üretildi.

2006 yılında Wang vd.[59], tarafından sonlu işlem hassasiyeti aslında, kaotik yörüngelerden geçen z-lojistik harita tabanlı kaotik bir rastgele sayı üreteci önermiştir.

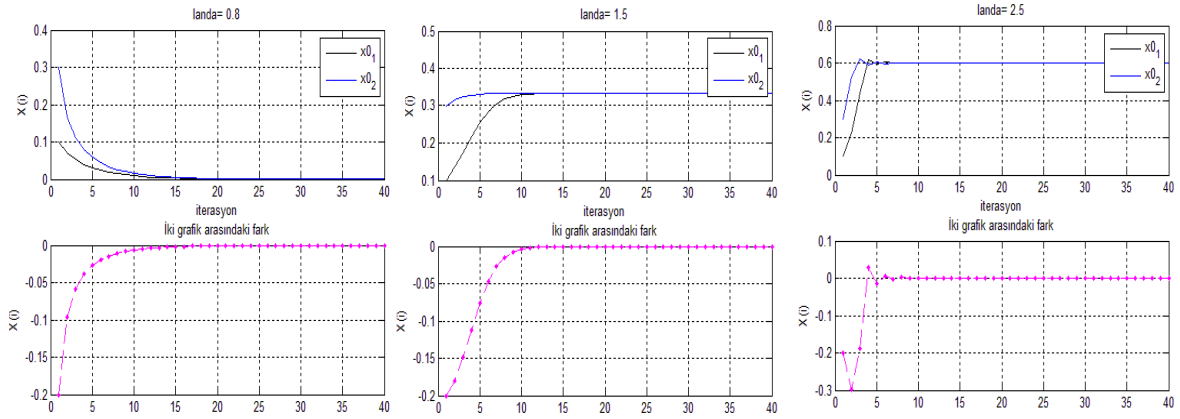
Yakın zaman 2007 yılında Ergun ve Ozogur [60] otonom olmayan kaotik bir elektrik devresinde Poncare haritası ile üretilen bit dizilerinin dört temel FIPS (Federal bilgi işleme standartı) 140-2 testlerinin yanı sıra NIST testlerinden de başarılı olduğunu gösterdi.

2009 yılında Hu vd., [61] bilgisayarın fare hareketleri üzerinde (örneğin bilgisayar fare hareketi tarafından bir 256-bit rastgele sayı üretir), rastgele bir sayı üreteci önerdi. Benzer fare hareket kalıplarının etkisini ortadan kaldırmak için üç tane kaos tabanlı yaklaşım kullanıldı; 2 boyutlu kaotik harita permutasyonu ile üreten, zamansal ve mekansal kaos ve MASK (maskeleme) algoritması. Sonuçlar NIST testleri paketi ile test edilmiştir. Yine aynı yıl Patidar vd., kaotik standart harita tabanlı bir rastgele sayı üreteci

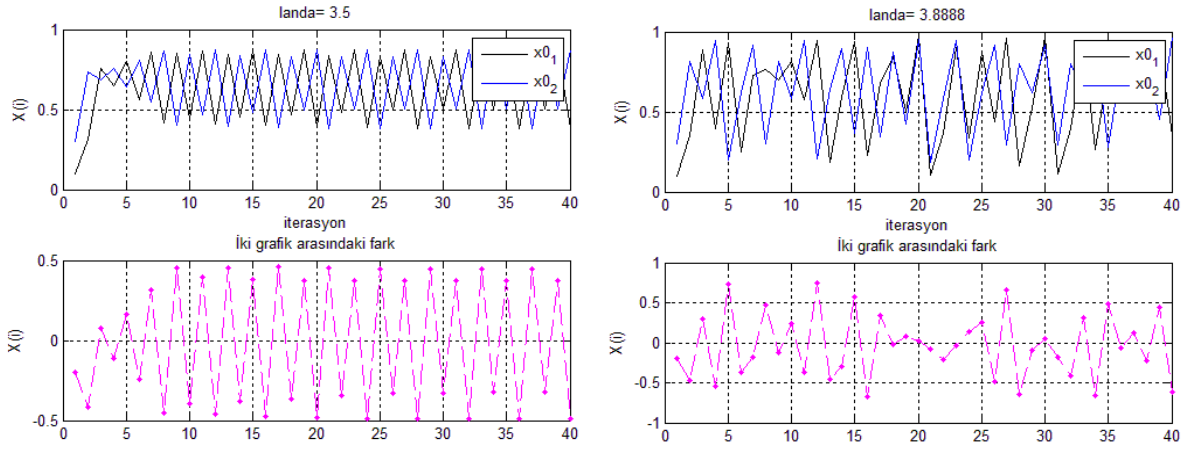
önermiş ve DIEHARD TESTS (fanatik testler) ve NIST testlerinden geçirmiştir. Bu iki testten de hiçbir başarısızlık gözlemlenmemiştir.

2.2. Kaotik Haritalar Kullanılarak Rastgele Sayı Üretilmesi

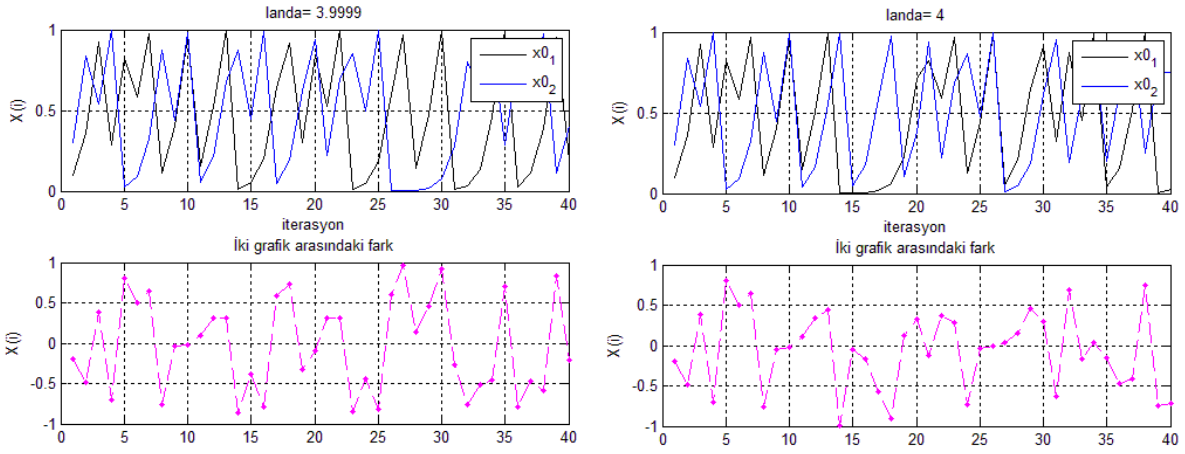
Kaotik Haritalar başlangıç değerlerine hassas duyarlılık gösterdiğinden dolayı sistemin başlangıç değeri ve kontrol parametrelerinin iyi seçilmesi gerekir. Yapılan çalışmada öncelikle lojistik haritalar kullanılmış, farklı başlangıç değerleri ve farklı kontrol değişkenlerine göre sistemin çalışması gözlemlenerek uygun başlangıç değerine karar verilmiştir. Şekil 2.2-2.4'te lojistik haritanın çok küçük değer değişikliklerinde bile sistemin farklı davranışı gösterilmiştir.



Şekil 2.2. Lojistik haritanın $0 < \lambda < 3$ aralığındaki davranışı

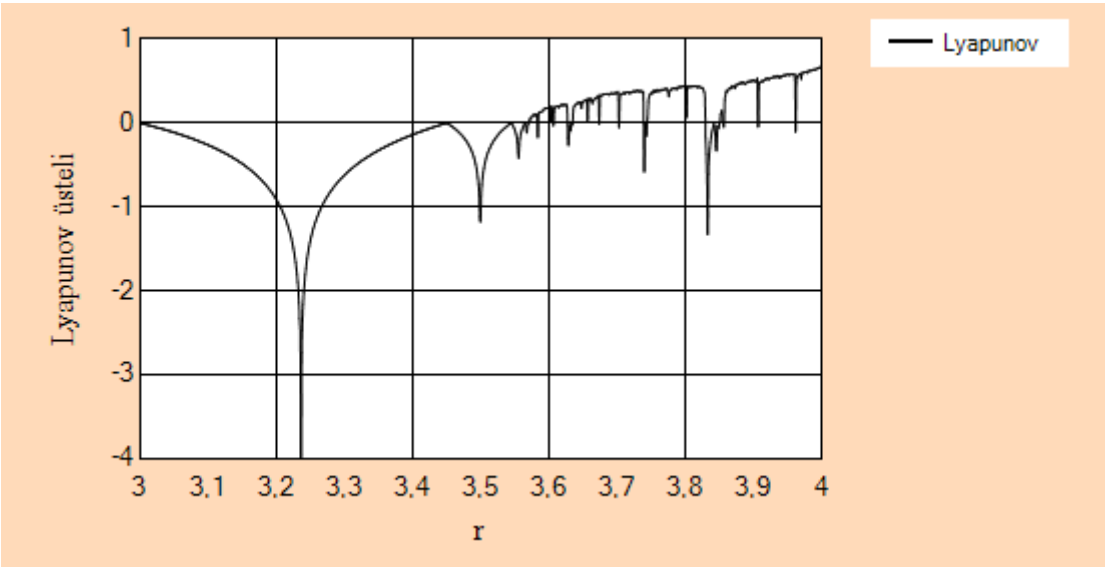


Şekil 2.3. Lojistik haritanın $3.5 < \lambda \leq 4$ arasındaki davranışı



Şekil 2.4. Lojistik haritanın $3.8888 < \lambda \leq 4$ arasındaki davranışı

Şekil 2.2.-2.4.'te lojistik haritanın farklı iki başlangıç değeri ile kontrol değişkeni (λ)'ya olan duyarlılığı ve her bir parametrede başlangıç değerleri değiştirilen sinyallerin fark grafikleri verilmiştir. Şekil 2.2.-2.4.'ten görüleceği üzere lojistik harita λ>3.5 değerinden sonra kaosa girmiş, $x0_1$ ve $x0_2$ diye gösterilen sinyaller arasındaki fark, sistem kaosa girdikten sonra tahmin edilemez bir davranış göstermiştir. Şekil 2.5.'te lojistik haritanın 3<λ≤4 arasındaki Lyapunov üsteli hesaplanmış, burada en yüksek Lyapunov üstünü λ'nın 4'e en yakın olduğu değer vermiştir. Yapılan denemelerden sonra uygulamada kullanılacak lojistik harita için en uygun değer olan λ=3.9999 kullanılmıştır.



Şekil 2.5. Lojistik Haritanın Lyapunov üsteli

2.3. Üretilen Sayıların Rastgelelik Testlerine Tabi Tutulması

2.3.1. Monobit Testi

Literatürde Frekans Testi olarak da geçen monobit testi, bir sistem tarafından üretilen rastgele sayıların (pseudorandom or true random numbers) rastgeleliğini araştıran bir test yöntemidir. Buradaki temel dayanak, rastgele üretilecek değerlerin, sistem tarafından gerçekten rastsal olarak dağılıp dağılmadığını araştırmaktır[62].

Frekans testi aşağıdaki 3 adımdan oluşmaktadır.

1. Rastgele sayı üret.
2. Üretilen sayıyı ikili tabanda hesapla ve bit dizisi oluştur.
3. İstatistiksel fonksiyonu (Denklem 2.1.) uygula.

$$X_1 = \frac{(n_0 - n_1)^2}{n} \tag{2.1}$$

Denklemde n_0 dizideki 0'ların sayısını, n_1 1'lerin sayısını, n toplam bit sayısını ifade etmektedir.

2.3.2. Serial Test

Bu testin amacı 00,01,10,11 gibi alt sıraların rastgele bir dizide beklenenle aynı olup olmadığının belirlenmesidir. Denklem (2.2)'de n_0 ve n_1 sırasıyla 0 ve 1'lerin, $n_{00}, n_{01}, n_{10}, n_{11}$ de sırası ile 00, 01, 10, 11'in toplam sayısını göstermektedir. Burada $n_{00} + n_{01} + n_{10} + n_{11} = n - 1$ olduğuna dikkat edilmelidir.

$$X_2 = \frac{4}{n-1}({n_{00}}^2 + {n_{01}}^2 + {n_{10}}^2 + {n_{11}}^2) + \frac{2}{n}({n_0}^2 + {n_1}^2) + 1 \tag{2.2}$$

2.3.3. Poker Testi

Denklem (2.3)'teki m, $^n/_m \geq 5x2^m$ şartını sağlayan bir pozitif sayıdır.

Teste tabi tutlacak dizi her biri m uzunluğunda birbiri ile örtüşmeyen k adet parçaya bölünür. n_i, m uzunluğundaki i. dizinin toplamda dizideki sayısıdır $(1 \leq i \leq 2^m)$. Poker testi (2.3)'te verilmiştir.

$$x_3 = \frac{2^m}{k}\left(\sum_{i=1}^{2^m} {n_i}^2\right) - k \tag{2.3}$$

2.3.4. Ki-Kare Testi

Bu 2 testte üretilmiş olan rastgele sayı katarının meydana getirdiği örnek dağılım ile teorik tek biçimli dağılım arasındaki uygunluğun derecesi ile ilgilenir. Aslında bu testte rastgele sayıların oluşturduğu dağılım ile teorik dağılım arasında dikkate değer fark olmadığını ifade eden sıfır hipotezine dayanır.

En önemli test dağılımının tek biçimli dağılımlılığı testidir. Bunun için x^2 dağılımı kullanılabilir. Bu test (0,1) aralığında gruplanan örnek bilginin sınıflara ayrılmasına dayanır. X^2 uygunluk testi, bilginin tek biçimli dağılımdan gelmesi halinde her 2 sınıfta gözlenen frekansın, beklenen frekanslara uygun olup olmadığını belirler. Kullanılan test istatistiği şöyledir:

- O_i= i. sınıfta gözlenen sayı
- E_i= i. sınıfta beklenen sayı

$$C = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i} \tag{2.4}$$

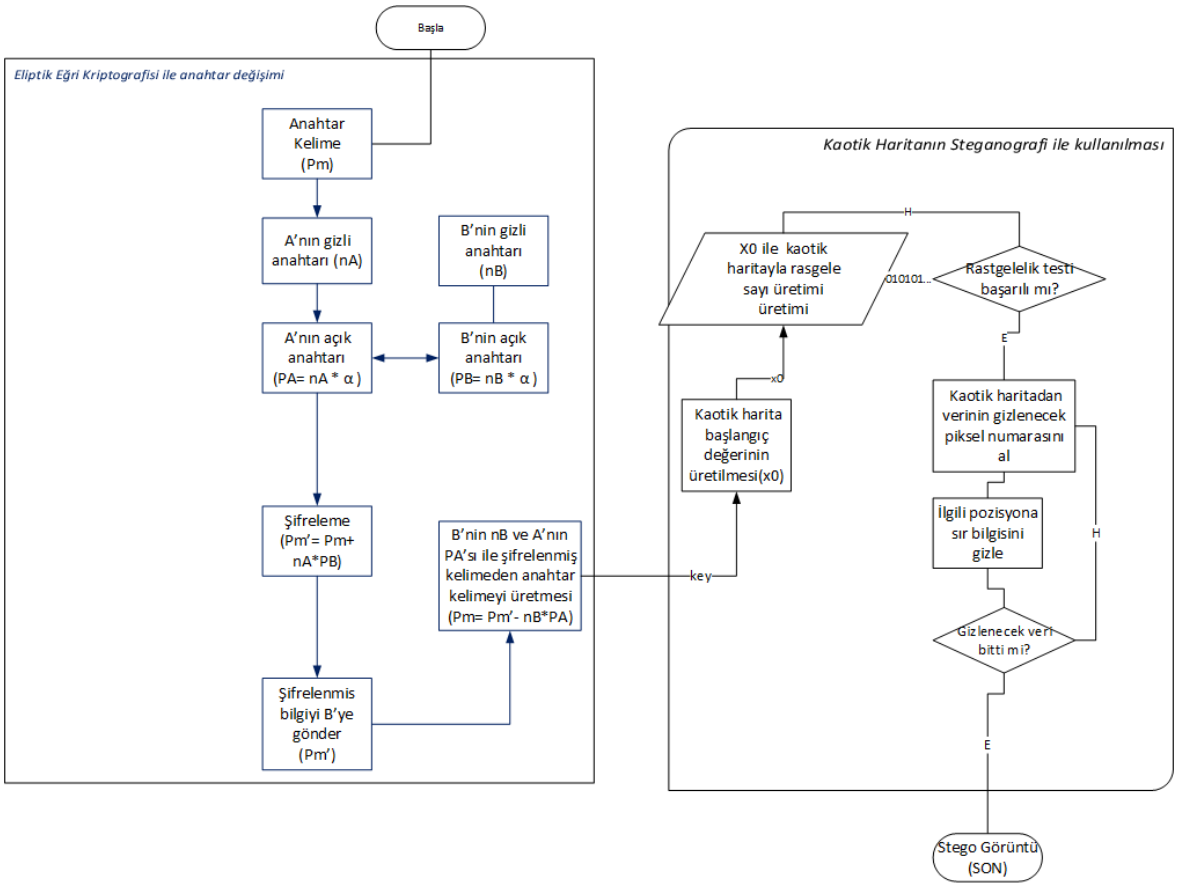
Ki-Kare testi Denklem (2.4)'te gösterildiği gibi hesaplanır ve bu teste ait serbestlik derecesine göre bağımsılık yüzdesi belirlenir. Örneğin sistem Ki-Kare testi sonucu 6.63 değerini üretsin. Tablo (2.1)'e göre sistem %99 bağımsızdır diyebiliriz.

Tablo 2.1. Ki-kare referans değer tablosu

p	Kritik Değerler
0.1	2.71
0.05	3.84
0.01	6.63
0.005	7.88
0.001	10.83

2.4. Kaotik Haritaların Görüntü Steganografisi ile Birlikte Kullanımı

Amaçlanan yöntemin akış şeması Şekil 2.6'da verilmiştir.



Şekil 2.6. Lojistik haritanın steganografi ile birlikte kullanımı akış şeması

2.4.1. Kaotik Haritada Başlangıç Değerinin Belirlenmesi İçin Kullanılan Anahtar Kelimenin Asimetrik Şifreleme ile Karşı Tarafa İletilmesi

Kaotik haritaların başlangıç koşullarına hassas duyarlılığı nedeniyle üretilecek kaotik haritanın başlangıç değerinin belirlenmesi bunu karşılıklı taraflara güvenli bir şekilde iletilmesi için bir anahtar değişim protokolü gerekmektedir. Anahtar kelimeden üretilen X_0 değeri, anahtar olarak belirlenen kelimeyi oluşturan her bir karakterin ASCII değerinin ikilik sistemdeki karşılığına dönüştürülerek $C_1, C_2, \dots, C_n$ şeklinde gösterilir ve daha sonra (2.5)'te gösterildiği şekilde hesaplanır.

$$X_0 = \frac{1}{2^n} * \sum_{i=1}^{n} C_i * 2^{n-i} \tag{2.5}$$

Başlangıç değeri X_0'ın üretilmesi için kullanılacak anahtar kelime karşı tarafa iletilmesi için Diffie-Hellman Anahtar değişimi protokolü ile Eliptik Eğri Kriptografi'si kullanılarak iletilmiştir. Eliptik Eğri Kriptografisi bir asimetrik şifreleme yöntemidir. Bu yöntem 2.4.1.3'te detaylandırılmıştır.

2.4.1.1. Açık (Asimetrik) Anahtarlı Kripto Sistemler

Açık anahtarlı kriptografi, daha önceki kriptografik yöntemlerden bir kopuştur. Açık anahtarlı kriptografik sistemlerin en önemli özellikleri, permutasyondan çok matematiksel işlemleri kullanmasıdır. Açık anahtarlı kriptografik geleneksel tek anahtar kullanan şifreleme yöntemlerinin aksine farklı iki anahtar kullanır. Anahtar dağıtımı gibi gizlilik ve güven gerektiren durumlarda, farklı iki anahtar kullanımı sistemin güvenliğini güçlendirmiştir.

Şekil 2.7. Çift anahtar ile açık anahtar (asimetrik şifreleme)

Şekil 2.7'de, açık anahtarlı şifreleme yöntemi gösterilmiştir. Başlıca adımlar şunlardır [63]:

1. Karşılıklı taraflar mesaj alındığında şifreleme ve şifre çözme için kullanacak olduğu anahtar parçasını yaratır.
2. Her iki taraf, şifreleme için kullanacağı anahtarını karşılıklı birbirine iletir. Bu anahtarın, açık olan kısmıdır (public key). Özel anahtar saklı tutulur.
3. Eğer, A, B'ye bir mesaj yollamak isterse, mesajı B'nin açık anahtarını kullanarak şifreler.
4. B, mesajı aldığında, bu mesajı kendi özel (gizli) anahtarını kullanarak şifreyi çözer.

2.4.1.2. Diffie-Hellman Anahtar Değişimi

Diffie-Hellman anahtar anlaşması, anahtar dağıtma problemine ilk pratik çözümdür. Üs alarak anahtar değiştirme olarak da bilinen bu sistem daha önce hiç haberleşme sağlamamış iki tarafın açık kanal üzerinden mesajlarını birbirlerine göndererek ortak bir anahtar yaratması temeline dayanır. Diffie-Hellman ortak gizli anahtar oluşturma sistemi ayrık logaritma problemini üzerine kurulmuş ve güvenirliği çok büyük asal sayıları seçmeye dayanmaktadır [64].

Diffie-Hellman anahtar anlaşması algoritması

- A, $0 < a < p - 2$ eşitsizliğini sağlayan ve rastgele bir a sayısı seçer. $c = g^a$ değerini bulur ve bunu B'ye gönderir.
- B, $0 < b < p - 2$ eşitsizliğini sağlayan ve rastgele bir b sayısı seçer. $d = g^b$ değerini bulur ve bunu A'ya gönderir.
- A, ortak anahtar k'yı şu şekilde hesaplar:

$$k = d^a = (g^b)^a$$

- B, ortak anahtar k'yı şu şekilde hesaplar:

$$k = c^b = (g^a)^b$$

Böylelikle A ve B aralarında ortak bir anahtar olan k için anlaşmış olurlar.

p yeteri kadar büyük bir asal sayı olsun ve Z^*p de ayrık logaritma problemini çözmek mümkün olmasın. Ayrıca, g Z^*_p' da ilkel bir kök olsun. p ve g herkes tarafından bilinsin. A ve B kişileri aşağıdaki yolu izleyerek ortak bir anahtar yaratabilirler.

Ortak anahtarı oluşturmak için öncelikle p sayısını p=541 ve g sayısını g=2 seçelim. A kişisi kendi gizli anahtarı olan a sayısını, a =137 ve B kişisi kendi gizli anahtarı olan b sayısını, b = 193 olarak belirlesin.

$$c = g^a \pmod{p} \rightarrow 208 = 2^{137} \ (mod\ 541)$$
$$d = g^b \pmod{p} \rightarrow 195 = 2^{193} \ (mod\ 541)$$

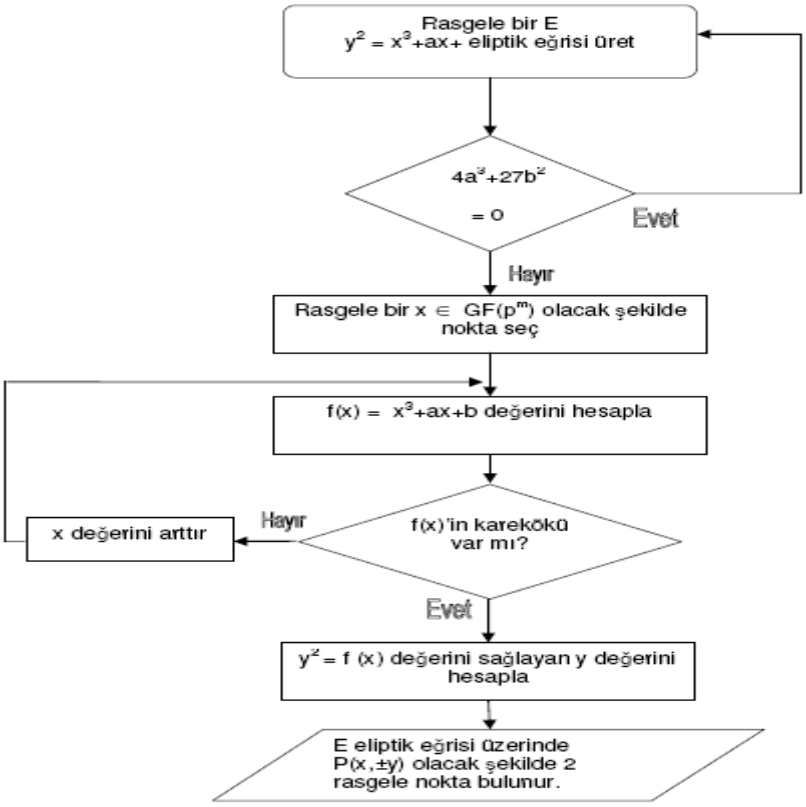
c ve d değerleri hesaplandıktan sonra a ve b kişileri bu değerleri birbirine gönderir ve ortak olan k anahtarı sayısal olarak şu şekilde hesaplanır;

$$k = c^b = (g^a)^b \pmod{p} \rightarrow (2^{137})^{193} (mod\ 541)$$
$$\rightarrow 208^{193} \ (mod\ 541)$$
$$\rightarrow 486 \pmod{541}$$

2.4.1.3. Eliptik Eğri Kriptografi Algoritmasının Açıklanması

2.4.1.3.1. Rastgele Bir Eliptik Eğrinin Oluşturulması

Öncelikle bir p asal sayısı ve $y^2 = x^3+ax+b$ denklemindeki eliptik eğri parametreleri olan a ve b değerleri girilecektir. Daha sonra bu değerlerin doğruluğu kontrol edildikten sonra, eliptik noktalar grubu olan $E_F(a,b)$ oluşturulacaktır. Sonrasında $E_F(a,b)$ içerisinden, başlangıç noktası olacak olan $a=(x_1,y_1)$ seçilecektir. A'nın başlangıç noktası olabilmesi için, bu noktadan yola çıkılarak eliptik eğri üzerindeki tüm noktaları elde etmemiz gerekir. Bu değer de seçildikten sonra, artık $E_F(a,b)$ ve a, kripto sistemin tüm katılımcılarına yayınlanabilir.



Şekil 2.8. Eliptik eğrisi üzerindeki noktaların bulunması

2.4.1.3.2. Eliptik Eğri Üzerinde Yer Alan Bir Noktaya Açık Metnin Gömülmesi

Programda kullanacağımız yöntemin 1/2^K kadar hata verme olasılığı vardır ve *K* genelde *30< K<50* arasında değer alan bir tam sayıdır. Burada *p* asal sayı ve $q=p^r$ nin büyük ve tek sayı olduğunu varsayıyoruz. F_q cismi *q>M*K* olacak şekilde seçilir. Böylece tam sayılar 1'den *M*K* ya kadar *m*K+j, 1< j <K,* şeklinde yazılabilir ve tam sayılarla Fq'nun elemanlar kümesi arasında bir birebir örten fonksiyon oluşturabiliriz.

Verilen bir m metin birimi, her bir *j=1,2,.,K* için, *m*K+j* karşılık gelen bir $\alpha \in F_q$ elemanı elde ederiz. Bir kez α 'yı elde ettiğimizde, $y^2 = f(\alpha)$*'yı* alarak bunu $y^2=x^3+ax+b$ eşitliğinin sağ tarafına gireriz. Sonrada y' yi bulabiliriz. x ve y'yi yerine koyduğumuzda bize $y^2=f(x)=x^3+ax+b$ eliptik eğrisinde $P_m = (x,y)$ noktasını verir. Eğer $y^2 = f(\alpha)$*'*nın p modülüne göre karekökleri yoksa o zaman *j* değerini bir artırıp işlemleri tekrardan yaparak Fq alanında bir *f(a)* değeri bulunur. Bu yaklaşımda dikkat edilmesi gereken husus, *j*'nin değeri K'dan büyük olursa P_m noktasından m metnini elde edilemeyeceğidir.

Örneğin, "ESRA" kelimesini yukarıdaki yöntemi kullanarak eliptik eğri üzerine iz düşürelim.

F_{751} cismi üzerinde E, $y^2=x^3-x+188$ eliptik eğrisi olsun. Bu eğri F_{751} üzerinde N=727 tane noktaya sahiptir. Varsayalım ki düz mesaj üniteleri 0'dan 9'a kadar desimal rakamları ve 10'dan 35'e kadar da A'dan Z'ye kadar olan harfleri simgelesin. K=20 alalım.

Çözüm

Mesaj "14 27 26 10" sayısal eşitliklerine sahiptir. j=1: m=14 ile başlıyoruz ve x = 14*20+1= 281' i deniyoruz. Görüyoruz ki f(x) = 404 mod 751: $=y^2$ dir. Daha sonra f(x)'in mod 751'de karekökünün olup olmadığını bulalım. Gerçekte 404 mod 751'de kökü olmayan bir sayıdır. Bundan dolayı yerine j=2'yi deneriz ve bu sefer x = 282, y^2 =63 buluruz, ki bunlar da mod 751'de kökü olmayan bir sayıdır. Kökü bulana kadar j değerlerini arttırıyoruz, j=3 için x = 283, y^2 = 663 bulunur ki, bunlar da var olan değerlerdir ve karekökü de √663 mod 751 = 54'dir. Bundan sonra $y^2=x^3-x+188$ eğrisinde (283,54) noktası olarak, "E" 'yi yerleştiririz.

Benzer yöntemi kullanarak geri kalan harfleri (543,61), (522,282), (201,5) noktalarına yerleştirmiş oluruz.

2.4.1.3.3. Anahtar Değişimi

Bir *A* ve *B* kullanıcısı arasındaki anahtar değişimi aşağıdaki gibi gerçekleşir:

1. A, n'den küçük bir n_A tam sayısı seçer. Bu A'nın özel anahtarıdır. Daha sonra A, $PA = nA * \alpha$ hesabıyla E^(a,b)'nin bir noktası olan kendi açık anahtarını oluşturur.
2. B'de aynı metotla kendi açık anahtarı PB'yİ oluşturur.
3. *A* gizli anahtarı $K = nAPB$ ile, B'de gizli anahtarı $K = nBPA$ ile elde eder.

Üçüncü aşamadaki iki hesaplamanın sonucu da aynıdır. Çünkü

$n_AP_B = n_A(n_Ba) = n_B(n_Aa) = n_BP_A$ eşitliği mevcuttur.

P_A ve P_B kullanıcıların açık anahtarı, n_A *ve* n_B kullanıcıların özel anahtarıdır.

2.4.1.3.4. Şifreleme

P_m gibi bir mesajı şifrelemek ve bir *B* kullanıcısına göndermek isteyen bir *A* kullanıcısı, rastgele pozitif bir *nA* tam sayısı seçer ve *Cm* şifreli metninin noktalarını aşağıdaki şekilde elde eder:

$Cm = (nA* \alpha, Pm + nA*PB)$

$= (PA, Pm+ K)$

Burada α, (x,y) şeklinde bir koordinatı gösterdiğinden, eliptik eğri üzerindeki koordinat işlemleri şu şekilde yapılır;

Eğer P= (x_1,y_1) ve Q= (x_2,y_2) ise ve P≠ -Q ise, bu durumda, P+Q =(x_3,y_3) şu kuralla hesaplanır;

$$x_3 = m^2 - x_1 - x_2 \quad (\text{mod p})$$
$$y_3 = m(x_1 - x_3) - y_1 \ (mod\ p) \tag{2.6}$$

m için koşul;

$$m = (y_2 - y_1)/(x_2 - x_1) \qquad eğer\ P \neq Q\ ise,$$
$$m = (3{x_1}^2 + a)/2y_1 \qquad eğer P = Q\ ise \tag{2.7}$$

Eliptik eğri üzerinde yer alan bir noktanın k katını almak demek P noktasının k defa toplanması anlamına gelmektedir.

Örneğin, Başlangıç noktasının α = (2,7) olan bir eliptik eğride,

1. A kullanıcısı özel anahtarını belirler.$nA = 3$ olsun.

2. B kullanıcısı özel anahtarını belirler. $nB = 7$ olsun.

3. A kullanıcısının açık anahtarı; $PA = nA * \alpha = 3 * (2,7) = (8,3)$ olur.

4. B kullanıcısının açık anahtarı; $PB = nB * \alpha = 7 * (2,7) = (7,2)$ olur.

5. Şifrelenecek mesaj $Pm = (10,9)$ olsun. (2.4.1.3.2)'de iz düşüm konusu anlatılmıştı.

6. Her kullanıcı karşıdan aldığı açık anahtarla K gizli anahtarı üretsin.

$K = nAPB = 3 * (7,2) = (3,5)$

$K = nBPA = 7 * (8,3) = (3,5)$

7. A kullanıcısı Pm 'yi şifrelesin

$(Pm + K) = (10,9) + (3,5) = (10,2)$ olur.

8. A kullanıcısı karşı tarafa açık anahtarı ile birlikte şifreli mesajı gönderir.

$(PA, (10,2))$

2.4.1.3.5. Şifre Çözme

B kullanıcısı aşağıdaki şekilde mesajın şifresini çözer,

$$(Pm + nAPB) - nB(nA\alpha) = Pm + nA(nB\alpha) - nB(nA\alpha) = Pm \quad (2.8)$$

Bir sonraki aşama metin birimi m'nin E üzerinde bulunduğu *Pm* tanımlama noktasından geri alınmasıdır. Bunu yapmak, *Pm* noktasındaki x koordinatına uyan tam sayının *x* ' olduğu yerlerde $m = [(x'-1) / K]$ formülünü kullanarak çok kolay bir hale dönüşmektedir.

2.4.2. Kaotik Haritalar ile Rastgele Değerlerin Üretilmesi ve Bu Sayıların Steganografide Kullanımı

Bu çalışmada, 1.4.6'da verilen, Lojistik Harita ve Çadır haritası kullanılmıştır. (2.8) ile kullanılan lojistik harita ve kontrol değişkeninin kullanılan aralığı verilmiştir. (2.9)'da kullanılan çadır haritası Skew Çadır haritası'nın denklemi verilmiştir.

$$X_{i+1} = \lambda X_i(1 - X_i) \quad (2.8)$$

$$x_{i+1} = F(\alpha, x_i) = \begin{cases} \frac{x_i}{\alpha} & x_i = [0, \alpha) \\ \frac{1-x_i}{1-\alpha} & x_i = (\alpha, 1] \end{cases} \quad (2.9)$$

(2.8)'deki Lojistik harita fonksiyonunun davranışı λ parametresine oldukça bağlı olup fonksiyonu kaotik bölgede çalıştırmak için $3.57 < \lambda < 4$ olmalıdır. Burada kaotik davranış için $\lambda = 3.9999$ seçilmiştir.

X_0 başlangıç değerleri ile bu denklemlerin tekrarlaması ile değerleri 0-1 arasında değişen gerçek sayılar üretilir. İlk amaçlanan yöntemde steganografide gizli bilginin yerleştirileceği piksel pozisyonunu belirlemek için tek kaotik harita (lojistik harita) kullanılmıştır. Burada lik bir resim dosyası için $N \times M$ piksel olduğundan dolayı, rastgele sayı üreteci ile değerleri 1 - ($N \times M$) arasında değişen sayılar üretilmesi gerekir. Bunun için Lojistik harita ile üretilen değerler NxM' oranında genişletilir.

Örneğin $N \times M$'lik bir resim dosyası için [1 1024] aralığında rastgele sayı üretilmesi gerekir. Bunun için toplam piksel sayısını *N* diye ifade edersek Denklem (2.10)'da verilen lineer transformasyon ile lojistik harita değerleri istenilen aralığa genişletilmiş olur. Bu transformasyondan sonra üretilen sayılar yuvarlanarak (round işlemi ile) tam sayı üretimi sağlanmıştır.

$$\big([0\ \ 1] * (N-1)\big) + 1 = [1\ \ N] \tag{2.10}$$

Burada çok küçük sayılarla işlem yapıldığından, genişletme işlemi sırasında aynı değere iz düşen tekrarlı eleman olması kaçınılmazdır. Rastgele sayı dizisini oluşturan algoritma aşağıda verilmiştir.

Adım 1: λ=3.9999 kontrol değişkeni seçilir.

Adım 2: $X_0 = \frac{1}{2^n} * \sum_{i=1}^{n} C_i * 2^{n-i}$ anahtar değişim protokolü ile karşı taraftan alınan anahtar kelimeden başlangıç değeri hesaplanır.

Adım 3: $X_{n+1} = \lambda X_n(1 - X_n)$, lojistik harita ile bir sonraki değer hesaplanır.

Adım 4: $X_n = X_{n+1}$, bir sonraki iterasyon için X_n değeri güncellenir.

Adım 5: $X_{n+1} = (X_{n+1} * (N-1)) + 1$**,**iterasyon listesine eklemek için lojistik harita ile üretilen değer istenilen oranda genişletilir.

Adım 6: Yeni X_{n+1} iterasyon listesinde yoksa diziye eklenir.

Adım 7: İterasyon listesinde istenilen sayıda eleman olana kadar 3 – 6 adımları tekrarlanır.

Adım 8: İterasyon listesindeki değerlere göre sır görüntüsünü ilgili piksele gizlenir.

İstenilen piksel sayısı kadar rastgele sayı üretimi esnasında tekrarlı elemanlardan dolayı iterasyon sayısı artmaktadır. Tablo 2.2'de lojistik harita ile üretilmek istenen dizi uzunlukları, hesaplamadaki iterasyon sayısı ve hesaplama süreleri verilmiştir.

Tablo 2.2. Lojistik harita ile rastgele sayı üretiminde hesaplama süresi ve iterasyon sayısı

Dizi Uzunluğu (resmin boyutu)	İterasyon Sayısı	Hesaplama Süresi
1024-> 32×32	11856	0.980830 s
4096-> 64×64	43404	7.156542 s
16384->128×128	266272	224.202061 s
*65536->256×256	500000	832.290096 s

* 65536 uzunluğunda dizi üretimi tamamlanamamış, dizi uzunluğu yaklaşık 14 dk. Sürede 65311 olmuş, sonrasında Matlab programı hesaplamayı tamamlayamamış, sistem cevap vermemiştir.

Rastgele sayı üretiminde kaos haritasında tekrarlı eleman elde edilmesinden dolayı resim boyutu olarak çok büyük bir resim sayılmayan 256×256'lık görüntü dosyaları üzerinde çalışılamamıştır. Uygulamada 200×200'lik görüntü dosyaları ile çalışılmış, daha sonra yöntemin bu zafiyetini ortadan kaldırmak için, rastgele sayı üretiminde 2 kaotik harita; Lojistik Harita ve Skew Çadır Haritası kullanılmıştır. Steganografide gizlenecek verinin resim üzerindeki pozisyonu belirlenirken x (satır) pozisyonları Lojistik Harita ile y (sütun) pozisyonları Skew Çadır Haritası ile belirlenmiştir. Böylelikle bir önceki yöntem (tek kaotik haritanın kullanıldığı)'de 256×256'lık resimler üzerinde yöntem çalışmazken, iki kaotik harita kullandığımızda her bir harita ile $128 \times 128 = 16384$ sayı üretilmesi, 16384×16384 görüntü boyutuna ulaşabilmektedir.

Diğer yöntemlerle kıyaslamak için Klasik LSB ve Ayrık Logaritma kullanan LSB yöntemleri ile aynı sır görüntüleri, aynı örtü nesnesine gizlenmiş ve sonuçlar, Deneyler ve Sonuçlar bölümünde verilmiştir.

Steganografi'de LSB'de kullanılan bit sayısı saklama kapasitesini belirlemektedir. Gri seviye bir görüntü üzerinde LSB'de 1 bit kullanıldığında, saklama oranı örtü görüntüsünün 1/8'i kadar olmaktadır. Örten Görüntü ve stego görüntü arasındaki farkların karesel toplamı, veri saklama sonrasında oluşan bozulma oranı hakkında bilgi vermektedir. PSNR hesaplaması Denklem (3.6)'da verilmiştir. Gri seviye bir görüntüde veri saklama sonrasında piksellerdeki değişimin insan gözünün ayırt edemeyecek boyutunda olması yapılan çalışmalarda gri seviye resimlerin kullanılmasına sebebiyet vermiştir.

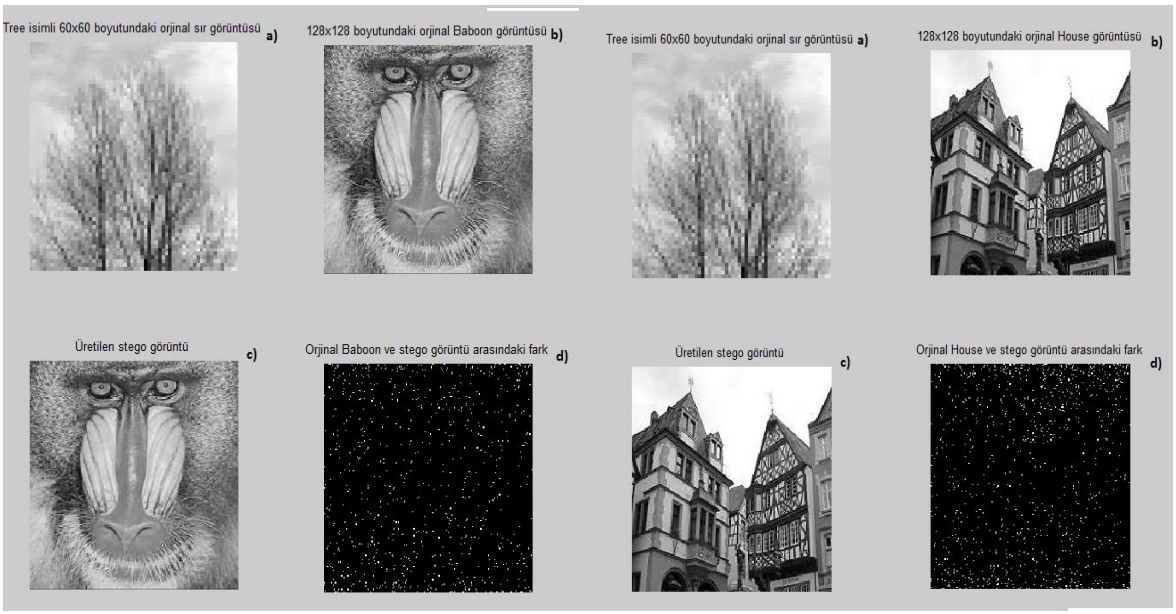
Bu çalışmada sır ve örten görüntü için kullanılan farklı gri seviye görüntüler Şekil (2.9) a-f'de verilmiştir.

Şekil 2.9. a) House, b) Baboon, c) Lena, d) Cameraman, e) Pepper, f) Tree

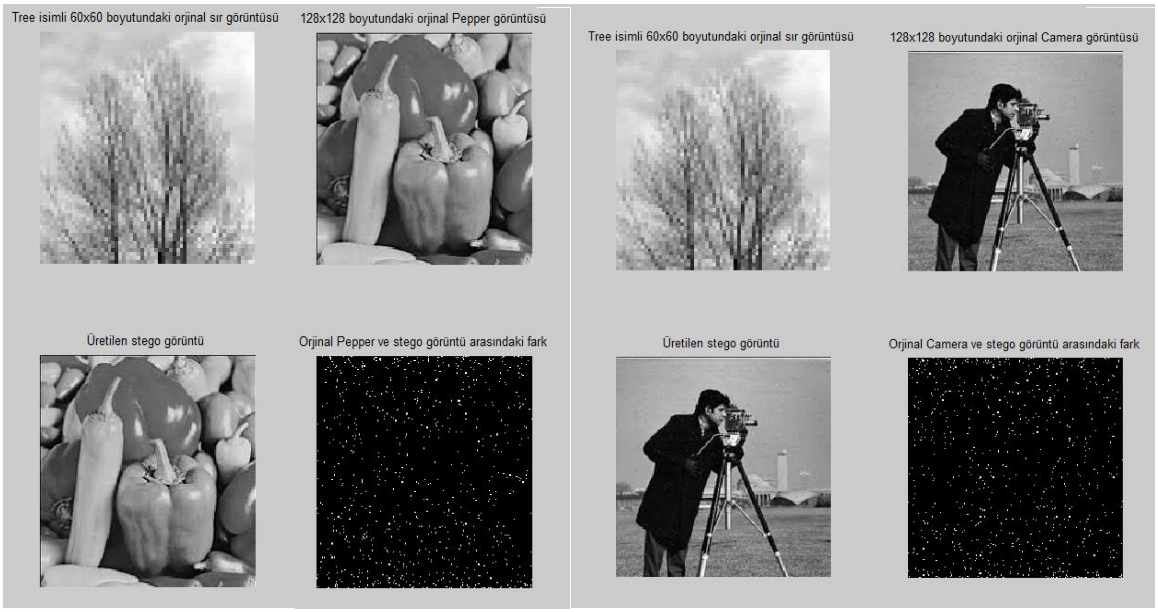
Deneylerden gözlemlenebileceği gibi tek kaotik harita kullanılınca, kaotik harita ile maksimum 61411 sayı üretilmiş, bu da yaklaşık olarak 248×248 boyutunda bir resimin

kullanılabileceği anlamına gelmektedir. LSB'de bir bit kullandığımızdan saklama kapasitemizin örten görüntünün boyutunun 1/8'i kadar olduğuna dikkat edilmelidir.

Şekil (2.10) a-d'de Tek kaotik harita (lojistik harita) ve LSB'de 1 bit'in kullanıldığı yöntemde 200×200 boyutunda örtü görüntüsü ve 60×60 boyutunda sır görüntüleri kullanılmıştır.



(a) (b)



(c) (d)

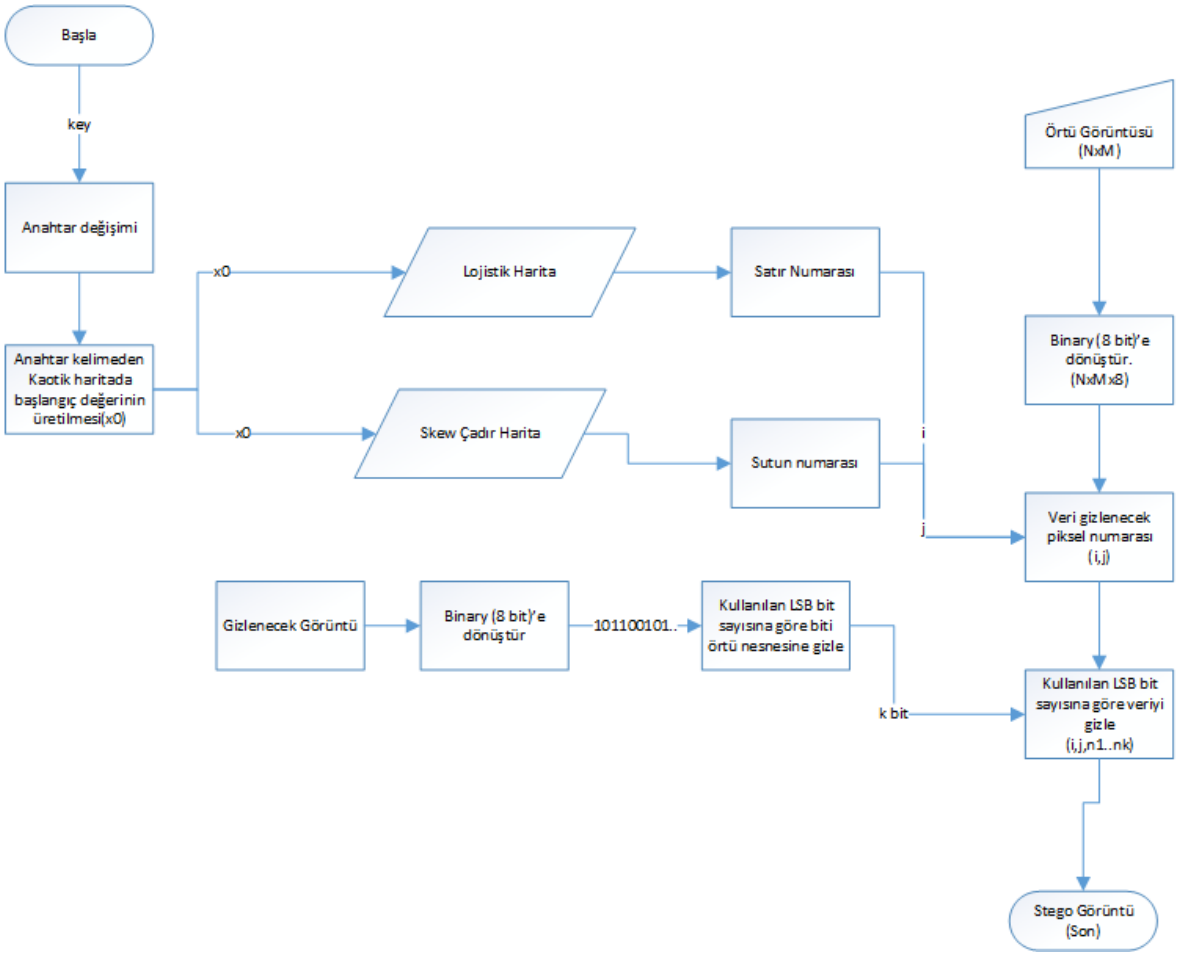
Şekil 2.10. a-b) Baboon ve House isimli örtü görüntüsüne Tree isimli sır görüntüsün tek kaotik harita kullanarak gizlenmesi, c-d) Pepper ve Cameraman isimli örtü görüntüsüne Tree isimli sır görüntüsün tek kaotik harita kullanarak gizlenmesi

Üretilen stego görüntülerin LSB bit kullanımı ve buna karşılık oluşan PSNR değerleri Tablo 2.3'teki gibidir.

Tablo 2.3. Tek kaotik harita kullanımı ile üretilen stego görüntülerin PSNR değerleri

LSB Bit sayısı	LENA (200x200) (PSNR)	BABOON (200x200) (PSNR)	HOUSE (200x200) (PSNR)	PEPPER (200x200) (PSNR)	CAMERAMAN (200x200) (PSNR)
1	52.3642	52.5799	52.6148	52.5382	52.5968
2	48.1519	48.6809	48.4552	48.6563	48.6690

Kullanılan örtü görüntüsünün boyutunun kaos haritasında üretilen değerlerin hesaplama süresine karşın kısıtlı olduğunu dolayısıyla gizlenecek sır görüntüsünün boyutunun da sınırlı olduğundan yukarıda bahsedilmişti. Yöntemin bu dezavantajını ortadan kaldırmak için iki tane kaotik harita kullanılmış (Lojistik Harita- Skew Çadır), böylelikle hem rastgele sayı üretiminde hız artırılmış (tek kaotik harita kullanan yönteme göre), hem de resmin satır ve sütun boyutu için farklı haritalarla sayı üretildiğinden, kullanılabilecek örtü verisinin boyutunun artması sağlanmış olmaktadır. Şekil (2.11)'de iki kaotik harita kullanılarak yapılan steganografi'nin akış şeması verilmiştir.



Şekil 2.11. İki kaotik harita kullanılarak yapılan steganografinin akış şeması

Tablo 2.4. Çift kaotik harita kullanımı ile üretilen stego görüntülerin PSNR değerleri

LSB Bit sayısı	LENA (200x200) (PSNR)	BABOON (200x200) (PSNR)	HOUSE (200x200) (PSNR)	PEPPER (200x200) (PSNR)	CAMERAMAN (200x200) (PSNR)
1	55.5392	55.5408	55.6831	55.5712	55.6023
2	48.0525	48.7274	48.4390	48.6673	48.6870

LSB yönteminde iki tabloda (Tablo 2.3, 2.4)'da görüldüğü üzere çift harita kullanımının PSNR değeri üzerinde önemli ölçüde bir etkisi olmamıştır. Daha öncede anlatıldığı gibi çift harita kullanımının avantajı, daha büyük örtü görüntüsü ve buna bağlı olarak daha büyük boyutlu sır görüntüsü kullanımına imkan sağlamasıdır.

3. BULGULAR VE İRDELEME

Bu bölümde yapılan uygulamanın 2 farklı steganografi tekniği ile kıyaslanmasına yer verilmiştir. Önerilen yöntemin avantaj ve dezavantajları irdelenmiştir. Steganografi tekniğinde en az anlamlı hane (LSB)'ye veri gizleme işlemi için üretilen rastgele sayı dizileri Yapılan Çalışmalar kısmında bahsedilen rastgelelik testlerine tabi tutulup, sonuçları verilecektir.

Yapılan tüm analizler Matlab 7.0 ortamında programlanmıştır. Testler, Intel(R) Core(TM) i7, 2.67 GHz işlemcili ve 8 GB RAM'i olan taşınabilir bir bilgisayar üzerinde gerçekleştirilmiştir. İşletim sistemi olarak Windows 7 Home Premium kullanılmıştır. NIST testleri Ubuntu işletim sistemi üzerinde gerçekleştirilmiştir.

3.1. Üretilen Sayıların Rastgelelik Testlerine Tabi Tutulması

Bu bölümde lojistik harita ve skew çadır haritası ile, farklı başlangıç koşu kümeleri için önerilen yöntem tarafından üretilen diziler için NIST [65-66] tarafından kriptografik uygulamalar için geliştirilen 15 adet istatistiksel test aracı kullanılmıştır. Test sonucu elde edilen p değerlerinin 0.01'den büyük olması teste tabi tutulan bit dizilerinin kabul edilebilir derecede rastgele olduğunu göstermektedir [65]. 1.5.4'te bahsedilen Frekans, Serial, Poker, Ki-kare testleri NIST'in bu 15 testinin içinde bulunduğundan ayrıca değerlendirilmemiştir. Lojistik Harita (X_0=0.4, λ=3.9999) ile üretilen dizinin NIST sonucu Tablo (3.1)'de gösterilmiştir.

Tablo 3.1. Lojistik haritanın NIST testlerine tabi tutulması

TEST ADI	**P_DEĞERİ**	**SONUÇ**
Frequency	0.928730	success
Block Frequency	0.388310	success
Cumulative Sums (Fr.)	0.453914	success
Cumulative Sums (Rv.)	0.525513	success
Runs	0.371187	success
LongestRun	0.985273	success

Tablo 3.1'in devamı

Rank	0.039105	success
FFT	0.304902	success
*NonOverlapping Temp.		success
Overlapping Temp.	0.488416	success
Universal		TEST NOT APPLICABLE
Approximate Entropy	0.000083	failure
Random Excursions		TEST NOT APPLICABLE
Serial p1 value	0.046011	success
Serial p2 value	0.570425	success
Linear Complexity	0.919679	success

Tablo (3.1)'de * ile gösterilen test örnek teşkil etmesi amacıyla ayrıntılı olarak Ek-1'de gösterilmiştir. Serial ve Poker testlerinde anlatıldığı gibi, bit dizisini farklı bit dizilimleri ile karşılaştırarak benzerliklerini araştırır. Burada 9 bitlik dizilerle 147 farklı permutasyon sonucu değerlendirilmiştir. Bu bit dizilerinden 12 tanesi ile örtüşme olmuş ve test sonucunu 'failure' başarısız olarak değerlendirilmiştir. Rastgelelik testinde bu sayıda bir rastgelme olağan bir durumdur. NonOverlapping Template testi 135/147 oranında yani %91.8 oranında başarı vermiştir.

Tablo 3.2. (X_0 =0.4, 0.8) parametreleri ile üretilen Skew Çadır haritasnın NIST test analizi

TEST ADI	**P_DEĞERİ**	**SONUÇ**
Frequency	0.788447	success
Block Frequency	0.681393	success
Cumulative Sums (Fr.)	0.790027	success
Cumulative Sums (Rv.)	0.706611	success
Runs	0.475253	success
LongestRun	0.988355	success
Rank	0.039155	success
FFT	1.000000	success
*NonOverlapping Temp.	%97.9	success
Overlapping Temp.	0.436564	success

Tablo 3.2'nin devamı

Universal		TEST NOT APPLICABLE
Approximate Entropy	0.000079	failure
Random Excursions		TEST NOT APPLICABLE
Serial p1 value	0.079044	success
Serial p2 value	0.716401	success
Linear Complexity	0.919212	success

3.2. Üretilen Stego Görüntülerin İstatistiksel Analizi

A. Kontrast

Bir görüntüde bulunan lokal varyasyonların miktarının ölçüsüdür. Sabit bir görüntü için kontrast 0'dır. Denklem (3.1)'de verilen ifade ile hesaplanır.

$$C = \sum_{i,j} |i-j|^2 p(i,j)^2 \tag{3.1}$$

B. Korelasyon

Bir pikselin tüm resim üzerinde, komşu pikseli ile nasıl ilişkili olduğu hakkında, eş oluşum (co-occurance) matrisini kullanarak verdiği istatistiksel bir ölçümdür. [-1 1] aralığında bir değerdir. Mükemmel pozitif ilişkili resim için 1, mükemmel negatif ilişkili resim için -1 değerini verir. Sabit bir resim için korelasyon değeri tanımsızdır. Korelasyonun hesaplanması için kullanılan ifade Denklem (3.2)'de verilmiştir.

$$\sum_{i,j} \frac{(i-\mu i)(j-\mu j)p(i,j)}{\sigma i \sigma j} \tag{3.2}$$

Verilen ifadede i, j piksel pozisyonunu, p(i,j) i.satır, j. sutundaki piksel değerini, μ varyansı, standart sapmayı göstermektedir.

C. Enerji

GLCM (gray level co-occurance matrix) eş oluşum matrisi unsurların karesi toplamını döndürür. Denklem (3.3)'te verilen ifade ile hesaplanır.

$$E = \sum_{i,j} p(i,j)^2 \tag{3.3}$$

D. Homojenlik

Diyagonal GLCM için GLCM öğelerin dağılımının yakınlığını ölçen bir değer döndürür. Denklem (3.4)'te verilen ifade ile hesaplanır.

$$\sum_{i,j} \frac{p(i,j)}{1 + |i - j|} \tag{3.4}$$

E. Entropy

Entropy rastgele bir süreçte gelen rassal bir değişkenin belirsizliğin büyüklüğüdür. Başka bir deyişle rastgele sayılar arasında belirsiz bir ilişkiyi bulmak demektir. Denklem (3.5)'te verilen ifade ile hesaplanır.

$$\mathrm{H} = \sum_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{p}(\mathrm{x_i}) \log_2 \mathrm{p}(\mathrm{x_i}) \tag{3.5}$$

Verilen ifadede $\boldsymbol{p(x_i)}$, rassal değişken $(\boldsymbol{x_i})$ değerinde olma olasılığıdır**.**

F. PSNR

Doğruluk oranını belirleyebilmek için PSNR değerleri hesaplanmıştır. PSNR değerlerinin hesaplanması için Denklem (3.6)'da verilen ifade kullanılmıştır. Değerlendirme için her üç yöntem tarafından elde edilen stego resimlerin doğruluk oranı dikkate alınmıştır.

$$PSNR = 10\log_{10} \frac{\max(p)^2}{MSE} dB$$
$$MSE = \frac{1}{N \times M} \sum_{i=1}^{N} \sum_{j=1}^{M} \left(p_{ij} - s_{ij}\right)^2 \tag{3.6}$$

3.2.1. Tek Kaotik Harita Kullanılarak Elde Edilen Stego Görüntülerin Analizi

Bu bölümde tek kaotik haritaların steganografi ile birlikte kullanılarak oluşturulan 200x200 boyutunda 4 farklı stego görüntü için istatistiksel güvenlik analiz parametreleri verilmiştir. Ayrık Logaritma Fonksiyonunu Kullanan Steganografi'nin ayrıntıları Ek-2'de verilmiştir.

Tablo 3.3. (200x200) Baboon görüntüsü ile tek kaotik harita kullanılarak elde edilen stego görüntünün diğer yöntemlerle karşılaştırılması

BABOON				
İstatistiksel Güvenlik Analiz Parametreleri	Orijinal Baboon Görüntüsü	Klasik LSB STEGO	Ayrık Log. LSB STEGO	Kaos tabanlı LSB STEGO
Kontrast	0.6702	0.6707	0.6695	0.6707
Korelasyon	0.7730	0.7729	0.7734	0.7732
Entropy	7.4062	7.4055	7.4054	7.4052
Enerji	0.0933	0.0932	0.0933	0.0932
Homojenlik	0.7705	0.7703	0.7705	0.7703

Tablo 3.4. (200x200) Cameraman görüntüsü ile tek kaotik harita kullanılarak elde edilen stego görüntünün diğer yöntemlerle karşılaştırılması

CAMERAMAN				
İstatistiksel Güvenlik Analiz Parametreleri	Orijinal Cameraman Görüntüsü	Klasik LSB STEGO	Ayrık Log. LSB STEGO	Kaos tabanlı LSB STEGO
Kontrast	0.5266	0.5273	0.5281	0.5275
Korelasyon	0.9209	0.9206	0.9205	0.9206
Entropy	7.2089	7.2099	7.2107	7.2106
Enerji	0.1476	0.1471	0.1473	0.1471
Homojenlik	0.8745	0.8737	0.8739	0.8737

Tablo 3.5. (200x200) Pepper görüntüsü ile tek kaotik harita kullanılarak elde edilen stego görüntünün diğer yöntemlerle karşılaştırılması

PEPPER				
İstatistiksel Güvenlik Analiz Parametreleri	Orijinal Pepper Görüntüsü	Klasik LSB STEGO	Ayrık Log. LSB STEGO	Kaos tabanlı LSB STEGO
Kontrast	0.3685	0.3686	0.3688	0.3692
Korelasyon	0.9131	0.9131	0.9130	0.9131

Tablo 3.5'in devamı

Entropy	7.5785	7.57795	7.57815	7.5784
Enerji	0.1282	0.1279	0.1278	0.1278
Homojenlik	0.8840	0.8836	0.8834	0.8834

Tablo 3. 6. (200x200) House görüntüsü ile tek kaotik harita kullanılarak elde edilen stego görüntünün diğer yöntemlerle karşılaştırılması

HOUSE				
İstatistiksel Güvenlik Analiz Parametreleri	Orijinal House Görüntüsü	Klasik LSB STEGO	Ayrık Log. LSB STEGO	Kaos tabanlı LSB STEGO
Kontrast	1.5606	1.5619	1.5577	1.5589
Korelasyon	0.8583	0.8583	0.8585	0.8584
Entropy	7.2409	7.3338	7.3260	7.3275
Enerji	0.0896	0.0893	0.0891	0.0894
Homojenlik	0.7613	0.7606	0.7608	0.7608

3.2.2. İki kaotik Harita Kullanılarak Elde Edilen Stego Görüntüler ve Analizleri

Tablo 3.7. Baboon görüntüsü ile iki kaotik harita kullanılarak elde edilen stego görüntünün diğer yöntemlerle karşılaştırılması

BABOON				
İstatistiksel Güvenlik Analiz Parametreleri	Orijinal BABOON Görüntüsü	Klasik LSB STEGO	Ayrık Log. LSB STEGO	Kaos tabanlı LSB STEGO
Kontrast	0.6702	0.6707	0.6695	0.6700
Korelasyon	0.7730	0.7729	0.7734	0.7733
Entropy	7.4062	7.4055	7.4054	7.4047
Enerji	0.0933	0.0932	0.0932	0.0932
Homojenlik	0.7705	0.7703	0.7707	0.7706

Tablo 3.8. Camera görüntüsü ile iki kaotik harita kullanılarak elde edilen stego görüntünün diğer yöntemlerle karşılaştırılması

CAMERAMAN				
İstatistiksel Güvenlik Analiz Parametreleri	Orijinal Cameraman Görüntüsü	Klasik LSB STEGO	Ayrık Log. LSB STEGO	Kaos tabanlı LSB STEGO
Kontrast	0.5266	0.5273	0.5281	0.5276
Korelasyon	0.9209	0.9206	0.9205	0.9207

Tablo 3.8'in devamı

Entropy	7.2089	7.2099	7.2107	7.2079
Enerji	0.1476	0.1471	0.1473	0.1474
Homojenlik	0.8745	0.8737	0.8739	0.8743

Tablo 3.9. Pepper görüntüsü ile iki kaotik harita kullanılarak elde edilen stego görüntünün diğer yöntemlerle karşılaştırılması

PEPPER				
İstatistiksel Güvenlik Analiz Parametreleri	Orijinal Pepper Görüntüsü	Klasik LSB STEGO	Ayrık Log. LSB STEGO	Kaos tabanlı LSB STEGO
Kontrast	0.3685	0.3686	0.3688	0.3687
Korelasyon	0.9131	0.9131	0.9130	0.9131
Entropy	7.5785	7.5779	7.5781	7.5764
Enerji	0.1282	0.1279	0.1278	0.1280
Homojenlik	0.8840	0.8836	0.8834	0.8837

Tablo 3.10. House görüntüsü ile iki kaotik harita kullanılarak elde edilen stego görüntünün diğer yöntemlerle karşılaştırılması

HOUSE				
İstatistiksel Güvenlik Analiz Parametreleri	Orijinal House Görüntüsü	Klasik LSB STEGO	Ayrık Log. LSB STEGO	Kaos tabanlı LSB STEGO
Kontrast	1.5606	1.5619	1.5577	1.5580
Korelasyon	0.8583	0.8583	0.8585	0.8585
Entropy	7.24094	7.3338	7.3260	7.3282
Enerji	0.0896	0.0893	0.0891	0.0892
Homojenlik	0.7613	0.7606	0.7608	0.7606

Tablo (3.3-3.11)'de, 3 farklı steganografik yöntemle oluşturulan stego görüntülerin ve orjinal görüntülerin istatistiksel değerleri gösterilmektedir. Bu özellikler her görüntünün co-occurence matrisini oluşturduktan sonra çıkarılmıştır.

Görüntü için tablo değerlerine bakıldığında oluşturulan stego görüntülerinin entropy ve korelasyon değerlerindeki değişimler bize görüntü içinde bilgi gizlendiğini ele vermektedir. Üç yöntemde de istatistiksel güvenlik analiz parametrelerinde değişim olduğundan, içinde gizli bilgi sakladığını belli etmeme yönü ile birbirlerine üstünlüklerinden bahsedilemez. Lojistik harita kullanılarak uygulanan steganografide

klasik LSB'deki sıralılık zafiyetine karşı üstünlüğünden söz edilebilir. Yine kendisi gibi rastgele sıralı LSB tekniğini uygulayan Ayrık Logaritma Fonksiyonunu kullanan steganografi tekniği ile karşılaştırıldığında yukarıdaki 5 parametre için yakın sonuçlar verdiği görülmektedir. Önerilen yöntemin Ayrık Logaritma kullanan steganografi'ye göre üstünlüğü karmaşık matematiksel hesaplamalara ihtiyaç duymaması ve bunun da rastgele sayı üretiminde hesaplama kolaylığı sağlamasıdır. Tek kaotik haritanın kullanıldığı yöntem yavaş çalışsa da, iki kaotik haritanın kullanıldığı yöntemde diğer iki yönteme göre (Tek kaotik harita ile steganografi- Ayrık Logaritma kullanan steganografi) hesaplama süresinde hızlanma sağlanmıştır.

4. SONUÇLAR

Bu tezde başlangıç şartlarının bilinmesi durumunda nasıl davranacağı tamamıyla bilinebilen aksi halde ise rasgele davranışlar sergileyen kaotik sistemlerin başlangıç şartlarına aşırı duyarlılık göstermesi ve bu sayede birbirinden farklı çok sayıda rastgele sayı dizisinin elde edilebilmesi yönüyle bu tür sistemlerin ürettiği dizilerle steganografinin birlikte kullanılması amaçlanmıştır.

Kaotik haritalar 0-1 aralığında rastgele değerler üretirler. Kaotik haritalarla veri gizlenecek görüntü için toplam piksel sayısı kadar tam sayıya ihtiyaç duyulduğundan, 0-1 aralığındaki sayıları 1-(NxM) aralığına iz düşürme işlemi esnasında çok küçük sayılarla işlem yapıldığından, tekrarsız tam sayı dizisi üretme esnasında çok fazla tekrarlı elemanla karşılaşma kaçınılmaz olmuş bu da beraberinde çok büyük iterasyon sayısını getirmiştir.

Tablo (2.2)'de gösterildiği üzere 256x256 boyutunda bir örtü görüntüsü için toplamda 65536 sayı yaklaşık 14dk.da üretilememiştir. 128x128 boyutunda bir resim için 16384 tane rastgele sayı yaklaşık 4dk.da üretilmiştir. Toplam piksel sayısı kadar sayı üretiminde kaotik haritalar yavaş çalıştığından görüntü dosyasının satır sayısı ve sütün sayısı için farklı 2 kaotik harita kullanma fikrini doğurmuştur. Böylelikle hem örtü görüntü boyutu hem de gizlenecek veri boyutunda artış sağlanmıştır.

5. ÖNERİLER

Yapılan çalışmada tek kaotik harita kullanarak üretilen rastgele tam sayılar Tablo (2.2)'de verildiği gibi 65311'dir. LSB'de 2 bit kullanarak 128×128 boyutunda bir sır görüntüsünü en az 256×256 boyutunda bir örtü görüntüsüne gizleyebileceğimizden, ürettiğimiz rastgele sayı dizisi uzunluğu buna yetmemektedir.

Gizlenecek verinin resim dosyası değil de metin doyası olduğu düşünülürse $128 \times 128 = 16384$ karakter uzunluğuna sahip bir metin dosyası, 256×256'lık bir resim dosyasına gizlenebilir. Bu da yaklaşık 16 KB boyutunda bir metindosyasına karşı gelmektedir.

Tek kaotik haritaların kullanıldığı yöntem gizlenecek veri olarak görüntü dosyasında efektif bir kullanım sunmamış, veri boyutu küçük boyutta sınırlı kalmıştır. Bu yöntem metin dosyalarının gizlenmesinde daha efektif bir sonuç sağlayacaktır. Görüntü dosyalarının başka bir görüntü dosyasına gizlenmesi için ise iki farklı kaotik haritanın kullanıldığı yöntemin kullanılması önerilmiştir.

6. KAYNAKLAR

1. Cummins, J., Diskin, P., Lau, S., ve Parlett, R., Steganography and Digital Watermarking, http://www.cs.bham.ac.uk/~mdr/teaching/modules03/security/students/SS5/Steganography.pdf 21 Şubat 2014

2. Gustavus J. Simmons, The Prisoners' Problem and The Subliminal Channel, Sandia National Laboratories Albuquerque, NM 87185,1984.

3. Caldwell, 2nd Lt. J., Steganography, CROSSTALK The Journal of Defense Software Engineering, (2003) 25-27.

4. Jamil, T., Steganography: the art of hiding information in plain sight, Potentials, IEEE , 18,1 (1999) 10-12.

5. Samphaiboon N., Dailey M. N, Steganography in Thai Text, Proceedings of ECTI-CON(2008), 133-136.

6. Shirali-Shahreza M. H. ve Shirali-Shahreza M., A New Approach to Persian/Arabic Text Steganography, ICIS-COMSAR'06, USA (2006) 310-315.

7. Meral H. M., Sankur B. ve Özsoy A. S., Türkçe metin belgeleri için damgalama, Signal Processing and Communications Applications-SIU (2006) İstanbul, 1-4

8. Asad M., Gilani J. ve Khalid A., An Enhanced Least Significant Bit Modification Technique for Audio Steganography, ICCNIT IEEE, Pakistan (2011) 143-147

9. Yan, D., Wang R., Huffman table swapping-based steganography for MP3 audio, Multimedia Tools and Applications, 52, 2-3 (2009) 291-305

10. Shirali-Shahreza M. H. ve Shirali-Shahreza M., A New Synonym Text Steganography, International Conference on Intelligent Information Hiding and Multimedia Signal Processing, Iran (2008) 1524-1526

11. Sağıroğlu, S. ve Tunckanat, M., A Secure Internet Communication Tool, Turkish Journal of Telecommunications, 1, 1 (2002) 40-46.

12. Brisbane, G., Safavi-Naini, R. ve Ogunbona, P., High-capacity steganography using a shared colour palette, IEE Proceedings of Vision, Image and Signal Processing, 152 (2005) 787- 792.

13. Lee, Y.K. ve Chen, L.H., High capacity image steganographic model, IEE Proceedings of Vision, Image and Signal Processing, 147, 3 (2000) 288-294.

14. Niimi, M., Noda, H., Kawaguchi, E. ve Eason, R.O., High capacity and secure digital steganography to palette-based images, International Conference of Image Processing, Rochester, New York, USA, 2 (2002) 917-920.

15. Noda, H., Spaulding, J., Shirazi, M.N. ve Kawaguchi, E., Application of bitplane decomposition steganography to JPEG2000 encoded images, Signal Processing Letters, IEEE 9, 12 (2002) 410-413.

16. Shahreza, S., Stealth steganography in SMS, Wireless and Optical Communications Networks, IFIP International Conference, April 2006 , Bangalore, India, 5.

17. Srinivasan, Y., Nutter, B., Mitra, S., Phillips, B. ve Ferris, D., Secure Transmission of Medical Records Using High Capacity Steganography, Proceedings of the 17th IEEE Symposium on Computer-Based Medical Systems, June 2004, Bethesda, Maryland, 122-127.

18. Tseng, H.W., Chang, C.C., Steganography using JPEG-compressed images, Computer and Information Technology. The Fourth International Conference, September 2004, Wuhan, China, 12- 17.

19. Amin, M.M., Salleh, M., Ibrahim, S., Katmin, M.R. ve Shamsuddin, M.Z.I., Information Hiding Using Steganography, Telecommunication Technology, January 2003, Shah Alam, Selangor, 21- 25

20. Johnson, N.F., Jajodia, S., Exploring Steganography: Seeing the Unseen, IEEE Computer, 31, 2 (1998) 26-34.

21. Erkin, Z., Design Of A Steganographic Library, Yüksek Lisans Tezi, İ.T.Ü. Fen Bilimleri Enstitüsü, Trabzon, 2005.

22. Bender W., Gruhl D., Morimoto N. ve Lu A., "Techniques for data hiding", IBM Systems Journal, 35, 3-4, 1996.

23. Pitas I., Nikolaidis N., Copyright Protection of Images using Robust Digital Signatures, IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 4 (1996) 2168-2171.

24. Marvel L. M., Boncelet C. G. ve Retter C. T., Spread Spectrum Image Steganography, IEEE transactions of Image Processing, 8 (1999) 1075 - 1083

25. K. Satish, T. Jayakar, Tobin C., K. Madhavi ve K. Murali, Chaos Based Spread Spectrum Image Steganography, Consumer Electronics, (2004) 587-590

26. Youail R. S., Samawi V. W., Kadhim A. A-R., Combining a Spread Spectrum Technique with Error-Correction Code to Design an Immune Stegosystem, Anti-counterfeiting, Security and Identification, 2008, 245-248.

27. Agrawal N. ve Gupta A., DCT Domain Message Embedding In Spread-Spectrum Steganography System, Data Compression Conference , (2009) 433

28. Ruelle D., Rastlantı ve Kaos, Deniz Yurtören, TÜBİTAK Popüler Bilim Kitapları, Ankara, 2004.

29. http//www.kasisar.org/2012/05/kaos-karmasklk-bilimi-ve-yeni-bilimsel.html. 20 Ocak 2014

30. Stone, E. ve Campbell, S.A., Stability and Bifurcation Analysis of a Nonlinear DDE Model for Drilling, J. Nonlinear Science, 14(1) (2004) 27-57.

31. Kım, S., Campbell, S. A. ve Liu, X. Z., Stability of a Class of Linear Switching Systems with Time Delay, IEEE Transactions on Circuits and Systems I, 53,2 (2006) 384-393

32. Golubıtsky, M. ve Langford, W. F., Classification and Unfoldings of Degenerate Hopf Bifurcations, Journal of Mathematical Analysis and Applications, 41 (1981) 375-414

33. Chow, S.N. ve Hale, J.K., Methods of Bifurcation Theory, Springer-Verlag, New York, Berlin, Heidelberg, Tokyo, 1982.

34. Hassard, B. D., Kazarinoff, N. D. ve Wan, Y. H., Theory and Applications of Hopf Bifurcation, London Mathematical Society Lecture Note Series, 41 (1983)

35. Golubıtsky, M. ve Langford, W. F., Classification and Unfoldings of Degenerate Hopf Bifurcations, Journal of Mathematical Analysis and Applications, 41 (1981) 375-414

36. Hale, J. K., Magalhaes, L. T. ve Oliva, W. L., Dynamics in Infinite Dimensions, 2nd ed., Springer-Verlag, New York, 2002.

37. Faria, T. ve Magalhaes, L. T., Normal Form for Retarded Functional Differential Equations and Applications to Bogdanov-Takens Singularity, Journal of Differential Equations, 122 (1995) 201-224

38. Wolf, A., Swift, J.B., Swinney, H.L. ve Vastano, J.A., Determining Lyapunov Exponents from a Time Series, Physica D-Nonlinear Phonema, Elsevier, 16 (1985) 285-317.

39. Lorenz, E. N., Deterministic Nonperiodic Flow, Journal of Atmospheric Sciences, 20 (1963) 130-141

40. Yardım, F.E. ve Afacan, E., Lorenz Tabanlı Diferansiyel Kaos Kaydırmalı Anahtarlama (DCSK) modeli kullanılarak Kaotik Bir Haberleşme Sisteminin Simülasyonu, Gazi Üniv. Müh.Mim. Fak. Der. 25:1 (2010) 101-110.

41. May, R.M., Simple mathematical models with very complicated dynamics. Nature, 261 (1976) 459-465.

42. Crampin, M. and Heal, B., On the Chaotic Behavior of the Tent Map, Teaching Mathematics Applications, 13,2 (1994) 83-89.

43. Henon. M., A Two-dimensional Mapping with a Strange Attractor, Communications in Mathematical Physics, Springer-Verlag, 50 (1976) 69-77.

44. Chua, L.O., Desoer, C. A., Kuh, E. S., Linear and Nonlinear Circuits, McGraw Hill, New York, 1987.

45. Madan, R. N., Chua's circuit: a paradigm for chaos, World Scientific Publishing Company (1993) 725-739.

46. Chua, L. O., Matsumoto, T. ve Komuro, M., The Double Scroll, IEEE Transactions on Circuits and Systems, CAS-32 (1985) 798–818.

47. Oishi S., Inoue H., Pseudo-random number generators and chaos. Transactions of the Institute of Electronics and Communication Engineers of Japan E, 65 (1982) 534-541.

48. Lin T., Chua L. O., New class of pseudo-random number generator based on chaos in digital filters. International Journal of Circuit Theory and Applications, 21 (1993) 473-480.

49. Andrecut M., Logistic map as a random number generator, International Journal of Modern Physics B, 12 (1998) 921-930.

50. Gonzalez J. A., Pino R., Random Number Generator Based on Unpredictable Chaotic Functions. Computer Physics Communications, 120 (1999) 109-114.

51. Kolesov V. V., Belyaev R. V. ve Voronov G. M., A Digital Random-Number Generator Based on the Chaotic Signal Algorithm. Journal of Communications Technology and Electronics, 46 (2001) 1258-1263.

52. Stojanovski T. ve Kocarev L., Chaos-Based Random Number Generators, Part I: IEEE Transactions on Circuits and Systems I:Fundamental Theory and Applications, 48 (2001) 281-288.

53. Stojanovski T., Pihl J. ve Kocarev L. Chaos based random number generators - Part II: Practical realization. IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, 48 (2001) 382-385.

54. Li, S., Mou, X. ve Cai, Y., Pseudo-Random Bit Generator Based on Couple Chaotic Systems and İts Application in Stream-Ciphers Cryptography. Progress in Cryptology – INDOCRYPT 2001, Lecture Notes in Computer Science, 2247 (2001) 316–329.

55. Kocarev L. ve Jakimoski G., Pseudorandom Bits Generated by Chaotic Maps. IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, 50 (2003) 123-126.

56. Fu S. M., Chen Z. Y. ve Zhou Y. A., Chaos Based Random Number Generators. Computer Research and Development, 41 (2004) 749-754.

57. Li X. M., Shen H. B. ve Yan X. L., Characteristic Analysis Of A Chaotic Random Number Generator Using Piece-Wise-Linear Map. Journal of Electronics and Information Technology, 27 (2005) 874-878.

58. Liu, J., Design Of A Chaotic Random Sequence and Its Application, Computer Engineering, 31 (2005) 150-152.

59. Wang, L., Wang F.P ve Wang Z.J., Novel Chaos-Based Pseudo-Random Number Generator, Acta Physica Sinica, 55-8 (2006) 3964-3968.

60. Ergun S. ve Ozoguz, S., Truly Random Number Generators Based on a Non-Autonomous Chaotic Oscillator. AEU-International J. Electronics & Communications, 62 (2007) 235-242.

61. Hu Y., Liao X., Wong K.W. ve Zhou Q., A True Random Number Generator Based on Mouse Movement and Chaotic Cryptography. Chaos Solitons and Fractals, 40 (2009) 2286-2293.

62. Menezes A.J., Oorschot P.C.V. ve Vanstone S.A., Handbook of Applied Cryptography, CRC Press, Boca Raton, 1997.

63. Bölücek, A. Ç., Akçam, N., Güvenilir, Hata Düzeltmeli ve Dijital İmzalı Protokol Geliştirme, Ulusal Elektronik İmza Sempozyumu, Aralık 2006.

64. Diffie W., Hellman M.E., New directions in cryptography, IEEE Transactions, 22-6 (1976) 644–654.

65. Rukhin, A., Soto, J., Nechvatal, J., Smid, M., Barker, E., Leigh, S., Levenson, M., Vangel, M., Banks, D., Heckert, A., Dray, J. ve Vo, S., Test Suite for Random and Pseudo-random Number Generators for Cryptographic Applications, Special Publication Revision 1a, National Institute of Standards and Technology, (2010) 800-22

66. Sadique J. K. M., Zaman, U. ve Ghosh, R,, Review on fifteen Statistical Tests proposed by NIST, Journal of Theoretical Physics and Cryptography, l1 (2012).

67. Şahin, A., Buluş, E. ve Sakallı, M.T., Gri Seviye Resimler Üzerinde Rasgele LSB Yöntemini ve Sayı Teorisini Kullanarak Bilgi Gizleme ve Steganaliz, Akademik Bilişim Konferansları, Şubat 2006.

7. EKLER

Ek 1. NonOverlapping Template Matching Method (Örtüşmeyen Şablon Eşleştirme Yöntemi)

Burada 9 bitlik dizilerle 147 farklı permutasyon sonucu değerlendirilmiştir. Bu bit dizilerinden 12 tanesi ile örtüşme olmuş ve test sonucunu ‘failure’ başarısız olarak değerlendirilmiştir. Rastgelelik testinde bu sayıda bir rastgelme olağan bir durumdur. NonOverlapping Template testi 135/147 oranında yani %91.8 oranında başarı vermiştir.

Ek Tablo 1- NonOverlapping Template Matching Testi

NONPERIODIC TEMPLATES TEST

COMPUTATIONAL INFORMATION

LAMBDA = 0.472656 M = 250 N = 8 m = 9 n = 2000

F R E Q U E N C Y

Template	W_1	W_2	W_3	W_4	W_5	W_6	W_7	W_8	Chi^2	P_value	Assignment	Index
000000001	1	0	0	0	0	0	0	0	3.901802	0.865873	SUCCESS	0
000000011	1	0	1	0	0	0	0	0	4.017648	0.855528	SUCCESS	1
000000101	0	1	0	0	0	2	0	0	8.370166	0.398171	SUCCESS	2
000000111	3	0	0	0	1	0	0	0	16.959354	0.030535	SUCCESS	3
000001001	1	0	1	0	1	0	1	1	4.365188	0.822764	SUCCESS	4
000001011	0	0	0	0	0	2	2	0	12.722683	0.121754	SUCCESS	5
000001101	0	0	1	1	0	1	0	1	4.249341	0.833955	SUCCESS	6
000001111	1	0	0	0	3	0	0	0	16.959354	0.030535	SUCCESS	7
000010001	0	0	0	0	0	0	0	0	3.785956	0.875900	SUCCESS	8
000010011	1	2	2	0	2	1	1	1	17.654432	0.023972	SUCCESS	9
000010101	0	2	1	0	0	0	1	0	8.486012	0.387489	SUCCESS	10
000010111	0	1	0	0	0	1	1	0	4.133495	0.844883	SUCCESS	11

Ek 1’in devamı

000011001	0	0	0	0	1	1	0	0	4.017648	0.855528	SUCCESS 12
000011011	0	0	0	2	0	0	0	2	12.722683	0.121754	SUCCESS 13
000011101	0	0	1	1	0	1	0	0	4.133495	0.844883	SUCCESS 14
000011111	1	0	0	0	1	0	0	0	4.017648	0.855528	SUCCESS 15
000100011	0	1	0	1	0	1	0	0	4.133495	0.844883	SUCCESS 16
000100101	0	1	1	0	0	0	0	1	4.133495	0.844883	SUCCESS 17
000100111	2	1	0	1	2	0	0	1	13.070222	0.109456	SUCCESS 18
000101001	1	1	0	0	0	0	0	0	4.017648	0.855528	SUCCESS 19
000101011	2	0	1	0	0	1	1	0	8.601859	0.376987	SUCCESS 20
000101101	0	0	1	0	0	2	2	0	12.838529	0.117527	SUCCESS 21
000101111	0	0	0	0	0	0	2	0	8.254319	0.409030	SUCCESS 22
000110011	0	0	1	0	0	1	0	0	4.017648	0.855528	SUCCESS 23
000110101	0	0	0	0	0	2	0	0	8.254319	0.409030	SUCCESS 24
000110111	0	0	0	3	1	0	0	2	21.427717	0.006094	FAILURE 25
000111001	0	0	0	0	0	0	0	0	3.785956	0.875900	SUCCESS 26
000111011	1	0	1	0	0	2	0	0	8.486012	0.387489	SUCCESS 27
000111101	0	0	0	0	2	0	0	0	8.254319	0.409030	SUCCESS 28
000111111	2	0	1	1	1	0	0	0	8.601859	0.376987	SUCCESS 29
001000011	0	0	0	0	0	0	0	2	8.254319	0.409030	SUCCESS 30
001000101	0	0	0	1	1	0	0	0	4.017648	0.855528	SUCCESS 31
001000111	0	2	0	0	0	1	0	1	8.486012	0.387489	SUCCESS 32
001001011	0	1	1	0	0	0	0	0	4.017648	0.855528	SUCCESS 33
001001101	0	1	3	0	1	1	1	0	17.306893	0.027067	SUCCESS 34
001001111	0	0	0	0	2	0	0	1	8.370166	0.398171	SUCCESS 35
001010011	2	1	0	0	0	0	1	3	21.543564	0.005835	FAILURE 36
001010101	0	1	0	0	0	0	3	0	16.959354	0.030535	SUCCESS 37
001010111	2	0	1	0	0	0	1	0	8.486012	0.387489	SUCCESS 38

Ek 2. Ayrık Logaritma Fonksiyonunu Kullanan Steganografi

(E.1)'de verildiği gibi tanımlanan ayrık logaritma fonksiyonu resim içine rasgele şekilde veri gizlemeyi sağlar [67].

$$y_i = a^i \ (mod\ p) \qquad \text{(E.1)}$$

Bu fonksiyonda;

- y_i mesajın i. bitinin resmin içinde saklanacağı pozisyonu;
- i gizlenecek mesajın bit indeksini göstermektedir.
- Buradaki p büyük bir asal sayı ve a ise p'den üretilen asal bir köktür.
- a değeri üsler şeklinde yazıldığında 1'den $p-1$'e kadar tüm tam sayıları verecek şekilde seçilmelidir.
- Yani p ile a kendi aralarında asal olmalıdırlar.
- p değerinin asal olmasının nedeni aynı değerin tekrar üretilmemesidir.

Gizlenecek metnin uzunluğu m, içine veri gizlenecek resmin büyüklüğü l ise p değeri, $m<p<l$ şartını sağlamalıdır.

Aşağıda veri gizleme işleminin algoritması gösterilmektedir.

Adım 1: $m<p<l$ şartını sağlayan en büyük asal sayıyı (p) seç.

Adım 2: p'nin asal elemanları sayısını $(Ø)$ bul.

Adım 3: Asal elemaları üretmek için en küçük böleninden başlayarak üsler şeklinde yazıldığında 1'den (p-1)'e kadar tüm tam sayıları veren böleni bul.

Adım 4: OBEB(i, p-1) =1 şartını sağlayan i değerlerini bul.

Adım 5: (mod p)'ye göre $bölen^i$ değerlerini hesapla ve büyük olanlardan birini a olarak seç.

Adım 6: $y_i = a^i \ (mod\ p)$ denklemine göre mesajın bitlerinin hangi piksele yerleşeceğini bul.

Ek 2'nin devamı

- Adım 1:
  - Örnek olarak $m<p<l$ şartını sağlayan p değerimizi 17 olarak seçelim.
- Adım 2:
  - p değerimizin kaç tane asal elemanı (φ) olduğunu hesaplayalım.
  - Bunun için öncelikle $p-1$ değerini çarpanlarına ayrılır ve üslü şekilde yazılır.
  - $p-1$'in çarpanları; $16=2x2x2x2=2^4$
  - $\varphi = 2^4-2^3=8$ adet asal elemanı vardır.
- Adım 3:
  - p değerinin asal elemanlarını bulmak için en küçük böleninden başlayarak üsler şeklinde yazıldığında 1'den $p-1$'e kadar tüm tam sayıları veren böleni bulunur.
  - Bölen olarak 2 seçildiğinde 1'den $p-1$'e kadar tüm tam sayıları vermemiş ve aynı zamanda tekrarlar olmuştur.
  - 3 seçildiğinde ise istenen şart sağlanmaktadır.
  - *mod 17*'ye göre 3^i değerleri p ile asaldır.

Ek 2'nin devamı

$2^0 = 1 \mod 17$	$3^0 = 1 \mod 17$
$2^1 = 2 \mod 17$	$3^1 = 3 \mod 17$
$2^2 = 4 \mod 17$	$3^2 = 9 \mod 17$
$2^3 = 8 \mod 17$	$3^3 = 10 \mod 17$
$2^4 = 16 \mod 17$	$3^4 = 13 \mod 17$
$2^5 = 15 \mod 17$	$3^5 = 5 \mod 17$
$2^6 = 13 \mod 17$	$3^6 = 15 \mod 17$
$2^7 = 9 \mod 17$	$3^7 = 11 \mod 17$
$2^8 = 1 \mod 17$	$3^8 = 16 \mod 17$
$2^9 = 2 \mod 17$	$3^9 = 14 \mod 17$
$2^{10} = 4 \mod 17$	$3^{10} = 8 \mod 17$
$2^{11} = 8 \mod 17$	$3^{11} = 7 \mod 17$
$2^{12} = 16 \mod 17$	$3^{12} = 4 \mod 17$
$2^{13} = 15 \mod 17$	$3^{13} = 12 \mod 17$
$2^{14} = 13 \mod 17$	$3^{14} = 2 \mod 17$
$2^{15} = 9 \mod 17$	$3^{15} = 6 \mod 17$
$2^{16} = 1 \mod 17$	$3^{16} = 1 \mod 17$

- Adım 4:
  - Bir sonraki adım *OBEB(i,p-1)=1* şartını sağlayan sayıları bulmaktır.
  - Bu değerler bulunduktan sonra 3^i'de karşılık gelen değerler hesaplanır ve bu bulunan değerlerden biri (tercihen büyük olanı) *a* değeri olarak seçilir.

Ek 2'nin devamı

- Adım 5:
  - OBEB'i 1 çıkan sayılar 1, 3, 5, 7, 9, 11, 13 ve 15'tir. *mod 17*'ye göre 3^i'de karşılık gelen değerleri aşağıdaki gibi hesaplanır.

- $3^1 \pmod{17}=3$
- $3^3 \pmod{17}=10$
- $3^5 \pmod{17}=5$
- $3^7 \pmod{17}=11$
- $3^9 \pmod{17}=14$
- $3^{11} \pmod{17}=7$
- $3^{13} \pmod{17}=12$

$3^{15} \pmod{17}=6$

a değerleri arasından büyük olanın seçilmesi daha uygundur. burada 14 değeri seçilebilir.

$y_i = 14^i \bmod 17$ *denklemine göre gizli mesaj resmin içine yerleşir.*

ÖZGEÇMİŞ

Esra ODABAŞ YILDIRIM; 1985 yılında Erzurum'da doğdu. İlk ve Orta öğrenimini Erzurum'da tamamladı. 2003 yılında Nevzat Karabağ Anadolu Öğretmen Lisesi'nden mezun oldu. 2004 yılında Karadeniz Teknik Üniversitesi Bilgisayar Mühendisliği Bölümü'nde lisans programına başladı ve 2010 yılı Ocak ayında mezun oldu. Yine aynı yıl bahar döneminde Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı'nda yüksek lisans çalışmalarına başladı. 2010 Temmuz ayında Atatürk Üniversitesi Bilgisayar Mühendisliği Bölümünde Dekanlığa bağlı Araştırma Görevlisi olarak göreve başladı. 2013 yılı Temmuz-Eylül ayları arasında tez ile ilgili araştırma-inceleme yapmak için Amerika'da UTSA (University of Texas at San Antonio) üniversitesine görevli olarak gitti. Yabancı dil olarak İngilizce bilmektedir. Başlıca yayınları aşağıda verilmiştir.

- Odabas Yildirim E., Ulutas M., Görsel Kriptografi'nin Kaos Tabanlı Steganografi ile Birlikte Kullanılması, International Conference on Electronic, Computer and Automation Technologies, (May 2014), Konya.