T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ŞİFRELİ VE ŞİFRESİZ VİDEOLAR İÇİN YİNELEMELİ HİSTOGRAM DEĞİŞTİRME TABANLI TERSİNİR VİDEO DAMGALAMA

DOKTORA TEZİ

İbrahim YILDIRIM

Enstitü Anabilim Dalı	**:**	**ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ**
Enstitü Bilim Dalı		**ELEKTRONİK**
Tez Danışmanı	**:**	**Prof. Dr. Cabir VURAL**

Şubat 2017

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ŞİFRELİ VE ŞİFRESİZ VİDEOLAR İÇİN YİNELEMELİ HİSTOGRAM DEĞİŞTİRME TABANLI TERSİNİR VİDEO DAMGALAMA

DOKTORA TEZİ

İbrahim YILDIRIM

Enstitü Anabilim Dalı	:	**ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ**
Enstitü Bilim Dalı	:	**ELEKTRONİK**

Bu tez 2 / 2 /2017 tarihinde aşağıdaki jüri tarafından oybirliği/~~oyçokluğu~~ ile kabul edilmiştir.

Prof. Dr.
Cem ÜNSALAN
Jüri Başkanı

Prof. Dr.
Cabir VURAL
Üye

Doç. Dr.
Mehmet Kemal GÜLLÜ
Üye

Doç. Dr.
Kürşat AYAN
Üye

Yrd. Doç. Dr.
Gökçen ÇETİNEL
Üye

BEYAN

Tez içindeki tüm verilerin akademik kurallar çerçevesinde tarafımdan elde edildiğini, görsel ve yazılı tüm bilgi ve sonuçların akademik ve etik kurallara uygun şekilde sunulduğunu, kullanılan verilerde herhangi bir tahrifat yapılmadığını, başkalarının eserlerinden yararlanılması durumunda bilimsel normlara uygun olarak atıfta bulunulduğunu, tezde yer alan verilerin bu üniversite veya başka bir üniversitede herhangi bir tez çalışmasında kullanılmadığını beyan ederim.

İbrahim YILDIRIM
17.01.2017

TEŞEKKÜR

Doktora eğitimim süresince değerli bilgi ve deneyimlerinden yararlandığım, her konuda bilgi ve desteğini almaktan çekinmediğim, araştırmanın planlanmasından yazılmasına kadar tüm aşamalarında yardımlarını esirgemeyen, teşvik eden, titizlikle beni yönlendiren değerli danışman hocam Prof. Dr. Cabir VURAL'a teşekkürlerimi sunarım.

Haklarını hiçbir zaman ödeyemeyeceğim, şefkat ve desteklerini her zaman özleyeceğim annem Şehnaz YILDIRIM ve babam Hayrettin YILDIRIM'a teşekkür ederim. Keşke bu satırları birlikte okuyabilseydik.

Varlığı en büyük motivasyon kaynağım, kızım Elif Betül YILDIRIM'a, destek, sabır ve metaneti için sevgili eşime teşekkür ederim.

Çalışmalarım boyunca maddi ve manevi destekleriyle beni hiçbir zaman yalnız bırakmayan başta kardeşlerim ve yeğenlerim olmak üzere tüm aileme teşekkür ederim.

Çalışma arkadaşlarım ve dostlarım Can YÜZKOLLAR, Dr. Burhan BARAKLI ve Dr. M. Kenan ERKAN'a teşekkür ederim.

İÇİNDEKİLER

SİMGELER VE KISALTMALAR LİSTESİ

BPP	: Piksel başına bit
DBS	: Damgalanan blok sayısı
FG	: Fark genişletme
HD	: Histogram değiştirme
HDÇA	: Hareket dengelenmiş çerçeve aradeğerleme
HK	: Hedef kapasite
KKTD	: Kendi kendini tersinir damgalama
LSB	: En düşük anlamlı bit
ÖHG	: Öngörü hatalarının genişletilmesi
PSNR	: Tepe-işaret gürültü oranı
SSIM	: Yapısal benzerlik indeksi
ŞÖBO	: Şifreleme öncesi boşluk oluşturma
ŞSBO	: Şifreleme sonrası boşluk oluşturma
TD	: Tersinir damgalama
TGD	: Tersinir görüntü damgalama
TK	: Taşma yaşanan piksel konumları
TKS	: Taşma konumları sayısı
TŞD	: Tersinir şifreli damgalama
TŞGD	: Tersinir şifreli görüntü damgalama
TŞVD	: Tersinir şifreli video damgalama
TVD	: Tersinir video damgalama
YHD	: Yinelemeli histogram değiştirme

ŞEKİLLER LİSTESİ

TABLOLAR LİSTESİ

ÖZET

Anahtar kelimeler: Tersinir damgalama, tersinir video damgalama, şifreli uzayda tersinir video damgalama, yinelemeli histogram değiştirme.

Bu tezde şifresiz ve şifreli videolar için tersinir video damgalama yöntemleri geliştirilmiştir. Her iki durumda da, video çerçeveleri arasındaki zamansal ilinti, hareket dengelemeli çerçeve aradeğerlemesi kullanılarak değerlendirilmiş ve damgalamada görüntüler için geliştirilmiş yinelemeli histogram değiştirime yönteminden faydalanılmıştır.

Her bir çerçeve için tersinirliğin garanti edilmesi ve toplam kapasitenin çerçevelere eşit bir biçimde dağıtılması problemlerinden dolayı yinelemeli histogram değiştirme yöntemi tersinir şifresiz video damgalamada doğrudan kullanılamaz. Bu tezde, bu iki problemin çözümünde özgün fikirler önerilmiştir. Önerilen yöntem, literatürde sıklıkla kullanılan video dizilerinde test edilmiş ve yöntemin mevcut tersinir şifresiz video damgalama yöntemlerinden kapasite ve bozunum performansı bakımından daha iyi sonuçlar verdiği bilgisayar benzetimleri ile gösterilmiştir.

Şifreli videolar için geliştirilen yöntemde, damgalama için çerçevede gerekli olan boşluk şifreleme öncesi oluşturulmaktadır. Boşluk oluşturma işlemi, orijinal çerçevedeki belirli piksellerin en düşük anlamlı bitlerinin ilgili çerçevenin hareket dengelenmiş aradeğerleme hatalarına yinelemeli histogram değiştirme yöntemi ile saklanması ile gerçekleştirilir. Damganın boşluk oluşturulmuş şifreli çerçevedeki piksellerin en düşük anlamlı bitleri ile yer değiştirilmesi ile damgalama işlemi tamamlanır. Orijinal video bir şifreleme anahtarı ile şifrelenir ve şifrelenmiş video bir damgalama anahtarı ile damgalanır. Damganın çıkartılması ve şifrenin çözülmesi işlemleri birbirinden bağımsız bir biçimde gerçekleştirilebilir. Her iki anahtarın bilinmesi durumunda damga ve orijinal video kayıpsız geri elde edilebilmektedir. Sadece damgalama anahtarının bilinmesi durumunda damgalanmış şifreli videodan damga çıkartılabilmekte fakat videonun içeriğine erişilememektedir. Sadece şifreleme anahtarı bilindiğinde ise damgaya erişilmeksizin orijinal videonun bozunumlu bir versiyonu elde edilebilmektedir. Bilgisayar benzetimleri önerilen yöntemin tersinir şifreli video damgalama uygulamalarındaki kullanılabilirliğini kanıtlamaktadır.

REVERSIBLE VIDEO WATERMARKING FOR PLAIN AND ENCRYPTED VIDEO BASED ON RECURSIVE HISTOGRAM MODIFICATION

SUMMARY

Keywords: Reversible watermarking, reversible video watermarking, encrypted domain reversible video watermarking, recursive histogram modification

In this thesis, reversible video watermarking methods are developed for plain and encrypted video. In both cases, temporal correlation among frames in the video are exploited by using motion compensated frame interpolation and recursive histogram modification algorithm developed for images are used in watermarking.

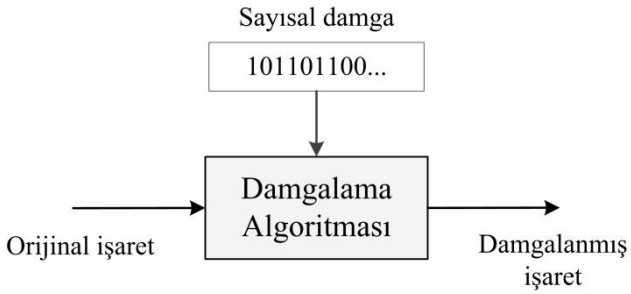
The recursive histogram modification can not be used directly for reversible plain video watermarking because of the important problems of ensuring reversibility for each frame and distribution of total capacity among frames. In this thesis, novel ideas are proposed to solve these two problems. The proposed method is tested on the video sequences commonly used in the literature. It is shown to give better performance than the existing reversible plain video watermarking algorithms in terms of capacity and distortion by means of computer simulations.

In the method developed for encrypted video, room required for the watermark is created before encryption. The room creating process is achieved by embedding the original least significant bits of some pixels of the frame into the corresponding motion compensated interpolation errors through recursive histogram modification. Watermarking is carried out by replacing the least significant bits of the pixels in the room reserved encrypted frame by the watermark bits. The original video is encrypted with an encryption key and the encrypted video is watermarked with a data hiding key. Watermark decoding and video decryption operations can be performed independently. The original video can be restored and the watermark can be decoded without error from the watermarked encrypted video if both keys are known. The watermark can be decoded from the the watermarked encrypted video but video content can not be accessed when only the data hiding key is known. A video signal close to the original one can be obtained without having access to the watermark when only the encryption key is available. Computer simulations prove applicability of the method in encrypted domain reversible video watermarking applications.

BÖLÜM 1. GİRİŞ

Sayısal damgalama, herhangi bir sayısal işarete işaretin anlamını bozmayacak şekilde bilgi ekleme olarak tanımlanabilir. İnternet ve sayısal çoklu ortam teknolojilerindeki gelişmeler ve bu teknolojilerin yaygınlaşması sayısal görüntü, video ve ses işaretlerinin dağıtım ve paylaşımını daha kolay ve hızlı bir hale getirmektedir. Yüksek çözünürlüklü sayısal veri kaydediciler (kamera, fotoğraf makinesi v.b.), geniş bantlı iletim kanalları, kablolu ve kablosuz ağlar ile veri saklama teknolojisindeki gelişmeler (DVD, CD, harici disk v.b.) sayısal görüntü, video ve ses işaretlerinin kullanım ve iletiminin yaygınlaşmasında güçlü bir altyapı hizmeti sunmaktadır. Bu gelişmelere paralel olarak, telefon, tablet ve kaydedici cihazlar başta olmak üzere sayısal görüntü ve video işleyen cihazların kullanımı yaygınlaşmaktadır. Ancak, tedbir alınmadığı takdirde sayısal veriler kayıpsız ve çok hızlı bir şekilde çoğaltılabilmektedir. Bu nedenle, medya içeriğinin güvenilirliğinin sağlanması ve medya üreticilerinin telif haklarının korunması önemli bir ihtiyaç olmuştur [1-6].

Sayısal dünya, üretilen medyanın kayıpsız ve çok hızlı bir şekilde kopyalanmasına ve dağıtılmasına olanak sağlamaktadır. Sayısal medya üreticilerinin haklarının korunması ve hassas veriler ile ilgilenilen askeri ve tıbbi gibi uygulamalarda medya içeriğinin güvenilirliğinin sağlanması amacıyla damgalama yöntemleri geliştirilmiştir. Farklı uygulama alanları için geliştirilmiş birçok sayısal damgalama yöntemi mevcuttur. Sayısal medya üreticilerinin haklarının korunması için geliştirilen *telif hakkı koruma*, yasal olmayan kopyaların kim tarafından yapıldığının tespit edilmesi için kullanılan *kimlik Tespiti*, sayısal medya kaydedicileri ile entegre çalışarak yasal olmayan kopyalamaların donanım kontrolü ile önüne geçmek için geliştirilen *kopyalama koruma*, ticari amaçlı reklamlara damga eklenerek otomatik bir denetleme sistemi ile reklamın sözleşmedeki gibi yayınlanıp yayınlanmadığının belirlenebildiği *yayın denetleme*, dayanıksız damgaların kullanıldığı *veri doğrulama*,



Şekil 1.1. Sayısal Damgalama

sayısal medyaya erişmede kullanılan *indeksleme*, tarih ve hasta bilgilerinin tıbbi görüntülere eklenmesiyle hatalı teşhislerin önüne geçilmesinde faydalı olan *tıbbi güvenlik*, gizli veya özel verilerin iletimini sağlayan *veri gizleme* sayısal damgalamanın başlıca uygulama alanları olarak listelenebilir [7-8].

Sayısal damgalama ile yakından ilişkili olan şifreleme, sayısal bilgilerin korunmasında tek başına yeterli değerlidir. Çünkü sayısal veriye uygulanan şifre bir kez çözüldükten sonra artık ilgili veri için herhangi bir koruma söz konusu değildir. Sayısal verinin içine işaretin anlamını bozmadan eklenen damga, işaret var olduğu sürece işaretle birlikte yaşayacak ve işaret anlamsızlaşmadığı sürece varlığını koruyarak yasadışı kopyalama, dağıtma ve sunma girişimlerine karşı işareti koruyacaktır [7-9].

Ayrıca kullanıcı, şifrelemeden sonra sayısal medyayı izlemek, dinlemek veya üzerinde değişiklik yapmak etmek isteyebilir, fakat bu işlemlerin her biri için şifre çözme işlemi gerekmektedir. Damgalanmış bir görüntü veya videoda ise medya içeriği herhangi bir ön işleme gerek duyulmaksızın erişilebilmekte iken aynı zamanda yetkili alıcılar tarafından damgaya erişim sağlanabilmektedir. Bu nedenle, sayısal damgalama şifrelemenin eksikliklerini gidermek amacıyla ortaya atılan tamamlayıcı bir teknoloji olarak düşünülebilir [1,2,7].

Damgalamanın blok diyagram gösterimi Şekil 1.1'de verilmiştir. Damganın formatı, uygulamaya göre değişiklik göstermekle birlikte, herhangi bir sayısal bilgi bir bit dizisiyle temsil edilebileceğinden literatürde gerçekleştirilen simülasyonlarda damga için rastgele oluşturulmuş bit dizileri kullanılmaktadır.

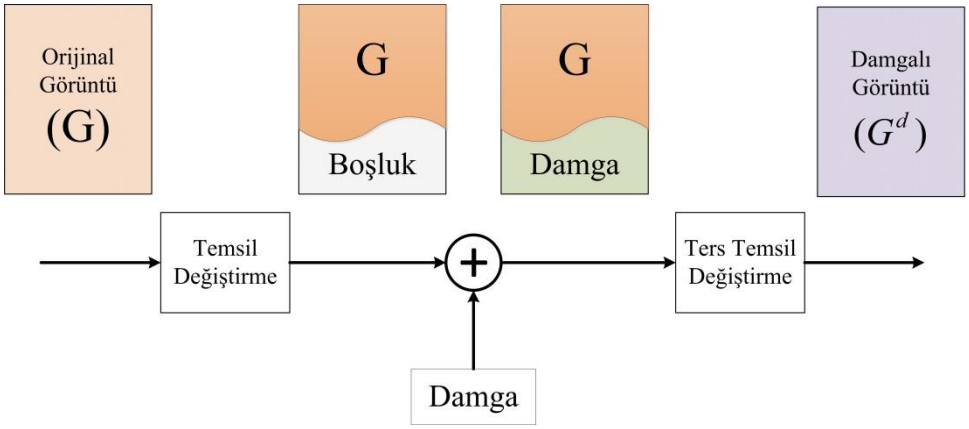
Herhangi bir sayısal damgalama işlemi sonucunda orijinal işarette değişiklikler gerçekleşir. Bunun sonucunda, alıcı tarafta orijinal işareti damgalı işaretten hatasız olarak geri elde etmek mümkün olmayabilir. Orijinal işaretteki değişiklikler veya bozunumlar önemli bazı uygulamalarda arzulanmaz. Örneğin, bir askeri uygulamada kullanılan bir harita üzerindeki bir piksel değerinin taşıdığı bilgi oldukça kritiktir. Benzer şekilde bir hastalığın teşhisi için kullanılan radyografik bir görüntüde meydana gelebilecek bir değişiklik yanlış teşhise yol açabilir. Hassas verilerin işlendiği askeri, tıbbi ve hukuki uygulamalarda kullanılma potansiyeline sahip, damga çıkartma işlemi sonucunda damgalı işaretten orijinal işaret ve damganın hatasız olarak geri elde edilebilmesine imkan veren damgalama yöntemleri Tersinir Damgalama (TD) olarak adlandırılır [10-11]. TD yöntemlerinde, önemli bilgiler içeren sayısal bir görüntü veya video kaydedilirken veya bir yerden başka bir yere iletilirken (askeri uygulamada operasyon, tıbbi uygulamada hastaya ilişkin veriler, hukuki uygulamada davaya ilişkin bilgiler) gizli bilgilerin orijinal işarette saklanması ile bir yandan veri güvenliği sağlanırken, diğer yandan gizli bilgi çözüldükten sonra orijinal işaret kayıpsız oluşturularak orijinal işaretin uygulamanın asıl amacına kesintisiz bir biçimde hizmet etmesi mümkün olmaktadır. Orijinal işaretteki bir bozunumun telafi edilemeyecek hatalara sebep olabileceği uygulamalarda TD yöntemleri orijinal verinin korunmasına olanak sağlamaktadır. TD yöntemlerinde, orijinal işaret, damgalanmış işaretten kayıpsız bir şekilde geri elde edilirken, uygulama amacına uygun miktarda damga bilgisinin orijinal işarete eklenmesi hedeflenir. Bir TD yönteminin kapasitesi veya eşdeğer olarak orijinal işarete ekleyebileceği maksimum bilgi miktarı damgalamanın nasıl yapıldığına bağlıdır.

TD, orijinal işaret ve eklenecek bilginin kritik olduğu uygulamalarda kullanılmaktadır. Özellikle askeri ve tıbbi uygulamalar, veri kaybına tahammülü olmayan yapılarıyla, TD yöntemlerinin kullanım alanlarının başında gelmektedir. Örneğin, herhangi bir hastalığın teşhisinde kullanılan bir tıbbi videonun içine hasta ve tedavi süreci ile ilgili bilgilerin eklenmesinde TD kullanılabilir. Aynı şekilde, hukuki bir vakanın aydınlatılmasında kanıt olarak kullanılacak bir videoya dava ile ilgili bilgiler TD kullanılarak eklenebilir. Bununla birlikte, bu tip uygulamalarda gizli bilginin sadece izin verilen alıcılar tarafından doğru bir biçimde alınması garanti

edilmek istendiğinde; TD yöntemi uygun bir şifreleme ve doğrulama algoritması ile bütünleştirilerek güvenlik seviyesi arttırılabilir [10-12].

Sayısal damgalama yöntemlerinde, yukarıda belirtilen uygulamalar için farklı gereksinimlere ihtiyaç vardır. Herhangi bir damgalama yöntemi için temel üç gereksinim; *algısal saydamlık*, saldırılara karşı *dayanıklılık* ve *kapasite* olarak belirtilebilir. Algısal saydamlık, orijinal işaret ile damgalı işaret arasındaki algısal benzerlik olarak tanımlanabilir. Algısal saydamlık, damgalama işlemi sonucunda orijinal işarette meydana gelecek bozunumun büyüklüğü ile ters orantılıdır. Dayanıklılık, damgalı işaret üzerinde kasıtlı veya kasıtsız değişiklik yapılması halinde damganın hangi doğrulukta tespit edilebileceğini belirtmek için kullanılır. Damgalı işaret bir noktadan başka bir noktaya iletilmesi esnasında, haberleşme kanalından kaynaklanan doğal bozulmalara uğrayabileceği gibi damga ve/veya orijinal işarete izinsiz erişmek amacıyla kasıtlı işaret işleme girişimlerine maruz kalabilir. Ayrıca, damgalı işaret kullanım amacına göre sıkıştırma, kırpma, döndürme gibi zorunlu değişimlere uğrayabilir. Damgalı işaretin değişime uğraması halinde damganın geri elde edilebilme doğruluğu damgalama yönteminin dayanıklılığını belirlemektedir. Kapasite, orijinal işarette saklanabilecek maksimum damga miktarıdır [13-14]. Bu üç temel gereksinime ek olarak yöntemlerin karmaşıklığını ve uygulanabilirliğini ölçmek için işlem yükü de damgalama yöntemlerinin performans değerlendirmesinde kullanılmaktadır.

Herhangi bir TD algoritması kullanılarak damgalanmış bir işarette oluşacak herhangi bir bozunum, damga çıkartmada hatalara neden olup orijinal işaret ve damganın kayıpsız geri elde edilmesine engel olacağından TD yöntemlerinde dayanıklılık bir performans ölçütü olarak kullanılmaz. Bu yüzden, TD yöntemlerinin performansı veri ekleme kapasitesi, orijinal işarette meydana gelen bozunum ve işlem yükü ölçütleri ile değerlendirilir.



Şekil 1.2. Tersinir görüntü damgalama algoritmalarının genel gösterimi.

TD'nın arkasındaki ana fikir, orijinal işareti temsil edildiği uzaydan farklı bir uzayda temsil ederek yeni temsilde boşluklar oluşturmak ve oluşan boşluğu damga ekleme için kullanmaktır. TD alanında yapılan çalışmaların büyük bir çoğunluğu görüntü üzerinde odaklanmıştır. Tersinir görüntü damgalama (TGD) yöntemleri, genellikle komşu pikseller arasındaki uzamsal ilintinin değerlendirilerek farklı bir uzayda görüntü üzerinde boşluk oluşturulması ve bu boşluğa damganın eklenmesi esasına dayanmaktadır. Görüntü temsil değiştirme yöntemiyle farklı bir uzayda temsil edilmekte ve yeni temsilde oluşturulan boşluğa damga yerleştirilmektedir. Şekil 1.2.'de TGD algoritmalarının genel gösterimi verilmiştir. TGD yöntemlerinde uzamsal ilinti, pikseller arası fark, öngörü veya aradeğerleme yöntemleri aracılığıyla görüntüde birbirinden ilintisiz iki bileşen oluşturulmakta ve elde edilen öngörü veya aradeğerleme farkları bileşenine veri saklama yöntemleri uygulanmaktadır.

Video damgalama konusundaki araştırmaların büyük bir çoğunluğu görüntü için geliştirilen yöntemlerden esinlenmiştir. Bir video damgalama yöntemi, herhangi bir görüntü damgalama yönteminin video çerçevelerinin her birine veya birkaçına bağımsız bir şekilde uygulanmasıyla geliştirilebilir. Ancak, böyle bir çözümde video çerçeveleri arasındaki zamansal ilinti yararlanılmamış olur. Bir video işareti arka arkaya gelen görüntü çerçevelerinden oluşmasına rağmen, görüntüler için geliştirilmiş TD algoritmalarının her bir video çerçevesine ayrı ayrı uygulanması ile etkin bir tersinir video damgalama (TVD) gerçekleştirilemez. Ayrıca, video işaretlerinin damgalanması esnasında bozunum, kapasite ve işlem yükü gibi

ölçütlerin en iyilenmesi görüntüdekinden oldukça farklıdır. İstenilen ölçütleri sağlayacak bir damgalama stratejisinin geliştirilmesi çözülmesi gereken bir problemdir. Yapılan değerlendirmeler ışığında görüntü için geliştirilen TD yöntemlerinin video işaretlerine doğrudan uygulanamayacağı anlaşılmaktadır. Video çerçeveleri arasındaki zamansal ilinti, hareket kestirimi yöntemleri yardımıyla kullanılarak etkin video damgalama yöntemleri geliştirilebilir [1-3]. Literatürde, orijinal işarette en az bozunum oluşacak şekilde en büyük kapasiteye ulaşmayı hedefleyen sınırlı sayıda TVD çalışması vardır. Diğer bir deyişle yeterli performansa sahip TVD yöntemlerinin geliştirilmesi amacıyla yeni araştırmaların yapılması gereklidir.

Bulut teknolojisinin, istemci-sunucu mimarisinde çalışan büyük veri merkezlerinin ve içerik sağlayıcılar tarafından üretilen medyanın son kullanıcılara üçüncü parti servis sağlayıcılar tarafından iletildiği özel uygulamaların yaygınlaşması sonucunda şifreli verilerin damgalanması ihtiyacı ortaya çıkmıştır. Herhangi bir içerik sağlayıcısı tarafından üretilen medyanın son kullanıcıya uzak bir sunucu aracılığıyla iletildiği bir uygulamada içerik sahibi, sunucu tarafında medyanın içeriğine erişilmesini istemeyebilir. Bununla birlikte, sunucu ilgili dosyaya içerik sağlayıcı ve son kullanıcı hakkında bilgiler, tarih/saat ve sayaç gibi etiketler veya doğrulama bilgisi eklemek zorunda olabilir. Örneğin, tıbbi bir uygulamada bir hasta ile ilgili görüntüye üçüncü şahısların erişmesi istenmez. Fakat veritabanı sunucusu bu görüntüye hasta, doktor veya hastane ile ilgili kayıtlar eklemek isteyebilir. Aynı zamanda, teşhis ve tedavinin doğruluğu için orijinal görüntü ve eklenen bilgilerin kayıpsız geri elde edilmesi gerekir. Böyle bir senaryoda, içerik sahibi iletmek istediği veriyi şifreleyip sunucuya gönderdiğinde, sunucu şifreli dosyaya gerekli bilgileri tersinir şifreli damgalama (TŞD) algoritması ile ekler. TŞD yöntemleri, hassas verilerin kullanıldığı uygulamalarda medya içeriğinin korunmasına ve orijinal verinin kayıpsız geri elde edilmesine imkân vermektedir. TŞD yöntemleri ilk olarak görüntüler için geliştirilmiştir [15-20]. TVD kapsamındaki tüm çalışmalarda videonun şifresiz olduğu varsayılmıştır. Bu nedenle şifreli videolar için TVD yöntemlerinin geliştirilmesi yeni bir araştırma alanıdır.

Bu tezde, şifresiz ve şifreli videolar için TD yöntemleri oluşturmak amacıyla araştırmalar yapılmıştır. Görüntü için geliştirilmiş yinelemeli histogram değiştirme (YHD) yöntemi temel alınarak şifreli ve şifresiz videolar için özgün iki TVD algoritması geliştirilmiştir. Tez aşağıdaki şekilde düzenlenmiştir.

Bölüm 2., tezde kullanılan matematiksel altyapının tanıtılmasına ayrılmıştır. Ayrıca, literatürdeki TGD ve TVD yöntemleri arasında tezle bağlantılı olanlar özetlenmiştir.

Bölüm 3.'de şifresiz videolar için bir TVD algoritması geliştirilmiştir. Sayısal görüntüler için geliştirilmiş YHD yönteminin video işaretlerine uyarlanmasında tersinirliğin sağlanması ve damgalı videodaki bozunum en az olacak şekilde toplam kapasitenin çerçevelere dağıtılması problemleriyle karşılaşılmaktadır. Karşılaşılan bu iki problemi çözmek amacıyla özgün fikirler önerilmiştir. Önerilen yöntemde, hareket dengelenmiş çerçeve aradeğerleme (HDÇA) hatalarına YHD uygulanmaktadır. Tersinirliği sağlamak adına video dizisinde önce çift, daha sonra tek numaralı çerçeveler damgalanmıştır. Tek numaralı orjinal çerçeveler kullanılarak çift numaralı çerçeveler için HDÇA hataları oluşturulmuş ve hatalar YHD kullanılarak damgalanmıştır. Daha sonra, damgalanmış çift numaralı çerçeveler kullanılarak tek numaralı çerçevelerin damgalanması benzer biçimde gerçekleştirilmiştir. Damga çözme kısmında, önce tek numaralı orijinal çerçeveler, daha sonra ise orijinal çift numaralı çerçeveler geri elde edilmektedir. Çerçevelerin bu sırada işlenmesi, damgalama algoritmasının tersinirliğini sağlamıştır. Ayrıca, hareket vektörlerinin damga çözücüye iletilmesi ihtiyacı ortadan kaldırılarak damga çözmek için gerekli yan bilgi miktarı azaltılmıştır. Toplam kapasitenin video çerçevelerine nasıl dağıtılacağı problemi, kapasite ve bozunum tüm çerçevelere eşit bir biçimde dağıtılarak çözülmüştür. Toplam kapasite, çerçevelere eşit bir biçimde dağıtıldıktan sonra, her bir çerçevenin damgalanmasında YHD yönteminin bozunum kısıtı yinelemeli bir biçimde değiştirilerek hedef kapasiteye minimum bozunumla erişilmesi sağlanmıştır. Bir video dizisi verilen bir kapasitede damgalanmak istendiğinde önerilen yöntem kapasiteyi video dizisindeki çerçevelere eşit bir biçimde paylaştırır. Her bir çerçeve için belirlenen hedef kapasite YHD yöntemi ile damgalanarak bozunum tüm videoya paylaştırılır. Önerilen yöntem test video dizileri

üzerinde uygulanmış ve mevcut TVD yöntemlerinden kapasite ve görsel kalite bakımlarından üstün olduğu gösterilmiştir.

Şifreli videolar için geliştirilen TVD algoritması Bölüm 4.'te verilmiştir. Bildiğimiz kadarıyla, bu konuda bir çalışma henüz mevcut değildir. Önerilen yöntemde, orijinal video bir şifreleme anahtarı ile şifrelenmekte ve veri ekleme işlemi damgalama anahtarı ile gerçekleştirilmektedir. Şifreli damgalı videodan hem damga hem de orijinal videonun kayıpsız geri elde edilmesi sadece her iki anahtar bilindiğinde mümkündür. Sadece damgalama anahtarı mevcutken, video içeriğine erişilmeksizin, damga çıkartılabilmektedir. Benzer şekilde, sadece şifreleme anahtarı biliniyorsa orijinal görüntü çok az bir bozunumla oluşturulabilmekte fakat damga çıkartılamamaktadır.

Bölüm 5.'te tezin şifreli ve şifresiz video dizilerinin tersinir damgalanmasında yaptığı katkılar özetlenmiştir. Ayrıca ileride yapılabilecek araştırmalar ve bunların TVD çalışmalarına potansiyel katkıları hakkında fikirler verilmiştir.

BÖLÜM 2. LİTERATÜR ÖZETİ

Sayısal damgalama, ses, görüntü veya video işaretlerine uygulanabilir. Ses işareti, görüntü veya video işaretlerinden farklı nitelikte olduğundan ses damgalama yöntemleri, görüntü ve video damgalama yöntemlerinden tamamen farklıdır. Görüntü ve video işaretlerinin damgalanmasında kullanılan teknikler ise işaretlerin birbiriyle ilişkisinden dolayı benzer prensiplere dayanmaktadır. Bunun temel nedeni sayısal videonun aslında sayısal görüntülerden oluşan bir dizi olmasıdır. Video damgalama konusunda yapılan araştırmalar doğal olarak görüntü için geliştirilmiş yöntemlerden esinlenmiştir. Bununla birlikte, ilk TD araştırmaları görüntüler için yapılmıştır.

Bu tezde, şifresiz ve şifreli videolar için TD yöntemleri geliştirmek hedeflenmiştir. Bu amaçla, özgün iki TD yöntemi geliştirilmiştir. Geliştirilen yöntemler, görüntüler için geliştirilmiş TD algoritmalarından faydalanmıştır. Literatürdeki şifresiz görüntüler için geliştirilmiş mevcut TD yöntemleri Bölüm 2.1.'de, şifreli görüntüler için geliştirilmiş TD algoritmaları ise Bölüm 2.2.'de özetlenmiştir. Bölüm 2.3.'te ise video işaretleri için geliştirilmiş TD yöntemleri tartışılmıştır. Bu tezde geliştirilen yöntemlere altyapı oluşturan Yinelemeli Histogram Değiştirme algoritmasının detayları Bölüm 2.4.'te anlatılmıştır. Yöntemin daha iyi anlaşılabilmesi için Bölüm 2.5.'te sayısal bir örnek verilmiştir.

2.1. Şifresiz Görüntüler İçin Geliştirilmiş Tersinir Damgalama Algoritmaları

Tersinir görüntü damgalama (TGD) alanında yapılan çalışmalar, başlangıçta pikseller arasındaki uzamsal ilintiyi değerlendirerek eklenecek veri için boşluk oluşturma esasına dayanmaktaydı. [12]'deki çalışmada kayıpsız sıkıştırma ve en düşük anlamlı bit (LSB) değiştirme tabanlı bir TGD yöntemi sunulmuştur. Yöntem, görüntüdeki her

bir pikselin en az anlamlı bitlerini (LSB) damgalama için boşaltıp bu bitlerin damga bitleri ile değiştirilmesi esasına dayanmaktadır.

[21]'deki çalışmada, Fark Genişletme (FG) olarak bilinen, bir görüntüdeki komşu iki piksel arasındaki ilintiden yararlanarak tersinir damgalama işlemini gerçekleştiren TGD yöntemi öne sürülmüştür. Yöntemde, damga komşu iki piksel arasındaki farkın LSB'sinde saklanmaktadır. Tamsayı dalgacık dönüşümü olarak da bilinen FG, bir görüntüdeki komşu iki piksel arasındaki ilintiden yararlanarak tersinir damgalama gerçekleştirir. FG yönteminde, tamsayı ile ifade edilen komşu iki piksel değeri birbirinden ilintisiz olan başka iki tamsayı değerine dönüştürülür. Bu ilintisiz iki tamsayıdan birisi piksellerin farkı diğeri piksellerin ortalamasıdır. Komşu piksel farkına bit eklenerek damgalama işlemi gerçekleştirilir. Başka bir ifade ile FG; görüntüdeki komşu iki piksele tamsayı dalgacık dönüşümü uygular (iki pikselin ortalaması ve farkı hesaplanır). Algoritmaya göre damga eklenebilme koşulları sağlanıyorsa, damga dizisinden bir bit alınarak elde ettiğimiz fark değerine eklenir. FG, yüksek kapasite sağlaması ve düşük bozunuma neden olması sebebiyle sonra gelen araştırmalara öncülük etmiştir.

FG, bütünlük açısından aşağıda özetlenmiştir. (x,y) komşu piksel çiftinin değerleri olmak üzere ortalama değer *l* ve fark *h*

$$l = \left\lfloor \frac{x+y}{2} \right\rfloor, \quad h = x - y \tag{2.1}$$

eşitliklerinden hesaplanır. Sonuç olarak, *(x,y)* çifti artık *(l,h)* çiftine dönüşür. *(l,h)* çiftinden orijinal *(x,y)* çiftinin geri elde edilebilmesine imkan veren ters dönüşüm

$$x = l + \left\lfloor \frac{h+1}{2} \right\rfloor, \quad y = l - \left\lfloor \frac{h}{2} \right\rfloor \tag{2.2}$$

eşitlikleriyle verilir. *b* damgalanacak bit olmak üzere, *h* değerinin

$$h' = 2 \times h + b \tag{2.3}$$

eşitliğine göre genişletilmesiyle damgalama gerçekleştirilir. Denklem (2.3), aslında fark değerinin 2'lik sayma sistemindeki temsilinde en az anlamlı bitinin sonuna damga bitini eklemek anlamına gelmektedir. Damgalanmış fark değeri h' Denklem (2.2)'de verilen ters dönüşüm formüllerinde yerine konarak damgalanmış (x', y') piksel çifti elde edilir.

Algoritmanın anlaşılmasını kolaylaştırmak için sayısal bir örnek verilecektir. Piksel çifti $(x, y) = (206{,}201)$ ve damgalanacak bit $b = 1$ olsun. Denklem (2.1)'de verilen dönüşüm formülleri yardımıyla ortalama ve fark değerleri,

$$l = \left\lfloor \frac{206 + 201}{2} \right\rfloor = 203, \quad h = 206 - 201 = 5$$

olarak elde edilir. Denklem (2.3)'den damgalanmış fark değeri

$$h' = 2 \times 5 + 1 = 11$$

şeklinde hesaplanır. Denklem (2.2)'de verilen ters dönüşüm formüllerinde yeni fark değeri h' ve orijinal ortlama değer *l* yerine konarak damgalanmış piksel çifti (x', y') aşağıdaki şekilde oluşturulur.

$$x' = 203 + \left\lfloor \frac{11 + 1}{2} \right\rfloor = 209, \quad y' = 203 - \left\lfloor \frac{11}{2} \right\rfloor = 198$$

Damgalanmış piksel çifti (x', y')'den damga biti *b* ve orijinal piksel çifti (x, y) geri elde edilebilir. İlk önce, damgalı pikseller için ortalama ve fark çifti Denklem (2.1)'den hesaplanır.

$$l' = \left\lfloor \frac{209 + 198}{2} \right\rfloor = 203, \quad h' = 209 - 198 = 11$$

Görüldüğü üzere, damgalanmış ve orijinal piksel çiftlerinden hesaplanan ortalama değer aynı kalmakta fark değeri ise değişmektedir. Damgalanmış farkın ikili sistem gösterimi $h' = (1011)_2$ olup gösterimdeki en az anlamlı bit damgalanmış bittir. Geri

kalan bitlerin onluk sayma sisteminde karşılığı orijinal h değerini verir. $LSB(h')$, h' dizisinin en az anlamlı biti ve $\lfloor h' \rfloor$, h''den küçük ve en büyük tamsayıyı belirtmek üzere, bu işlem matematiksel olarak

$$b = LSB(h') = 1, \quad h = \left\lfloor \frac{h'}{2} \right\rfloor = 5$$

ile ifade edilebilir. LSB işlevi en düşük anlamlı biti elde etme işlemini temsil etmektedir. Orijinal fark değeri bulunduktan sonra orijinal piksel çifti ters dönüşüm uygulanarak hesaplanır.

Gri seviyeli görüntülerdeki piksel değerleri genelde 8 bit ile temsil edildiğinden, bir pikselin değeri 0 ile 255 arasında bir tamsayıdır. Orijinal fark değerine bit eklendiğinden damgalı fark değeri 255'den büyük bir tamsayı olabilir. Bu durumda, piksel farkları, bit ekleme yerine bit değişikliği işlemine tabi tutulur. Bit değişikliği, fark değerinin en düşük anlamlı bitinin damga bitiyle değiştirilmesidir. Damga çözme aşamasında, fark değerine eklenen bit tespit edilebilir. Fakat bir farkın damgalanıp damgalanmadığı belirlenemez. Damgalı farkları diğerlerinden ayırt etmek amacıyla, 2 boyutlu ve 1 bitlik bir konum haritası oluşturulur. Konum haritasında, damgalı farklara ilişkin koordinatlarda 1 diğer koordinatlarda 0 değeri saklanır. Damga çözücü, konum haritası aracılığıyla damgalı farkları orijinal farklardan ayırabilir ve orijinal görüntüyü geri elde edebilir.

Şekil 2.1.'de 512 x 512 boyutlarındaki orijinal lena görüntüsü, bu görüntüye FG yöntemi ile yaklaşık 101.000 bit eklendiğinde elde edilen damgalanmış görüntü ve bu iki görüntü arasındaki hatalar gösterilmiştir. Hatırısayılır büyüklükte damgalama kapasitesinde FG yöntemi ile elde edilen görsel kalitenin ve orijinal görüntüde ortaya çıkan hataların tatmin edici seviyelerde olduğu Şekil 2.1'den gözlemlenebilmektedir.

FG yöntemi araştırmacılar tarafından bazı bakımlardan iyileştirilmiştir. [22]'deki çalışmada komşu piksellerden fark değeri yerine fark vektörü hesaplanmış, fark vektörüne birden fazla bit eklenerek FG yönteminin kapasitesi arttırılmıştır. [23]'te, konum haritasının boyutunu azaltmak için bir yaklaşım sunulmuştur. Daha sonraki

(a) (b) (c)

Şekil 2.1. (a) 512×512 boyutlarındaki gri seviyeli orijinal Lena görüntüsü, (b) Damgalanmış Lena görüntüsü, (c) Orijinal ve damgalanmış görüntü arasındaki fark.

bir çalışmada, konum haritasına gerek duymayan bir FG yöntemi geliştirilmiştir [24]. [25]'te fark değerlerinin yerine öngörü hatalarının damgalanmasının kapasiteyi önceki yaklaşımlara göre daha da arttıracağı gösterilmiştir. Yakın geçmişte, ara değerleme hatalarının damgalanmasına dayalı yeni bir TGD yöntemi geliştirilmiş, yöntemin yüksek kapasite sağlamasının yanında orijinal görüntüde az bozunuma neden olduğu gösterilmiştir [26].

Histogram Değiştirme (HD) olarak adlandırılan ve önceki yaklaşımlardan tamamıyla farklı bir TGD yöntemi [27]'de tanıtılmıştır. Yöntem, görüntünün histogramındaki maksimum ve minimum noktalardan faydalanarak az hesap yüklü, yüksek kapasiteli ve düşük bozunumlu bir TD sağlamaktadır. Yöntemde, orijinal görüntünün histogramındaki tepe noktasına ait piksel değeri damgaya yer açmak amacıyla boşaltılır. Görüntü histogramındaki maksimum nokta sayısı kadar boşluk oluşturulabileceğinden yöntemin kapasitesi histogramın tepe noktası tarafından belirlenir. Oluşturulan boşluklara, damga dizisinden sırayla bit eklenerek damgalanmış görüntü oluşturulur. Yöntemin anlaşılması için Lena görüntüsüne ait histogram üzerinden özet bir tartışma aşağıda yapılmıştır.

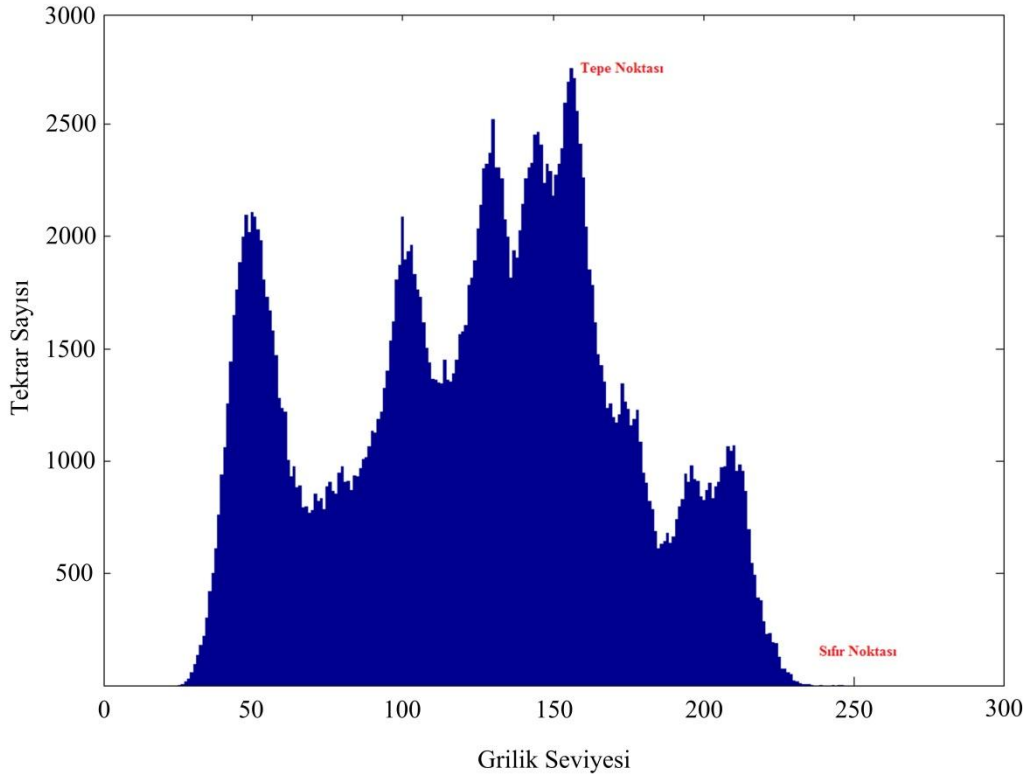
Şekil 2.1.'deki orijinal lena görüntüsünün, Şekil 2.2.'de verilen histogramını gözönüne alalım. Amaçlanan, görüntüde en fazla tekrarlanmış piksel değerini (tepe noktasına) damga bitine göre değiştirerek görüntüye tersinir olacak şekilde bilgi eklemektir. Öncelikle, görüntünün histogramı tepe noktasından sıfır noktasına doğru

1 birim ötelenerek Şekil 2.3.'te verilen ötelenmiş histogram oluşturulur. Bu adım sonunda görüntünün tepe noktası damga için boşaltılmış olur. Sonra, tepe noktasına ait piksel değerleri damga biti (0 veya 1) ile toplanarak damgalanmış görüntü elde edilir. Bu şekilde tepe noktası sayısı kadar bit görüntüye eklenebilir. Şekil 2.4.'te damgalanmış görüntünün histogramı gösterilmiştir. Orijinal görüntünün tekrar geri elde edilebilmesi için orijinal görüntüye ilişkin histogramın maksimum ve minimum noktaları alıcı tarafından bilinmelidir. Öteleme sonrası, damgalanmış görüntüde sadece tepe noktasına ait piksel değerleri, damga bitine göre değiştirilmiş olmaktadır. Değişim miktarı ise maksimum 1 birim olarak gerçekleşmektedir.

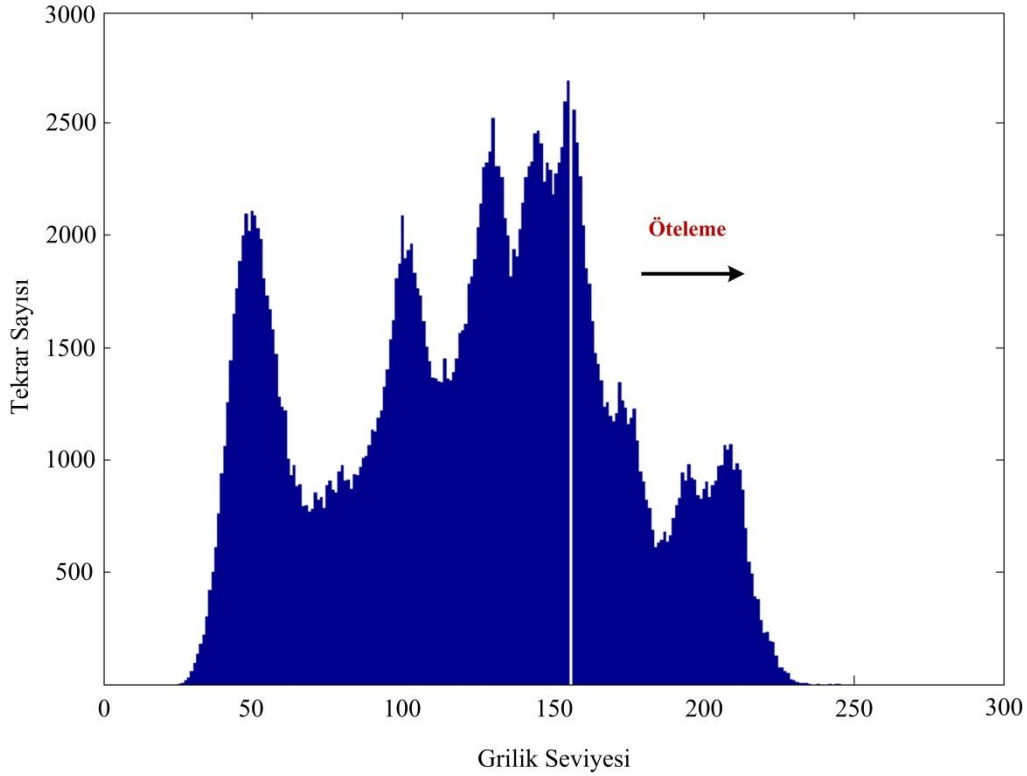
HD yöntemiyle damgalanmış görüntüde oluşabilecek bozunumun 48 dB'in altına düşmeyeceği teorik olarak garanti edilmiştir. Orijinal görüntüde, histogramdaki maksimum ve minimum noktalar arasındaki değerlere ait piksellerin grilik seviyesi damga ekleme sonrası 1 birim arttırılıp veya azaltılmaktadır. En kötü durumda orijinal görüntüdeki tüm piksellerin grilik seviyesi 1 birim değişecektir. Bu durumda ortalama karesel hata maksimum bire eşit olacaktır, $\mathrm{MSE} = 1$. Sonuç olarak, damgalanmış görüntü ile orijinal görüntü arasındaki minimum tepe işaret gürültü oranı (PSNR)

$$\mathrm{PSNR} = 10 \times \log_{10}\left(\frac{255 \times 255}{\mathrm{MSE}}\right) = 48{,}13 \text{ dB}$$

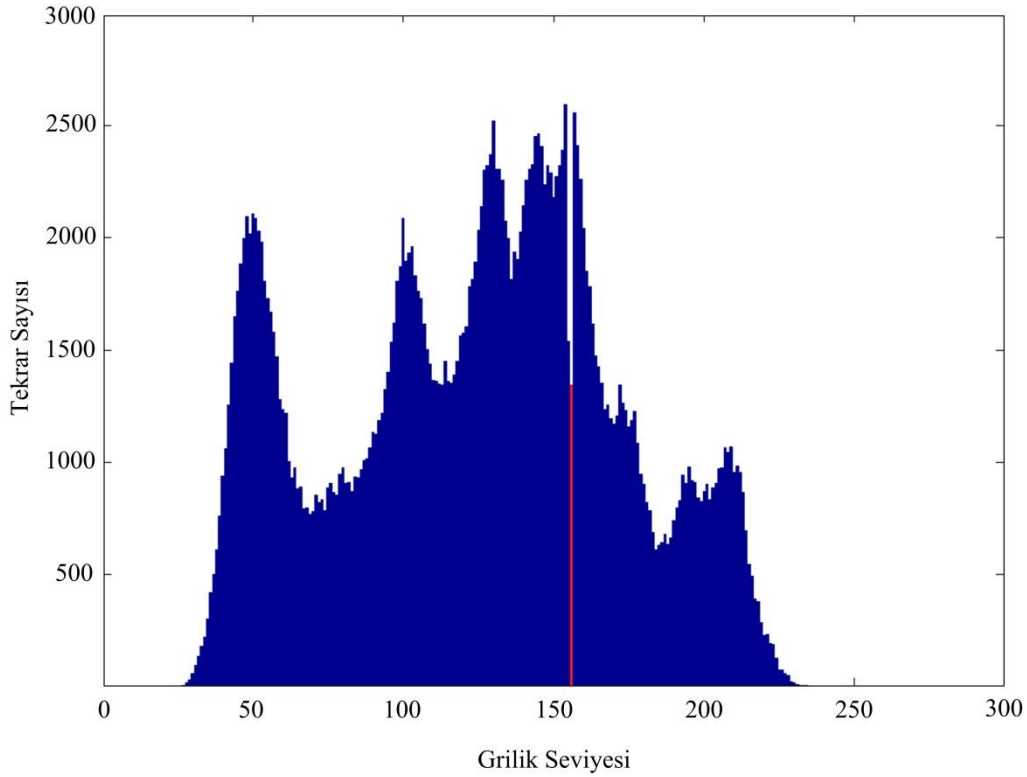
olacaktır. HD yöntemiyle damgalanan görüntü ile orijinal görüntü arasındaki bozunumun alt sınırının 48,13 dB olduğu sadece teorik olarak ispatlanmasının yanında farklı görüntüler üzerinde gerçekleştirilen deneylerle pratik olarakta desteklenmiştir. HD yöntemiyle elde edilen bu alt sınır, literatürdeki çoğu TD yöntemlerinde elde edilen bozunum değerlerinden daha yüksektir.



Şekil 2.2. Orijinal Lena görüntüsünün histogramı.



Şekil 2.3. Ötelenmiş Lena histogramı.



Şekil 2.4. Damgalanmış Lena histogramı.

HD yönteminin eksikliklerini gidermek amacıyla yapılmış çok sayıda araştırma arasında dikkat çekici olanlar geliştirilme sıralarına göre aşağıda özetlenmiştir. [28]'deki çalışmada histogramdaki maksimum ve minumum noktaların yerine konum haritasının kullanılması önerilmiştir. Başka bir çalışmada, üç piksel blok farklarının histogramına HD uygulanmıştır [29]. Alt örneklenmiş görüntü farklarının histogramına veri eklemenin performansta iyileşmeye neden olduğu gösterilmiştir [30]. Son olarak, HD ve FG yöntemlerinin üstünlüklerini birleştirerek öngörü hatalarının genişletilmesine (ÖHG) dayalı bir TGD yöntemi geliştirmiştir [11].

Son yıllarda TGD alanında oldukça özgün yöntemler geliştirilmiştir. [31]'deki çalışmada, görüntünün dağılımına göre yinelemeli bir algoritmayla oluşturulan transfer matrisi yardımıyla komşu pikseller arasındaki öngörü hataları damgalanmıştır. [32], görüntü üzerinde referans pikseller belirleyerek görüntü tamamlama yöntemi yardımıyla tahmin görüntüsü elde etmekte ve damgalamada HD yönteminden yararlanmaktadır. [33]'te öne sürülen yöntemde, değiştirme

doğrultusundan yararlanma algoritmasına tersinirlik eklenerek yeni bir TGD algoritması sunulmuştur.

Çok yakın bir geçmişte, Zhang ve arkadaşları orijinal işaretin histogramını literatürdeki diğer yöntemlerden farklı ve özgün bir biçimde değiştirerek damgalama işlemini gerçekleştiren bir TGD algoritması geliştirmiştir [34]. Yinelemeli Histogram Değiştirme (YHD) olarak adlandırılan bu yöntemde, verilen bir bozunum kısıtı için damgalanmış işaretin olasılık dağılımı hesaplanıp elde edilen dağılıma göre kayıpsız veri sıkıştırma algoritmaları kullanılarak damga (bit dizisi) ters sıkıştırılmakta ve orijinal işaret ters sıkıştırılmış sembol dizisi ile değiştirilmektedir. Yöntem görüntülerin öngörü ve aradeğerleme hatalarına uygulanmış ve görsel kalite bakımından mevcut yöntemlere göre daha iyi sonuçlar elde edilmiştir.

2.2. Şifreli Görüntüler İçin Geliştirilmiş Tersinir Damgalama Algoritmaları

TD yöntemleri başlangıçta şifresiz görüntü veya video işaretleri üzerinde geliştirilmişlerdir. Son yıllarda bulut tabanlı internet ve akıllı telefon uygulamalarının yaygınlaşması ile şifreli verilerin damgalanması ihtiyacı ortaya çıkmıştır. Şifreli damgalamada, damgalama ve şifreleme işlemlerinin birbirlerine bağımlılığı ve öncelik sıraları uygulama amacına göre farklılık göstermektedir. Damga çıkartma ve şifre çözme işlemlerinin birbirlerinden bağımsız bir şekilde yapılıp yapılmamasına göre tersinir şifreli damgalama (TŞD) yöntemleri Ayrıştırılamaz ve Ayrışabilir olmak üzere iki grupta sınıflandırılabilir. Damga çıkartma işleminin şifre çözme işleminden bağımsız olarak yapılamadığı yöntemler Ayrıştırılamaz, damga çıkartma işleminin şifre çözme işlemine ihtiyaç duymadan yapılabildiği yöntemler ise Ayrışabilir olarak adlandırılmıştır [17]. Ayrıştırılamaz yöntemlerde şifre çözme ve damga çıkartma işlemlerinin yapılabilmesi için damgalama ve şifreleme anahtarları bilinmelidir. Ayrışabilir yöntemlerde sadece damgalama anahtarının bilinmesi damga çözme için, sadece şifreleme anahtarının bilinmesi ise şifre çözme için yeterlidir.

TŞD algoritmaları, boşluk oluşturma işleminin şifrelemeden önce veya sonra yapılmasına göre şifreleme öncesi boşluk oluşturma (ŞÖBO) ve şifreleme sonrası

boşluk oluşturma (ŞSBO) tabanlı algoritmalar olmak üzere iki sınıfa ayrılabilir [20]. ŞSBO yöntemlerinde, orijinal işaret, doğrudan şifrelenir ve üçüncü parti veri damgalayıcı şifreli işareti değiştirerek damgayı ekler. ŞÖBO tabanlı yöntemlerde boşluk, şifreleme işlemi sonrasında da varlığını koruyacak şekilde şifreleme öncesinde oluşturulmaktadır.

Tersinir şifreli görüntü damgalama (TŞGD) konusunda çeşitli yöntemler önerilmiştir. [15], görüntüde her bir bloğu bir damgalama anahtarı yardımıyla iki ayrı kümeye ayırarak damga bitinin ‘1’ veya ‘0’ durumları için bir kümedeki piksellerin en az anlamlı üç bitini tersleyerek damgalama işlemini gerçekleştirmiştir. Damga çıkartma ve görüntü geri çatma amacıyla, öncelikle şifreleme anahtarı ile şifre çözülerek orijinal görüntünün bozunumlu bir hali elde edilir. Elde edilen yaklaşık görüntüden damgalama anahtarı yardımıyla damgalama adımındaki blok ve kümeler elde edilip her iki kümenin en az anlamlı 3 biti ayrı ayrı terslenerek iki blok elde edilir. İki bloğun pikselleri arasındaki dalgalanmadan orijinal blok belirlenir ve damga biti elde edilir. Blok boyutu doğru seçilmediğinde damga çıkartmada ve görüntü geriçatımında hatalar oluşabilmektedir. [16], bloklar arasındaki uzamsal ilintiden yararlanarak [15]’te tanıtılan yöntemin hata oranını düşürüp damgalama kapasitesini yükseltmiştir. İki çalışma da ŞSBO tabanlı ayrıştırılamaz yöntem sınıfındadır. Diğer bir çalışmada, şifreli görüntünün LSB’lerinin sıkıştırılmasında oluşturulan boşluğa damga ekleyen ŞSBO tabanlı ayrışabilir bir TŞGD yöntemi geliştirilmiştir [17].

ŞÖBO tabanlı, ayrışabilir bir TŞGD yöntemi [20]’de sunulmuştur. Yöntemde, şifreleme işlemi öncesinde oluşturulan boşluk, görüntü şifrelendikten sonra da varlığını korumaktadır. Yöntem, boşluk oluşturma, görüntü bölütleme, kendi kendini tersinir damgalama (KKTD) ve görüntü şifreleme adımlarından oluşmaktadır. Görüntü bölütleme adımında, orijinal görüntü A ve B olarak adlandırılan iki bölgeye ayrılmaktadır. A bölgesinin orijinal LSB’leri TGD yöntemlerinden herhangi biri ile B bölgesinde saklanarak A bölgesinde boşluk oluşturulur. Orijinal A ve değiştirilmiş B bölgesinden oluşan görüntü şifrelenir. Son adımda, damga A bölgesinin LSB’lerinde saklanarak damgalanmış şifreli görüntü elde edilir. A bölgesinin orijinal LSB’leri B bölgesinde saklı olduğu için damga çözme ve görüntü geriçatımı esnasında orijinal

görüntü hatasız geri çatılabilir. A bölgesinin boyutu damgalanabilecek piksel sayısını (kapasiteyi) belirlemektedir. Kapasite, şifreli görüntünün ilk 10 pikselinin LSB'lerinde saklanır. Bu yüzden, damgalama işlemine bu 10 pikselden sonra başlanır. [18]'de, yüksek anlamlı bit kestirimi ve kaynak kodlama tekniklerini birleştirerek yüksek kapasiteli ayrışabilir, ŞSBO tabanlı bir TŞGD gerçekleştirilmiştir. [19]'da ise, veri saklama anahtarına ihtiyaç duymayan ve damga çıkartmada şifreli ve şifreli olmayan blokların istatistiksel ayırt edilebilirliklerinden faydalanan ayrıştırılamaz, ŞSBO tabanlı bir algoritma geliştirilmiştir.

2.3. Video İşaretleri İçin Geliştirilmiş Tersinir Damgalama Algoritmaları

Videoda arka arkaya gelen çerçeveler arasında ilinti olduğundan, görüntü için geliştirilmiş damgalama yöntemleri video işaretlerine doğrudan uygulandığında var olan ilintiden yararlanılmaz. Video sıkıştırmada olduğu gibi, çerçeveler arası ilinti giderildikten sonra bir tersinir damgalama yöntemi videoya uygulandığında hem görsel kalite hem kapasite değerleri daha iyi sonuç verecektir. Görüntü için geliştirilen damgalama yöntemlerinin, çerçeveler arası zamansal ilinti giderildikten sonra, video işaretlerine uyarlanmalarıyla etkili video damgalama yöntemleri geliştirilebilir. Çerçeveler arası ilinti Hareket Dengelenmiş Aradeğerleme ve Hareket Dengelenmiş Öngörü hataları ile giderilebilir.

İlk TVD algoritmaları, çerçeveler arası hareket dengelenmiş öngörü hatalarının HD aracılığıyla değiştirilmesi esasına dayalıdır [35-36]. [37] ve [38] çalışmalarında, video çerçeveleri arasındaki ilintiyi hareket dengelenmiş aradeğerleme hatalarını kullanarak gideren TVD yöntemleri geliştirilmiştir. Yöntemlerde, aradeğerleme hatalarını damgalamak amacıyla görüntü için geliştirilen FG yöntemi videoya uyarlanmıştır. Bu tezde, YHD algoritmasından yararlanılarak şifreli ve şifresiz videolar için yeni iki TVD yöntemi geliştirilmiştir. Önerilen TVD algoritmalarının daha iyi anlaşılabilmesi için YHD yönteminin detayları aşağıda verilmiştir.

2.4. Yinelemeli Histogram Değiştirme Algoritması

Herhangi bir TD yöntemi geliştirilirken, verilen bir orjinal işaret için belli bir bozunum kısıtı altında erişilebilecek en büyük damgalama kapasitesi ve bu kapasiteye ulaşmak için orijinal işaret üzerinde yapılacak optimum değişikliğin belirlenmesi gereklidir.

B herhangi bir tamsayıyı belirtmek üzere, $[0, B)$ aralığında tamsayı değerler alan orijinal işaret $X = \{x, x \in \{0,1, \dots, B-1\}\}$ ve benzer şekilde damgalanmış işaret $Y = \{y, y \in \{0,1, \dots, B-1\}\}$ olarak tanımlansın. $H(\cdot)$ entropi fonksiyonunu göstermek üzere, verilen bir bozunum kısıtı Δ için damgalama kapasitesinin üst sınırı

$$C(\Delta) = max\{H(Y)\} - H(X) \tag{2.4}$$

ile verilebilir. Damgalanmış işaretin entropisi en büyüklenirken orijinal işarette oluşacak bozunum

$$\sum_{x,y} P_X(x).P_{Y|X}(y|x).D(x,y) \leq \Delta \tag{2.5}$$

eşitsizliğini sağlamalıdır. Denklem (2.5)'te $P_X(x)$ X'in olasılık yoğunluk fonksiyonunu, $P_{Y|X}(y|x)$ X altında Y'nin koşullu olasılık yoğunluk fonksiyonunu, $D(x,y)$ ise X ve Y arasındaki bozunum fonksiyonunu belirtmektedir. $D(x,y)$ genellikle $D(x,y) = (x-y)^2$ ile verilen karesel hata olarak seçilir.

2.4.1. Optimum damgalanmış işaret dağılımının hızlı kestirimi

Kapasite-Bozunum sınırını kestirmek amacıyla öncelikle Denklem (2.5)'e göre $P_{Y|X}(y|x)$ olasılıkları hesaplanmalıdır. Problemin dışbükeyliğinden dolayı çoğu dışbükey optimizasyon algoritması çözüm için kullanılabilir.

[39] ve [40]'daki çalışmalarda, Denklem (2.5) ile ifade edilen optimum kanal geçiş matrisinin bulunması probleminin "Kesişmeyen Kenarlar" özelliğine sahip olduğu

ispatlanmıştır. (Diğer bir deyişle, bir $P_{Y|X}$ optimum ise, herhangi iki ayrık olasılık geçiş durumları $P_{Y|X}(y_1|x_1) > 0$ ve $P_{Y|X}(y_2|x_2) > 0$ için $x_1 < x_2$ ise $y_1 \leq y_2$ sağlanır.) Lin ve arkadaşları, kesişmeyen kenarlar özelliğini kullanarak X ve Y için ortak olasılık dağılım fonksiyonunu aşağıdaki gibi ifade etmiştir [39].

$$P_{X,Y}(x,y) = \max\left\{0, \min\left(P_{CX}(x), P_{CY}(y)\right) - \max\left(P_{CX}(x-1), P_{CY}(y-1)\right)\right\} \quad (2.6)$$

Yukarıdaki denklemde P_{CX} ve P_{CY} sırasıyla X ve Y işaretlerinin toplam olasılık dağılım fonksiyonları olup nasıl hesaplandıkları Denklem (2.7)'de verilmiştir.

$$\begin{aligned} P_{CX}(x) &= \textstyle\sum_{i=0}^{x} P_X(i) \qquad x = 0, \dots, B-1 \\ P_{CY}(y) &= \textstyle\sum_{i=0}^{y} P_Y(i) \qquad y = 0, \dots, B-1 \end{aligned} \quad (2.7)$$

Toplam dağılım fonksiyonu 0 ile 1 arasında değer almaktadır. İlgili fonksiyonların argümanlarının alt sınıra eşit olması halinde 0; üst sınıra eşit olması halinde 1 değerini aldığı varsayılmıştır. Yani, $P_{CX}(-1) = 0$, $P_{CY}(-1) = 0$, $P_{CX}(B-1) = 1$ ve $P_{CY}(B-1) = 1$.

Böylelikle Denklem (2.5)'teki problem, damgalanmış işaretin optimal marjinal dağılımı $P_Y(y)$'nin bulunmasına indirgenmiştir. [39]'daki çalışmada, $P_Y(y)$'yi kestirmek için geri ve ileri yönde yinelemeli bir algoritma önerilmiştir. [41]'deki çalışmada [39]'dakine göre daha hızlı bir algoritma sunulmuştur. Bu tezde $P_Y(y)$ 'nin hesaplanmasında [41]'de verilen algoritma kullanılmıştır. $P_Y(y)$ elde edildikten sonra (2.6) ve (2.7) denklemleri yerine koyularak ortak olasılık dağılım fonksiyonu $P_{X,Y}(x,y)$ bulunur.

Daha sonra, X'den Y'ye ve Y'den X'e optimum olasılık geçiş matrisleri $Q_{Y|X}$ ve $Q_{X|Y}$,

$$\begin{aligned} Q_{Y|X}(x,y) &= [P_{X,Y}(x,y)/P_X(x)]^T \\ Q_{X|Y}(x,y) &= [P_{X,Y}(x,y)/P_Y(y)] \end{aligned} \quad (2.8)$$

eşitliklerinden hesaplanabilir. Denklem (2.5) sağlanacak şekilde Denklem (2.4)'ün çözümü, (eşdeğer olarak $P_{Y|X}(y|x)$ koşullu olasılıkların bulunması problemi) aslında orijinal işaretin histogramı üzerinde izin verilen optimum değişikliğin tespiti anlamına gelmektedir. Denklem (2.4) ve (2.5) ile verilen kısıtlı optimizasyon problemi [41]'de çözülmüştür. Hu ve arkadaşlarının çözümü, X'den Y'ye optimum olasılık geçiş matrisleri $Q_{Y|X}$ ve $Q_{X|Y}$'yi vermektedir. $Q_{Y|X}$, Δ bozunum kısıtı altında Y işaretinin sahip olması gereken olasılık dağılımlarını tutan bir matristir. Ancak, bu dağılımla sonuçlanacak damgalama işleminin nasıl yapılacağı bilinmemektedir. Bu amaçla, [34]'te Denklem (2.4)'te verilen kapasite-bozunum sınırına yakınsayan bir TGD yöntemi geliştirilmiştir. Bu tezde önerilen TVD algoritmaları [34]'te öne sürülen YHD'ye dayalıdır. YHD yöntemini tanıtmadan önce, algoritmanın önemli bir kısmını oluşturan aritmetik kodlama yönteminin tartışılması algoritmanın kavranılmasını kolaylaştıracaktır.

2.4.2. Aritmetik kodlama

Kodlama, verilen herhangi bir sembol dizisini bir bit dizisine dönüştürme olarak tanımlanabilir. Kodlamanın tersinir olması durumunda, bit dizisinden sembol dizisi hatasız olarak geri elde edilebilir. Tersinir bir kodlama yöntemi olan aritmetik kodlama, ilgili sembol dizileri üzerinde aritmetik işlemler gerçekleştirir. Aritmetik kodlama, her bir giriş sembolüne özel bir kod kelimesi atama yerine giriş sembollerinin istatistiksel dağılımına göre bir kodlama stratejisi geliştirir. Diğer bir deyişle, aritmetik kodlama yöntemiyle oluşturulmuş bir kod kelimesi mesajın dağılımı ile ilgili bir bilgi vermektedir. Bu nedenle aritmetik kodlama, YHD algoritmasının önemli bir parçasını oluşturmaktadır.

Aritmetik kodlamada, kodlanacak veri [0,1] aralığında bir aralıkla temsil edilir. Verinin boyutu arttıkça aralığın uzunluğu azalır ve bu aralığı temsil edecek bit sayısı artar. Başlangıçta veriye karşılık gelen aralığın [0,1) olduğu varsayılır. Bu aralık, sembol olasılıkları dikkate alınarak sembol sayısı kadar aralığa bölünür. Verilen bir

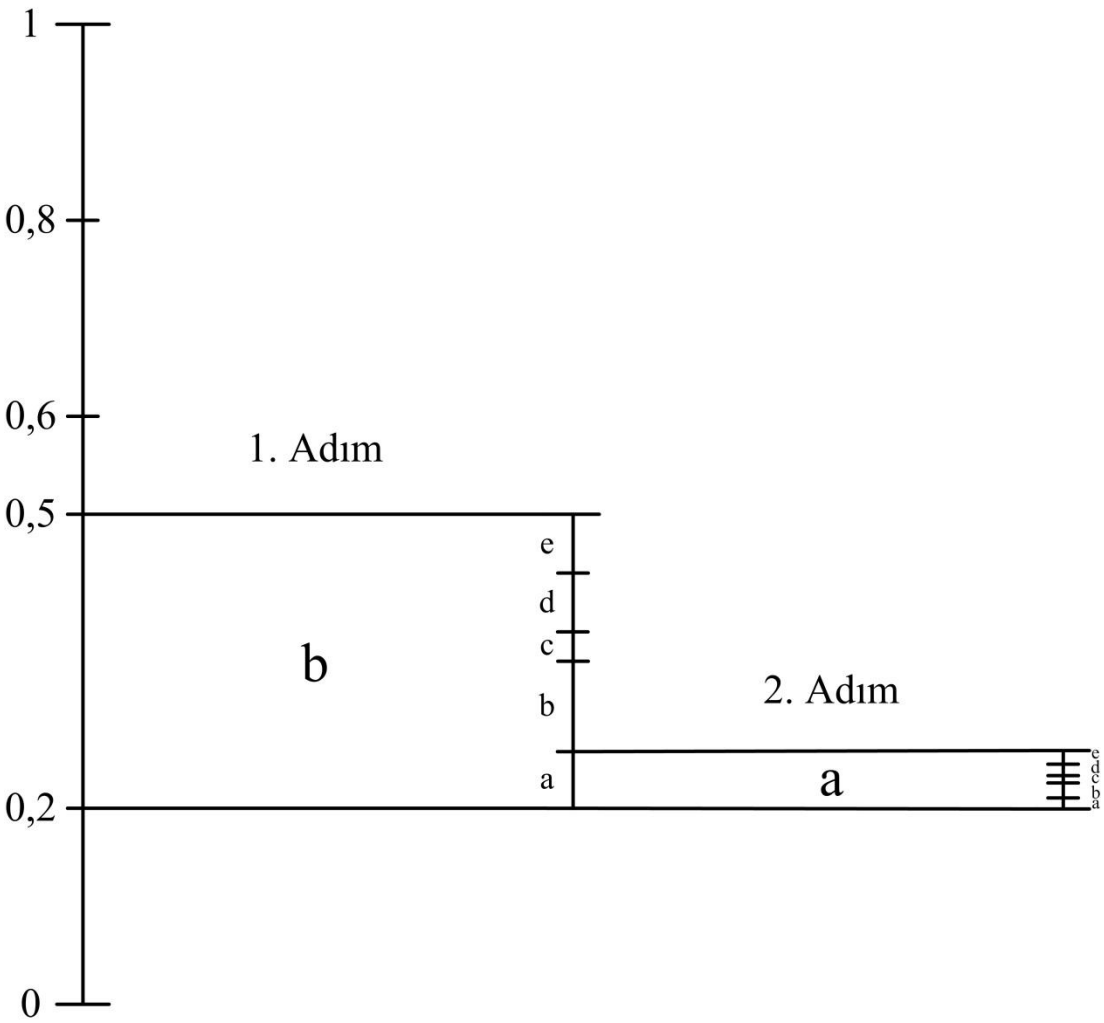
Tablo 2.1. Örnekteki sembollerin olasılık dağılımı.

Sembol	Olasılık (P)	Toplam Olasılık (C)	Aralık
a	0.2	0	[0, 0.2)
b	0.3	0.2	[0.2, 0.5)
c	0.1	0.5	[0.5, 0.6)
d	0.2	0.6	[0.6, 0.8)
e	0.2	0.8	[0.8, 1.0)

sembol kodlandığında kalan veriye ilişkin aralık bir önceki aralığın ilgili sembole ait kısmına daraltılır.

Örneğin, kodlanacak verinin {a, b, c, d, e} sembollerinden oluştuğu varsayılsın ve sembollerin olasılık dağılımları Tablo-1'de verildiği gibi olsun. [b a c c e] verisine karşılık gelen kod kelimesini belirleyelim. Başlangıçta, kodlayıcı veriye ilişkin aralığın [0, 1) olduğunu kabul eder. Kodlayıcı, ilk sembol b'yi işlerken aralığı [0.2, 0.5) aralığına daraltır. İkinci sembol a kodlanırken, öncelikle bir önceki sembolün aralığı olan [0.2, 0.5) aralığı tüm sembollere Tablo 2.1.'de verilen olasılık dağılımlarına göre dağıtılır ve yeni aralık [0.2, 0.26) olarak elde edilir. Şekil 2.5.'te öz yinelemeli bu işlemler grafiksel olarak gösterilmiştir. Bu şekilde her bir sembol için bir önceki aralık, sembolün olasılık değerine göre daraltılarak yeni aralık elde edilir. Bu işlem tüm semboller için tekrar edilerek örnekteki veri için Tablo 2.2.'de verilen aralıklar elde edilir.

Şekil 2.5.'te verilen grafiksel gösterim çok sayıda sembolden oluşan veriler için uygun değildir. Bunun yerine, Şekil 2.6.'da verilen alternatif gösterim tercih edilir. Her bir sembol için veri aralığı sembollerin dağılımına göre şekildeki gibi ölçeklenir ve iletilecek sembole göre bir sonraki sembole karşılık gelen aralık belirlenir.



Şekil 2.5. Aritmetik kodlama sürecinin gösterimi.

Herhangi bir veri için elde edilen aralığın en küçük değeri kod kelimesi olarak seçilebilir. Örnekteki sembol dizisi için kod kelimesi 0.23348 olarak seçilebilir. Yukarıda grafiksel olarak açıklanan aritmetik kodlama işlemi matematiksel olarak özyinelemeli iki işlem içermektedir. Birincisi, ilgili sembol için atanan aralığın en küçük değerinin hesaplanmasıdır. İkincisi ise, bu aralığın genişliğinin bulunmasıdır. Veriye karşılık gelen sembol dizisindeki herhangi bir sembol için elde edilecek aralığın en küçük değeri CV ve genişliği W ile gösterilsin. i. sembol için yeni kod noktası,

Tablo 2.2. Örneğe ait veri için semboller belirtilen sırada iletilirken karşılık gelen aralıklar.

Sembol	**Aralık**
Başlangıç	[0, 1)
b	[0.2, 0.5)
a	[0.2, 0.26)
c	[0.23, 0.236)
d	[0.233, 0.2336)
e	[0.23348, 0.23360)

$$CV_i = CV_{i-1} + (W_{i-1} \times C_i) \tag{2.9}$$

eşitliği ile hesaplanır. Denklem (2.9)'daki C_i, ilgili sembole ait toplam olasılık değerini ifade etmektedir. Bir sembole ait aralığın genişliği ise

$$W_i = W_{i-1} \times P_i \tag{2.10}$$

eşitliği ile elde edilir. Denklem (2.10)'daki P_i, ilgili sembole ait olasılık değerini belirtmektedir. Sembol dizisindeki ilk sembol için başlangıç değerlerinin Tablo 2.2'den de görülebileceği üzere $CV_0 = 0,\ W_0 = 1$ olduğu açıktır.

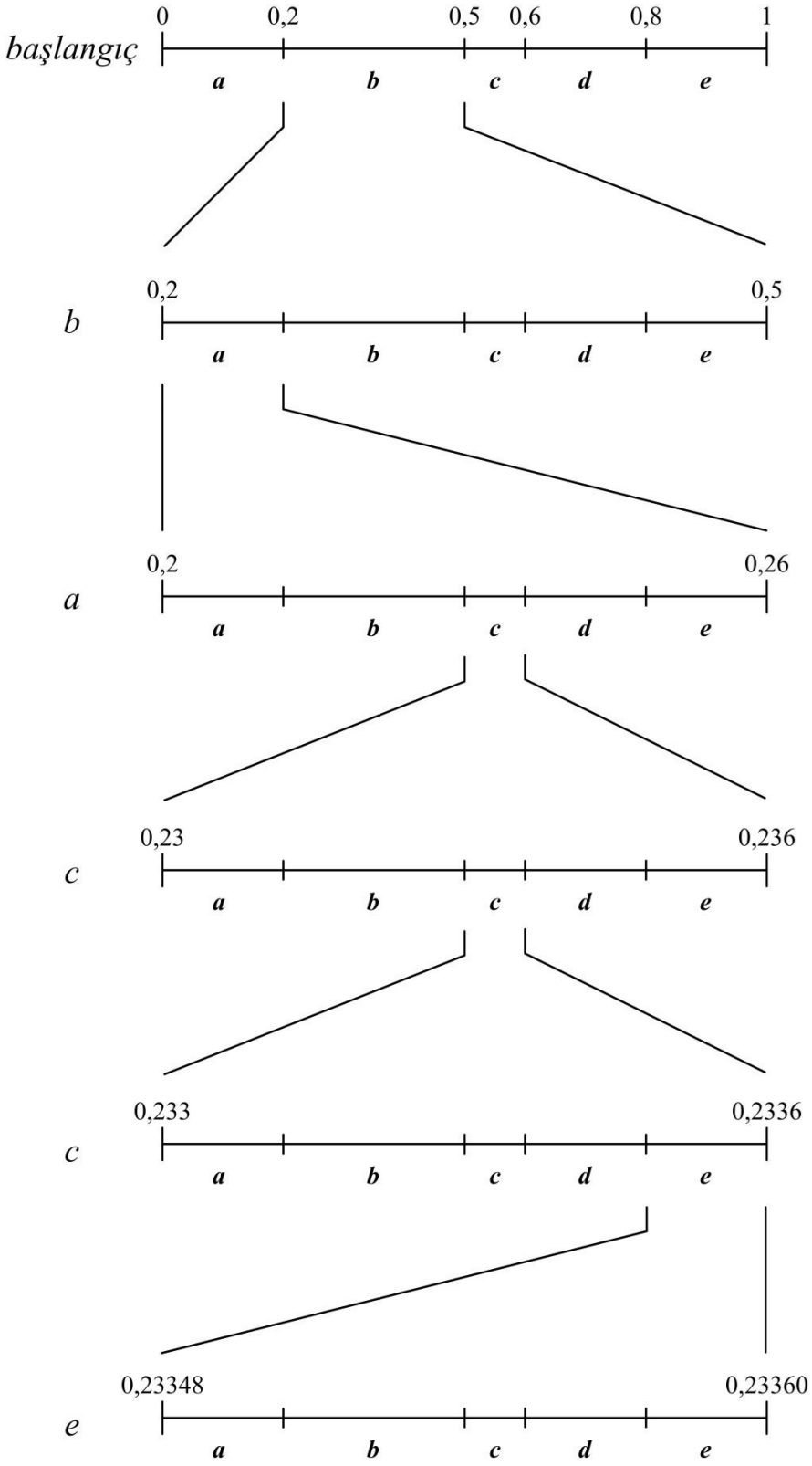
İlgili hesaplamalar örneğimiz için yapılacaktır. [b a c c e] sembol dizisinin ilk sembolü b için ilk önce $W_1 = W_0 \times P_b = 1 \times 0.3 = 0.3$ genişlik değeri hesaplanır. Daha sonra kod noktası değeri $CV_1 = CV_0 + (W_0 \times C_b) = 0 + (1 \times 0.2) = 0.2$ olarak hesaplanır. $CV_1 = 0.2$ ve $W_1 = 0.3$ değerleri Tablo-2'de verilen [0.2, 0.5) aralığına işaret etmektedir. Benzer şekilde daha sonra gelen a sembolü için $W_2 =$

$W_1 \times P_a = 0.3 \times 0.2 = 0.06$ genişliğinde $CV_2 = CV_1 + (W_1 \times C_a) = 0.2 + (0.3 \times 0) = 0.2$ kod değeri ile ifade edilen [0.2, 0.26) aralığı elde edilir.

Özetle, kodlayıcı mevcut kod noktası değeri ve aralık genişliği ile bir sonraki sembole ait aralığı, sembollerin olasılık ve toplam olasılık dağılımları yardımıyla sistemli bir biçimde hesaplayabilmektedir. Yukarıda anlatılan işlemler [b a c c e] dizisindeki tüm semboller için tekrar edilerek 0.23348 kod kelimesi elde edilir. Başka bir deyişle, [b a c c e] sembol dizisi mesajı aritmetik kodlayıcı ile kodlandığında 0.23348 kod kelimesi elde edilmektedir.

Kod çözücü [0.23348, 0.2336) aralığında herhangi bir sayıya karşılık gelen sembolün b olduğuna karar verir. Bunun nedeni, 0.23348 sayısının Tablo 2.1.'de belirtilen aralıklardan b sembolüne karşılık gelen aralığa düşmesidir. İletilen ilk sembol b olarak belirlendikten sonra iletilen sonraki sembol için ilgili aralık Denklem (10) yardımıyla [0.2, 0.5) olarak hesaplanır. Bu aralık Tablo 2.1.'de verilen sembol olasılık dağılımlarına göre alt aralıklara bölündüğünde 0.23348 sayısı bu yeni cetvelde a sembolüne karşılık gelen aralığa düştüğünden ikinci sembol a olarak elde edilir ve bir sonraki sembolün elde edilebilmesi için yeni aralık Denklem (10) yardımıyla yeniden hesaplanır. Bu işlemler, tüm semboller elde edilene kadar tekrar edilir. Bununla birlikte, kod çözücü kod çözme işlemini ne kadar sürdüreceğini bilmek zorundadır. Bu yüzden, aritmetik kodlamada kodlayıcı ve kod çözücü tarafından bilinen bir mesaj sonu (EOF) sembolü kullanılır.

Aritmetik kodlamada, kod kelimesi aslında kod çözücüye kodlayıcıda hangi işlemlerin yapıldığı bilgisini vermektedir. Tablo 2.1.'de verilen bilgilerin kod çözücü tarafından da bilindiği varsayılırsa kod çözücüde gerçekleştirilen matematiksel işlemler, öncelikle kod kelimesinin başlangıç aralığı [0,1)'de hangi sembole ait alt aralığa karşı geldiği belirlenerek ilk sembolün elde edilmesiyle başlar. Bundan sonra farklı iki kod çözücü algoritma tasarlanabilir. Birinci çözümde elde edilen her bir sembol için başlangıç aralığı yukarıda açıklandığı şekilde daraltılıp kod kelimesinin o aralıktaki hangi sembole karşılık geldiğine bakılarak sonraki sembol elde edilir.



Şekil 2.6. Tablo 2.1.'de verilen sembol olasılıkları için [b a c c e] sembol dizisine karşılık gelen aralıklar.

İkinci çözümde ise, ilk sembol çözüldükten sonra başlangıç kod kelimesi kullanılarak bir sonraki sembol için yeni kod noktası değeri hesaplanabilir. Bu yeni kod noktasının, başlangıç aralığı [0,1)'de hangi sembole karşılık geldiği bulunarak bir sonraki sembol elde edilmiş olur. İkinci yöntemdeki yeni kod noktasının değeri

$$CV_i = \frac{CV_{i-1} - C_{i-1}}{P_i} \tag{2.11}$$

denklemi ile hesaplanır. Kod çözücü, 0.23348 sayısını işlerken bu değer [0,1) aralığında *b* sembolüne karşılık geldiğinden ilk sembol olarak *b* seçilir ve bir sonraki sembol için kod noktası değeri Denklem (2.11) yardımıyla

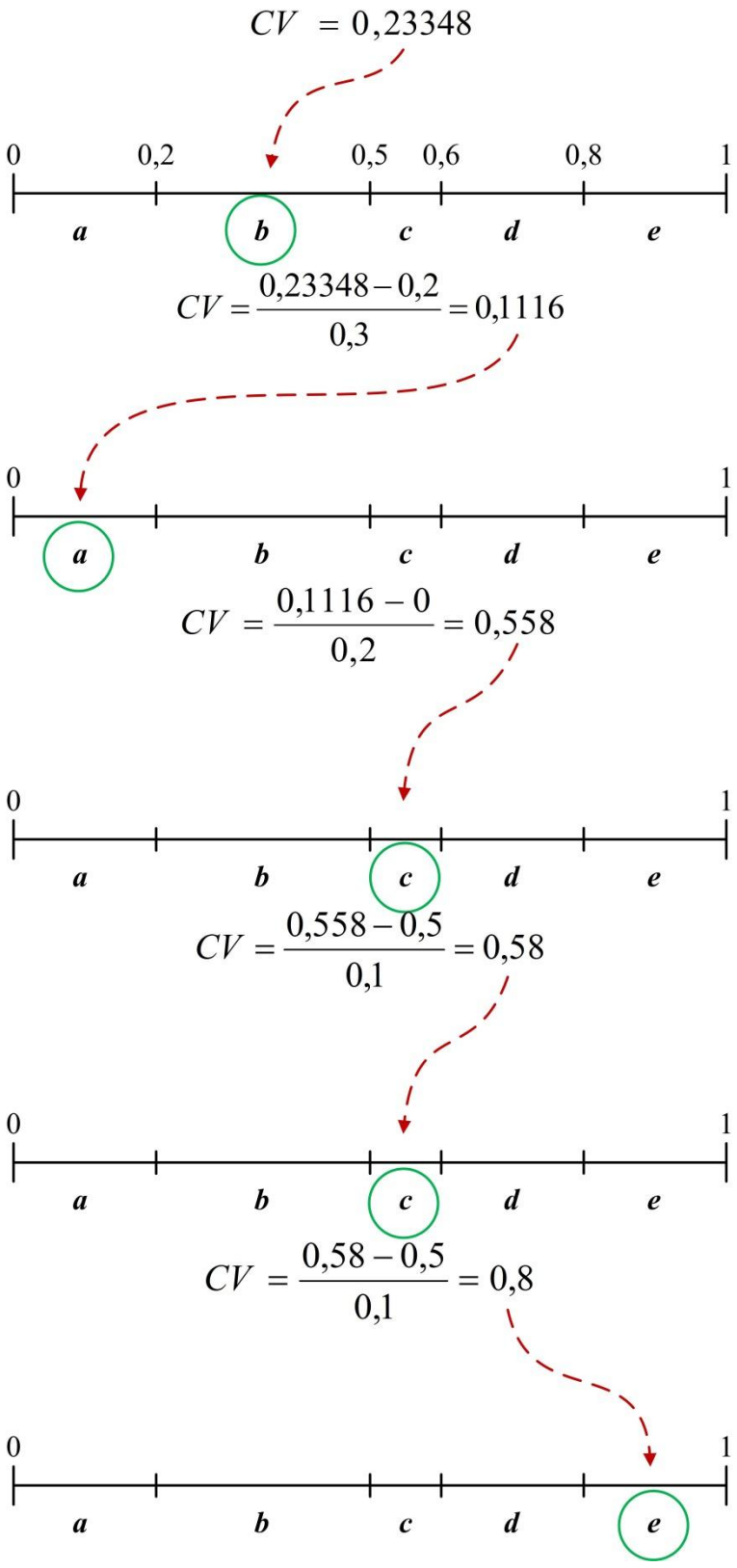
$$\mathrm{CV_i} = \frac{(\mathrm{CV_{i-1}} - \mathrm{C_{i-1}})}{\mathrm{P_i}} = \frac{(0.23348 - 0.2)}{0.3} = 0.1116$$

olarak elde edilir. 0.1116 sayısı [0,1) aralığında *a* sembolüne karşılık geldiğinden ikinci sembol *a* olarak elde edilir. Bu işlemler Şekil 2.7.'de gösterildiği gibi tüm semboller çıkartılana kadar devam ettirilir.

Yukarıda detayları verilen aritmetik kodlama yöntemi, kodlanacak verideki sembollerin belirme olasılıklarında büyük farkların olduğu durumlarda etkin bir kodlama sunmaktadır. Kod çözücüde verinin geri elde edilebilmesi için sembol olasılıkları bilinmelidir. Aritmetik kodlamada kod çözücü, elindeki bit dizisini gönderilen mesajın dağılımına uygun bir biçimde alfabedeki sembol dizisine dönüştürmektedir. Bu yapısıyla aritmetik kodlama yöntemi aşağıda detayları verilen YHD yönteminin temel bileşenlerinden birini oluşturmaktadır [42-43].

2.4.3. Yinelemeli histogram değiştirme

Damgalanmış işaretin sahip olması gereken dağılım hesaplandıktan sonra, orijinal işaretin bu hedef dağılıma sahip olacak şekilde nasıl değiştirileceği ve bu esnada damganın nasıl saklanacağı üzerinde çalışılan güncel bir problemdir. YHD yönteminde, eklenecek bit dizisi aritmetik kodlama algoritması kullanılarak ters



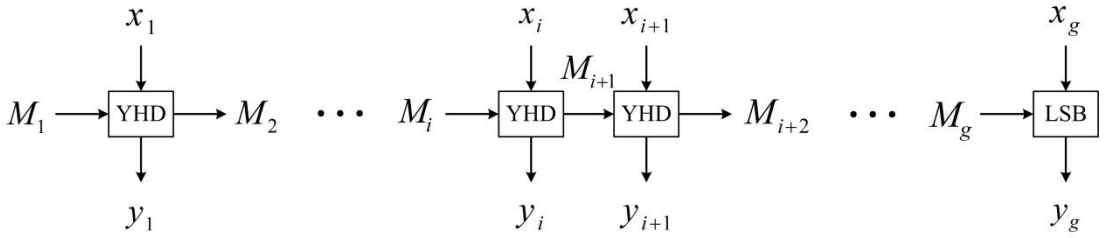
Şekil 2.7. Aritmetik kod çözücü için grafiksel gösterim.

sıkıştırılıp dağılımı hedef dağılıma eşit olan bir sembol dizisi oluşturulur. Daha sonra, bu sembol dizisi orijinal sembol dizisi ile değiştirilerek damgalama gerçekleştirilir. Yöntemde, orijinal işaret bloklara ayrılmaktadır ve histogram değiştirme yöntemi her bir bloğa ayrı ayrı uygulanarak damga eklenmektedir. $P_X(x), x \in \{0,1,\dots,B-1\}$ dağılımlı orijinal işaret X, g adet örtüşmeyen bloklara ayrılır. Her bir blok ayrı ayrı damgalama işlemine tabi tutulur. 0 ve 1'lerden oluşan damga M dizisi ile gösterilsin. $P_X(x)$ ve Δ bozunum kısıtı kullanılarak hedef dağılım $P_Y(y)$ hesaplanır. Her bir orijinal blok, dağılımı hedef dağılıma eşit olacak şekilde damgalanır. Orijinal bloktaki her $x \in \{0,1,\dots,B-1\}$ değerinin damgalı blokta hangi değerleri alabileceği $Q_{Y|X}$ matrisinin ilgili sütununda verilmiştir. Bu dağılıma göre, bit dizisi M orijinal işaretteki her bir $x \in \{0,1,\dots,B-1\}$ değer için aritmetik kodlama ile ters sıkıştırılıp $y \in \{0,1,\dots,B-1\}$ değerlerinden oluşan bir sembol dizisine dönüştürülür. Daha sonra, orijinal işaretteki x değerleri y değerleri ile değiştirilerek damgalama işlemi gerçekleştirilir.

Damga dizisi M'nin i. blok damgalanmadan önceki hali M_i, damgalama işlemi sonrası hali M_{i+1} olmak üzere, blokların damgalanması Şekil 2.8.'de gösterilmiştir. x_i bloğu damgalanırken M_i damga dizisinin başından, bloğa eklenecek kadar bit alınıp veri ekleme işlemi gerçekleştirilir. Orijinal blok ve eklenen damganın, damga çözücüde kayıpsız geri elde edilebilmesi için gerekli yan bilgi $O(x_i)$ oluşturulup M_i'nin kalan kısmının başına eklenmesiyle bir sonraki bloğa ilişkin damga dizisi M_{i+1} oluşturulur.

Damgalama işleminde i. bloğa eklenen bit dizisi $M_{e,i}$, ve M_i dizisinden geriye kalan kısım $M_{r,i}$ ile gösterilsin. $\oplus$, bitsel temsilde uç uca eklemeyi ifade etmek üzere, $M_i = M_{e,i} \oplus M_{r,i}$ yazılabilir. Mevcut bloğa eklenecek bit dizisi

$$M_{e,i} = M_i(1: C(x_i)) \tag{2.12}$$



Şekil 2.8. YHD yöntemi ile blok tabanlı damgalama.

denklemiyle elde edilir. Denklem (2.12)'de C(x_i), x_i bloğunun damgalama kapasitesini, $M(P_1{:}P_2)$ notasyonu ise M dizisinin P_1'den P_2'ye kadar olan elemanlarını ifade etmektedir. M_i dizisinden geriye kalan bit dizisi,

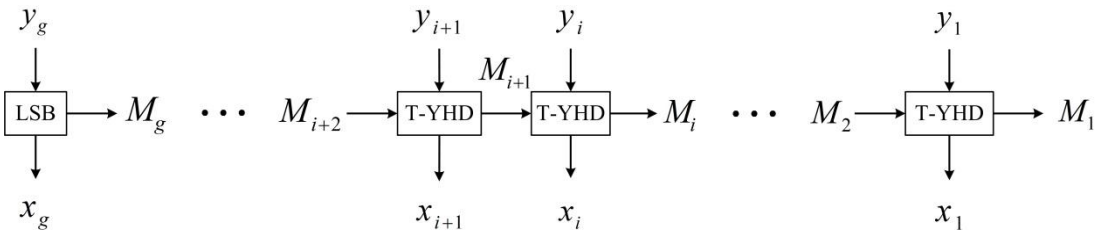
$$M_{r,i} = M_i(C(x_i) + 1{:}\, uzunluk(M_i))$$

eşitliğinden belirlenebilir. *i.* blok için gerekli yan bilgi O(x_i),

$$M_{i+1} = \mathrm{O}(x_i) \oplus M_{r,i}$$

Eşitliği ile damga dizisinin kalan kısmının başına eklenerek (i+1). bloğa ait damga dizisi elde edilir. Yukarıda belirtilen işlemler birinci bloktan başlayarak, sırasıyla tüm bloklara uygulanarak damgalama gerçekleştirilmiş olur.

Damga çıkartmada, Şekil 2.9.'da gösterildiği gibi son bloktan başa doğru işlemler gerçekleştirilir. Şekil-2.9.'daki LSB bloğu en az anlamlı bit değiştirme yöntemi ile veri saklamayı ifade etmektedir. Damgalanmış her bir blok y_i için iki işlem gerçekleştirildiğine dikkat ediniz. İlk olarak, (y_{i+1}). blok orijinal haline dönüştürülürken oluşturulan M_{i+1} içindeki yan bilgiler kullanılarak orijinal blok x_i elde edilir. İkinci aşamada, (i-1). bloğun işlenmesinde gerekli olan M_i bit dizisi oluşturulur.



Şekil 2.9. YHD yöntemi ile blok tabanlı damga çıkartma ve orijinal blokları geri elde etme.

2.5. Sayısal Örnek

2.5.1. Damga ekleme

YHD yönteminin daha iyi bir şekilde anlaşılabilmesi için sadece bir bloğun damgalanması sayısal bir örnek üzerinden aşağıda açıklanmıştır. Blok uzunluğunun 16 olduğunu ve bloğun $\{1, 2, 3, 4\}$ değerlerinden oluştuğunu kabul edelim. Damgalanacak blok e_i^k aşağıda verilmiştir.

$$e_i^k = [1\ \ 2\ \ 4\ \ 3\ \ 4\ \ 2\ \ 4\ \ 3\ \ 3\ \ 4\ \ 4\ \ 1\ \ 4\ \ 3\ \ 4\ \ 4]$$

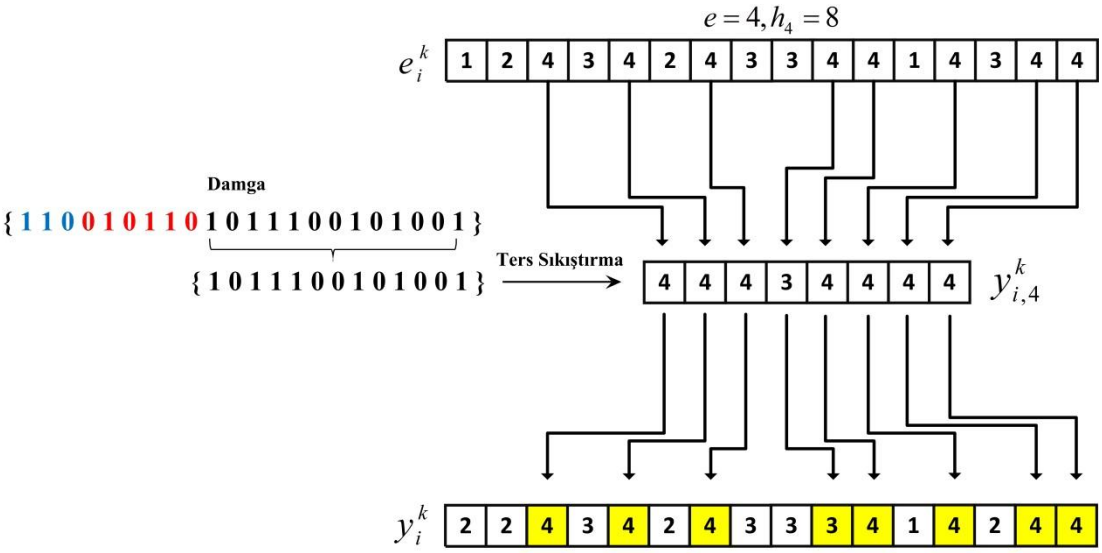
İlgili olasılık geçiş matrislerinin, $P_X(x)$ ve Δ yardımıyla

$$Q_{Y|X} = \begin{bmatrix} 1/2 & 0 & 0 & 0 \\ 1/2 & 1 & 1/4 & 0 \\ 0 & 0 & 3/4 & 1/8 \\ 0 & 0 & 0 & 7/8 \end{bmatrix}$$

$$Q_{X|Y} = \begin{bmatrix} 1 & 1/4 & 0 & 0 \\ 0 & 2/4 & 0 & 0 \\ 0 & 1/4 & 3/4 & 0 \\ 0 & 0 & 1/4 & 1 \end{bmatrix}$$

şeklinde hesaplandığını varsayalım. Bloğa eklenecek damga dizisi

$$M_i = [1\ 1\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 1 \ldots]$$



Şekil 2.10. Örnek bloktaki ‘4’ sembolünün damgalanması.

ile verilsin. e_i^k bloğunda $h_1 = 2$ adet ‘1’ olduğu için M_i bit dizisi 2 sembol uzunluğunda bir çıktı verene kadar $Q_{Y|X}$ matrisinin ilk sütununda verilen dağılıma göre yinelemeli bir ters sıkıştırma işlemine sokulur. Bu işlemin sonucunda M_i’nin ilk 3 biti (1 1 0) ters sıkıştırılarak $y_{i,1}^k = [2 \;\; 1]$ sembol dizisi elde edilir. Sonra e_i^k bloğundaki ‘1’ sembolleri $y_{i,1}^k$ sembol dizisi ile değiştirilir. Bu işlem e_i^k bloğundaki ‘3’ ve ‘4’ sembolleri için tekrarlanır. M_i’nin sonraki 6 biti (0 1 0 1 1 0) $Q_{Y|X}$ matrisinin 3. sütunundaki dağılıma göre ters sıkıştırılarak $y_{i,3}^k = [3 \;\; 3 \;\; 3 \;\; 2]$ elde edilir. Benzer şekilde sıradaki 13 adet bit $Q_{Y|X}$ matrisinin 4. sütunundaki dağılıma göre ters sıkıştırılarak $y_{i,4}^k = [4 \;\; 4 \;\; 4 \;\; 3 \;\; 4 \;\; 4 \;\; 4 \;\; 4]$ sembol dizisi elde edilir. $y_{i,3}^k$ ve $y_{i,4}^k$ dizileri sırasıyla orijinal bloktaki ‘3’ ve ‘4’ sembolleri ile değiştirilir. $Q_{Y|X}$ matrisinin 2. sütunundaki dağılımın entropisi sıfıra eşit olduğu için ‘2’ sembolüne damgalama işlemi yapılmaz. Bu işlemler sonucunda aşağıda verilen damgalanmış blok y_i^k elde edilir:

$$y_i^k = [2 \;\; 2 \;\; 4 \;\; 3 \;\; 4 \;\; 2 \;\; 4 \;\; 3 \;\; 3 \; 3 \;\; 4 \;\; 1 \;\; 4 \;\; 2 \;\; 4 \;\; 4]$$

Gerçekleştirilen işlemler ‘4’ sembolü için Şekil 2.10.’da gösterilmiştir. Bloktaki tüm semboller damgalandıktan sonra damga çözücünün y_i^k ve $Q_{X|Y}$ yardımıyla orijinal e_i^k bloğunu elde edebilmesi için gerekli yan bilginin oluşturulması gerekmektedir. Önce

damgalanmış bloktaki ‘1’,’2’,’3’ ve ‘4’ sembollerinin konumları $IY_1 = [12]$, $IY_2 = [1\ \ 2\ \ 6\ \ 14]$, $IY_3 = [4\ \ 8\ \ 9\ \ 10]$ ve $IY_4 = [3\ \ 5\ \ 7\ \ 11\ \ 13\ \ 15\ \ 16]$ elde edilir. $Q_{X|Y}$ matrisinin 1. ve 4. sütunlarındaki dağılımların entropileri sıfıra eşit olduğu için ‘1’ ve ‘4’ sembollerinin sıkıştırılıp damga çözücüye iletilmesine gerek yoktur (entropilerin sıfıra eşit olması, damgalanmış bloktaki ‘1’ ve ‘4’ sembollerinin orijinal bloktada ‘1’ ve ‘4’ olarak yer alması anlamına gelir). Daha sonra, orjinal blokta IY_2 ve IY_3 ile gösterilen konumlardaki sembol dizileri $Q_{X|Y}$ matrisinin ilgili sütunları yardımıyla sıkıştırılıp uçuca eklenerek $O(e_i^k)$ yan bilgisi

$$O\left(e_i^k\right) = O\left(e_i^k, 2\right) \| O\left(e_i^k, 3\right) = [1\ \ 1\ \ 0\ \ 1\ \ 0\ \ 0\ \ 1\ \ 0\ \ 0\ \ 1\ \ 0\ \ 1]$$

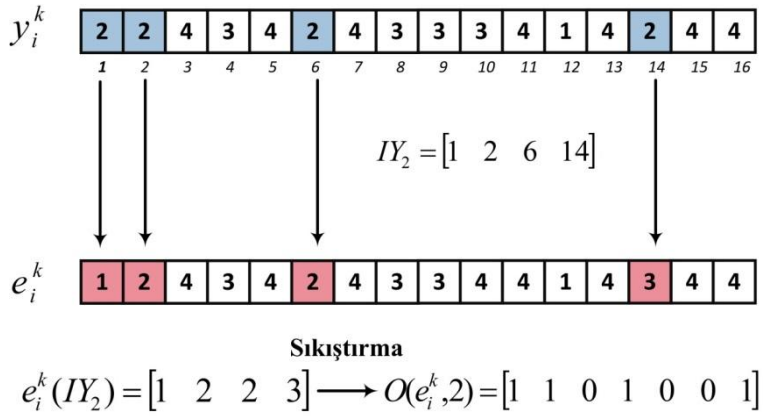
şeklinde elde edilir. IY_2 ile gösterilen semboller $e_i^k(IY_2) = [1\ \ 2\ \ 2\ \ 3]$ sıkıştırıldığında $O\left(e_i^k, 2\right) = [1\ \ 1\ \ 0\ \ 1\ \ 0\ \ 0\ \ 1]$ bit dizisi, IY_3 ile gösterilen semboller $e_i^k(IY_3) = [3\ \ 3\ \ 3\ \ 4]$ sıkıştırıldığında $O\left(e_i^k, 3\right) = [0\ \ 0\ \ 1\ \ 0\ \ 1]$ bit dizisi elde edilir. Elde edilen bu yan bilgi bir sonraki bloğun damga bit dizisinin başına

$$M_{i+1} = [\mathbf{1\ \ 1\ \ 0\ \ 1\ \ 0\ \ 0\ \ 1\ \ 0\ \ 0\ \ 1\ \ 0\ \ 1}\ 0\ \ 1\ \ 1\ \ .\ \ .\ \ .]$$

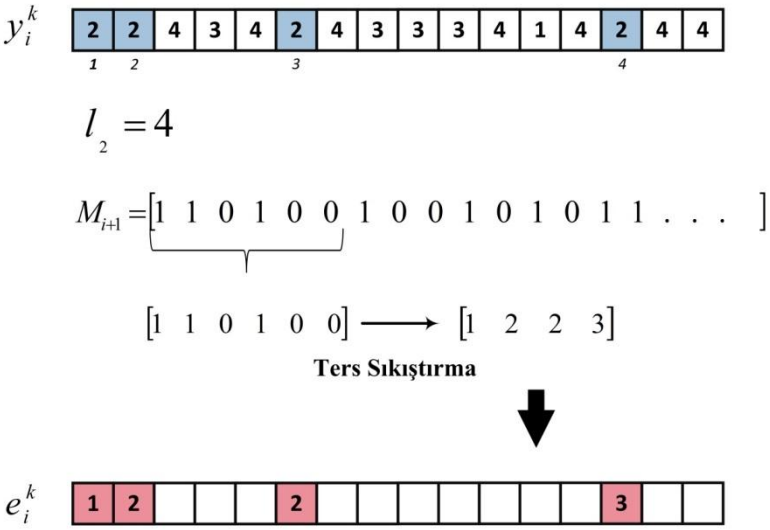
şeklinde eklenir. Yan bilgilerin oluşturulması işlemi ‘2’ sembolü için Şekil 2.11.’de gösterilmiştir.

2.5.2. Damga çıkartma ve orijinal bloğun geri elde edilmesi

Damga çözücü tarafında öncelikle $Q_{X|Y}$ ve y_i^k yardımıyla orijinal blok e_i^k elde edilir. $Q_{X|Y}$ matrisinin 1. ve 4. sütunlarına bakılarak damgalanmış bloktaki ‘1’ ve ‘4’ sembollerinin orijinal blokta da yer alacakları anlaşılır. Sonrasında, damgalanmış bloktaki ‘2’ sembolünün tekrarlanma sayısı $l_2 = 4$ hesaplanır ve (i+1). bloğun damga dizisi, l_2 uzunluğunda bir sembol dizisi verene kadar $Q_{X|Y}$ matrisinin 2. sütununda verilen dağılım altında aritmetik ters sıkıştırma algoritmasına tabi tutulur. Aritmetik

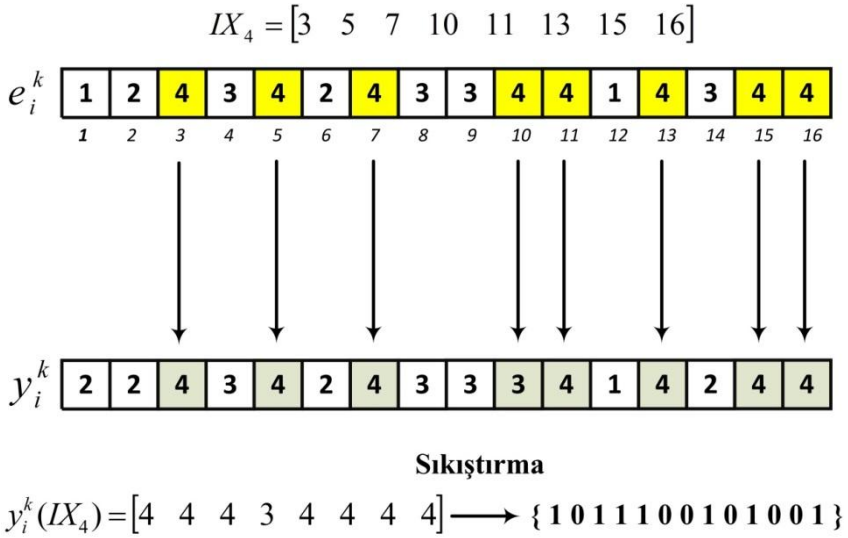


Şekil 2.11. '2' sembolü için yan bilginin oluşturulması.



Şekil 2.12. '2' sembolü için orijinal bloğun geri elde edilmesi.

kodlama algoritması tersinir olduğundan bu işlem, [1 1 0 1 0 0 1] bit dizisinin ters sıkıştırılarak [1 2 2 3] sembol dizisinin elde edilmesi ile sonuçlanır. Elde edilen sembol dizisi y_i^k bloğundaki '2' sembolleriyle yer değiştirilir. Aynı işlem



Şekil 2.13. ‘4’ sembolü için damganın geri elde edilmesi.

damgalanmış bloktaki ’3’ sembolleri için tekrarlanarak orijinal blok e_i^k elde edilmiş olur. Orijinal bloğun geri elde edilmesi işlemi ‘2’ sembolü için Şekil 2.12.’de gösterilmiştir.

Orijinal blok elde edildikten sonra, orijinal blok e_i^k, damgalanmış blok y_i^k ve $Q_{Y|X}$ dağılımları kullanılarak damga geri çatılır. Öncelikle e_i^k orijinal bloktaki ‘1’,’2’,’3’ ve ‘4’ sembollerinin konumları $IX_1 = [1\ \ 12]$, $IX_2 = [\,2\ \ 6]$, $IX_3 = [4\ \ 8\ \ 9\ \ 14]$ ve $IX_4 = [3\ \ 5\ \ 7\ \ 10\ \ 11\ \ 13\ \ 15\ \ 16]$ dizilerinde saklanır. $Q_{Y|X}$ matrisinin 2. sütununa bakılarak orijinal bloktaki ‘2’ sembollerinin damgalanmadığı anlaşılır. Diğer semboller için, damgalanmış blokta IX_1, IX_3 ve IX_4 dizileri tarafından belirtilen ilgili sembol dizileri $y_i^k(IX_1) = [2\ \ 1]$, $y_i^k(IX_3) = [3\ \ 3\ \ 3\ \ 2]$ ve $y_i^k(IX_4) = [4\ \ 4\ \ 4\ \ 3\ \ 4\ \ 4\ \ 4\ \ 4]$ $Q_{Y|X}$ matrisinin ilgili sütunlarında verilen dağılımlar altında sıkıştırılarak damga dizisi elde edilir. $y_i^k(IX_1)$ sıkıştırılarak [1 1 0] bitleri, $y_i^k(IX_3)$ sıkıştırılarak [0 1 0 1 1 0] bitleri ve $y_i^k(IX_4)$ sıkıştırılarak [1 0 1 1 1 0 0 1 0 1 0 0 1] bitleri elde edilir. Elde edilen bit dizileri uç uca eklenerek *i.* bloğa ait damga,

$$M_i = [\mathbf{1\,1\,0\,0\,1\,0\,1\,1\,0\,1\,0\,1\,1\,1\,0\,0\,1\,0\,1\,0\,0\,1}\,0\,1\,1\ldots]$$

elde edilir. Damganın geri elde edilmesi işlemi ‘4’ sembolü için Şekil 2.13.’te gösterilmiştir.

BÖLÜM 3. ŞİFRESİZ VİDEOLAR İÇİN YİNELEMELİ HİSTOGRAM DEĞİŞTİRME TABANLI TERSİNİR VİDEO DAMGALAMA

3.1. Giriş

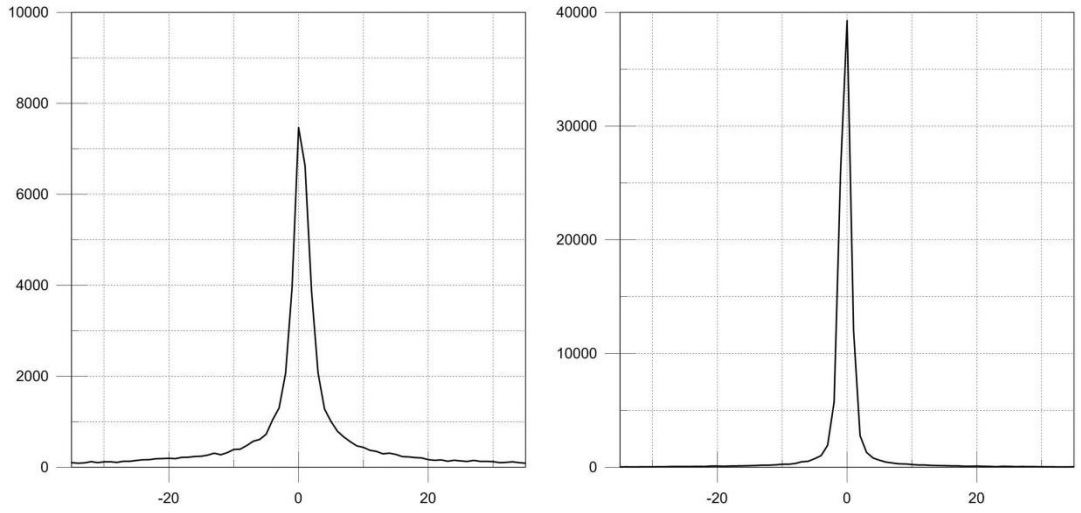
Hemen hemen bütün TD yöntemleri iki aşamadan oluşur. Birinci aşamada orijinal işaretten daha düşük entropili (yani daha dik ve dar histogramlı) bir işaret elde edilir. Bu problem, görüntü damgalamada öngörü hataları, aradeğerleme hataları veya komşu piksel farkları kullanmak suretiyle çözülmüştür. İkinci aşamada, elde edilen düşük entropili işaretin histogramı tersinir olacak şekilde değiştirilir. Bir önceki bölümde özetlenen YHD yöntemi, TGD amacıyla görüntü pikselleri arasındaki aradeğerleme hatalarına uygulanmıştır. Bu bölümde, YHD yönteminin video işaretlerinin damgalanmasında nasıl kullanılabileceği araştırılmış ve YHD yöntemine dayalı bir TVD yöntemi geliştirilmiştir. YHD yöntemi video çerçeveleri arasındaki HDÇA hatalarını damgalamak için kullanılmıştır. Video çerçevelerinin HDÇA'larını hesaplamak için [44]'te verilen yöntem kullanılmıştır.

Bu bölümde, şifresiz videolar için geliştirilen TVD algoritmasının detayları verilmiştir. Bölüm 3.2.'de video çerçeveleri arasındaki zamansal ilintinin nasıl değerlendirildiği açıklanmıştır. Verilen bir damgalama kapasitesinin video çeçrvelerine homojen bir biçimde dağıtılarak bozunumun tüm videoya paylaştırılması Bölüm 3.3.'te tartışılmıştır. Bölüm 3.4. ve 3.5.'te damgalama algoritmasının detayları verilmiştir. Alıcı tarafta damganın ve orijinal videonun nasıl kayıpsız geri elde edildiği Bölüm 3.6.'da anlatılmıştır. Önerilen yöntemin görsel kalite ve kapasite performansı bakımından mevcut yöntemlerle karşılaştırılması ve sonuçların değerlendirilmesi Bölüm 3.7.'de verilmiştir.

3.2. Çerçeve Aradeğerleme Hatası ve Yinelemeli Histogram Değiştirme

Hemen hemen tüm TGD yöntemleri, görüntüdeki pikseller arası uzamsal ilintiyi değerlendirerek, damgayı pikseller arası öngörü veya aradeğerleme hatalarına saklamışlardır. Video dizilerinde ise pikseller arası uzamsal ilintinin yanında çerçeveler arası zamansal ilinti mevcuttur. Ayrıca, zamansal ilinti uzamsal ilintiye göre daha yüksektir. Örneğin 'Paris' videosunda uzamsal ilinti katsayısı 0,86 iken zamansal ilinti katsayısı 0,99'dur. Dolayısıyla, video çerçeveleri arasındaki zamansal ilintiyi gideren ardeğerleme veya öngörü hatalarının histogramlarının bir çerçevedeki pikseller arası uzamsal ilintiyi gideren aradeğerleme veya öngörü hatalarının histogramlarından daha dar ve keskin olması beklenir. Şekil 3.1.'de 'Hall Monitor' videosundaki bir çerçeve içi hesaplanmış zamansal ve uzamsal aradeğerleme hatalarının histogramları verilmiştir. TGD yöntemleri dar ve keskin histogramlı işaretler için daha etkin damgalama yaptıklarından, video damgalamada çerçeveler arası aradeğerleme veya öngörü hatalarının kullanılması daha iyi sonuçlar verecektir.

TGD yöntemlerinde tersinirliğin sağlanması için damga ekleme sırasında kullanılan öngörü veya aradeğerleme hatalarının aynısı damga çözücüde elde edilmelidir. Bu nedenle görüntüdeki piksellerin önemli bir kısmı damgalama için kullanılmamaktadır. Damgalanmayan pikseller damga çözücüde damga ekleme adımındaki öngörü veya aradeğerleme hatalarının hesaplanmasına imkân vermektedir. Tüm piksellerin damgalanmaması sonucunda damgalama kapasitesi düşmektedir. Video damgalamada, video dizileri çerçeveler arası zamansal ilintinin değerlendirilmesi yardımıyla damgalandığında, tersinirlik üzerinde işlem yapılmakta olan çerçevenin komşu çerçeveleri üzerinden sağlandığından damgalanacak çerçevedeki tüm pikseller veri saklama için kullanılabilmekte ve buna bağlı olarak kapasite artmaktadır.



Şekil 3.1. Hall Monitor videosu için (a) uzamsal ve (b) zamansal aradeğerleme hatalarının histogramı.

Yukarıda verilen açıklamalar doğrultusunda video işaretlerinin kendilerine özgü özelliklerinden faydalanılarak bir TVD yöntemi geliştirmek, görüntü için geliştirilmiş TGD algoritmasını video çerçevelerine bağımsız bir şekilde tek tek uygulanmasına göre daha iyi performans sağlayabilir. Görüntü için geliştirilen herhangi bir damgalama yöntemi video dizisindeki her bir çerçeveye ayrı ayrı uygulandığında çerçeveler arasındaki zamansal ilinti kullanılmadığından etkin bir damgalama gerçekleştirilmemiş olur. Video işareti daha düşük entropiye sahip farklı bir işarete dönüştürülebilirse, düşük entropili işaret üzerinde yapılan bir damgalama daha iyi sonuçlar verebilir. Video çerçeveleri arasındaki zamansal ilintiyi gidermek için çerçevelerarası fark, hareket dengelenmiş öngörü veya aradeğerleme hataları kullanılabilir. Çerçevelerarası fark, videodaki hareket bilgisini dikkate almadığından entropide etkin bir iyileştirme sağlayamamaktadır. Hareket dengelenmiş öngörü hatalarının kullanıldığı damgalama yöntemlerinde, hareket bilgisinden faydalanılmasına rağmen tersinirliğin sağlanması için damga çıkartma aşamasında hareket vektörlerinin bilinmesine gerek duyulduğundan yan bilgi miktarı fazladır ve buna bağlı olarak yöntemlerin sağladığı damgalama kapasitesi düşüktür. Buna karşın, HDÇA hatalarının kullanıldığı TVD yöntemlerinde damga çözme esnasında hareket vektörlerinin bilinmesine ihtiyaç yoktur. Buna bağlı olarak, HDÇA hatalarının damgalandığı TVD algoritmaları daha yüksek damgalama kapasitelerine

çıkabilmektedirler. Bu yüzden, bu bölümde önerilen yöntemde HDÇA hatalarının damgalanması tercih edilmiştir.

Önerilen yöntemde, video dizisindeki herhangi bir çerçevenin nasıl damgalandığı Şekil 3.2.'de (üst bölüm) gösterilmiştir. F_k bir video dizisindeki *k.* çerçeveyi göstermek üzere, detayları [44]'te verilen HDÇA yöntemi kullanılarak, F_{k-1} ve F_{k+1} çerçevelerinden F_k' ile belirtilen aradeğerlenmiş çerçeve elde edilir ve

$$E_k = F_k - F_k' \tag{3.1}$$

eşitliğinden aradeğerleme hatası oluşturulur. E_k, Bölüm 2'de özetlenen YHD yöntemiyle damgalanarak damgalanmış aradeğerleme hatası E_k^w oluşturulur. E_k^w, F_k' aradeğerleme çerçevesine eklenerek F_k^w ile belirtilen damgalanmış çerçeve elde edilir:

$$F_k^w = F_k' + E_k^w \tag{3.2}$$

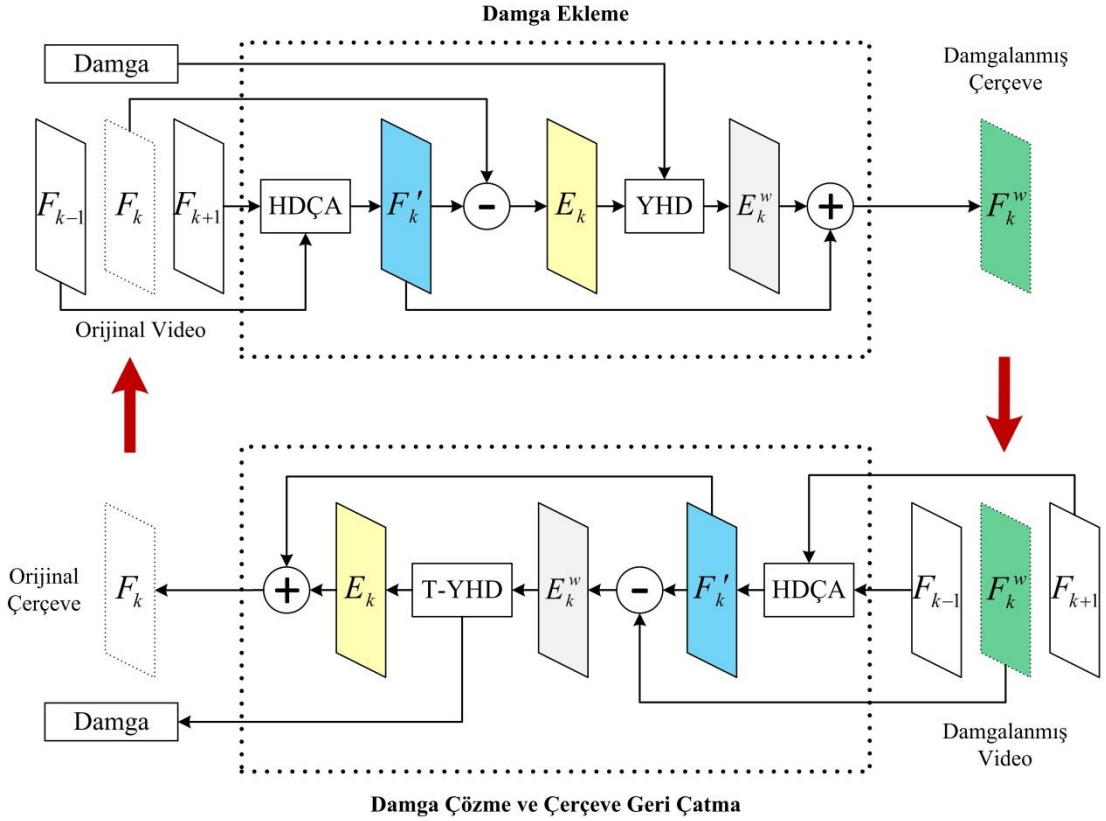
Bir çerçevenin geri elde edilmesi ve çerçevedeki damganın çıkartılmasına karşılık gelen blok diyagram Şekil 3.2.'de (alt bölüm) verilmiştir. Şekil 3.2.'de görüldüğü gibi, HDÇA kullanılarak önce F_k' elde edilir. Daha sonra,

$$E_k^w = F_k^w - F_k' \tag{3.3}$$

ilişkisinden E_k^w çıkartılır. E_k^w, Ters YHD yöntemine (T-YHD) tabi tutularak damga çıkartılır ve E_k geri elde edilir. Son olarak orijinal çerçeve,

$$F_k = E_k + F_k' \tag{3.4}$$

eşitliğinden hesaplanır.



Şekil 3.2. Önerilen yöntemde bir çerçevenin damgalanmasına, geri elde edilmesine ve çerçevedeki damganın çıkartılmasına karşılık gelen blok diyagram.

Damgalamanın tersinir olması için, damga çözme ve damga ekleme esnasında hesaplanan aradeğerlenmiş çerçeveler F'_k eşit olmalıdır. Damga eklemedeki F'_k'nın damga çözücüde aynı şekilde elde edilebilmesi için k. çerçevenin geri çatılması aşamasında F_{k-1} ve F_{k+1} çerçeveleri geri çatılmış olmalıdır. Bu amaçla, damgalama işleminde önce çift numaralı çerçeveler damgalanır. Daha sonra, damgalanmış çift çerçeveler aradeğerleme için kullanılarak tek numaralı çerçeveler damgalanır. Bir video dizisindeki çerçevelerin damgalama sırası Şekil 3.3.'te gösterilmektedir. Şekilde görüldüğü gibi, çift numaralı çerçeveler damgalanırken aradeğerleme için orijinal tek numaralı çerçeveler kullanılır. Tüm çift çerçeveler damgalandıktan sonra tek numaralı çerçevelerin aradeğerleme hataları hesaplanırken damgalanmış çift çerçeveler kullanılır. İlk ve son çerçeveler için HDÇA'da kullanılacak iki çerçeve bulunmadığından, damgalanacak çerçeve sayısı çift bir sayı ise son çerçevenin aradeğerlemesi bir önceki çerçeve; ilk çerçevenin aradeğerlemesi ise damgalı ikinci

çerçeve kabul edilir. Çerçeve sayısı tek bir sayı ise son çerçevenin aradeğerlemesi bir önceki damgalı çerçeve kabul edilerek damgalama işlemi gerçekleştirilir.

Damga çıkartma işlemi önce tek çerçeveler üzerinde gerçekleştirilir. Orijinal tek numaralı çerçeveler elde edildikten sonra çift numaralı çerçeveler damga çıkartma işlemine tabi tutulur. Bu yaklaşımla damga ekleme aşamasındaki aradeğerleme çerçevelerinin aynısının damga çözme işleminde elde edildiğine ve tersinirliğin sağlandığına dikkat ediniz.

3.3. Damganın Çerçevelere Homojen Dağıtılması ve Kapasite Parametresinin Belirlenmesi

Herhangi bir video damgalama yönteminde toplam kapasitenin çerçevelere nasıl dağıtılacağı, yöntemin görsel kalite performansını belirleyen önemli bir problemdir. Etkin bir görsel performans elde etmek için kapasite ve bozunum tüm çerçevelere eşit bir şekilde dağıtılmalıdır. Bu bölümde, YHD yönteminin video çerçevelerine uyarlanmasında eşit kapasite ve bozunum dağılımlı bir TVD algoritması geliştirilmiştir.

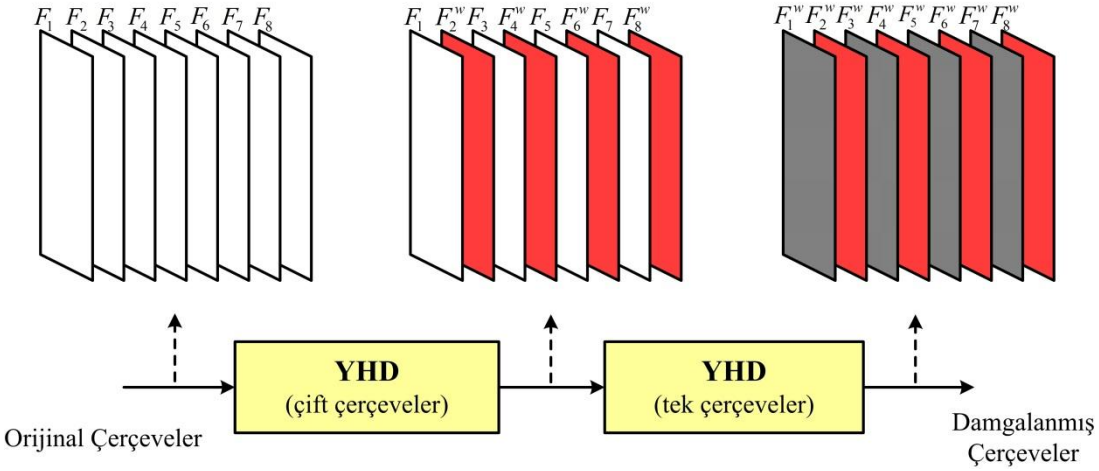
YHD, verilen bir bozunum kısıtı altında erişilebilecek en büyük kapasiteye ulaşmayı hedeflemektedir. Bu nedenle, YHD yönteminin kapasitesi bozunum kısıtının değiştirilmesi ile arttırılıp azaltılabilir. Verilen bir bozunum kısıtı altında, damgalanmış işaretin optimal marjinal dağılımının kestirilmesi problemi [41]'de çözülmüştür.

Bir video dizisine belirli miktarda bitten oluşan bir damga eklenirken toplam bit sayısı video dizisindeki çerçeve sayısına bölünerek çerçevelerin hedef kapasitesi (HK) belirlenir. Her bir çerçeve için, belirlenen HK'yı veren bozunum kısıtı hesaplanır ve damgalama işlemi gerçekleştirilir. Bir çerçevenin HK'sına ulaşıldığında bir sonraki çerçevenin damgalanmasına geçilir. Her bir çerçeveye ait bozunum kısıtı, ilgili çerçevenin yan bilgi kısmında saklanır. İlk çerçeve için bozunum kısıtının ilk değeri belirlenir. Daha sonra HK'ya ulaşılıncaya dek bozunum kısıtı arttıtılır veya

azaltılır. İlk çerçevede HK'ya ulaşıldığında elde edilen bozunum kısıtı 2. çerçeve için ilk değer kabul edilerek hesap yükü azaltılabilir.

3.4. Damga Ekleme

Damgalanacak orijinal video $V = \{F_k | k = 1,2,3, \dots, n\}$ ile gösterilsin. Damgalama işlemine her bir çerçevenin HK'sı belirlenerek başlanır. Toplam kapasite P, çerçeveler için hedef kapasite P_f ile gösterilsin. P_f, $\lceil x \rceil$ notasyonu x'den büyük veya eşit en küçük tamsayıyı belirtmek üzere, $P_f = \left\lceil \frac{P}{n} \right\rceil$ ilişkisinden hesaplanır. Çerçeveler damgalanırken HK'ya tam olarak ulaşılamayabilir, bu durumda gerçekleşen kapasitenin HK'dan büyük veya eşit olması yeterli olmaktadır. k. çerçevenin damgalanması sonucu gerçekleşen kapasite P_k^R ile gösterilsin. $P_f \leq P_k^R$ eşitsizliğini sağlayan en küçük P_k^R değeri elde edilene kadar başlangıç bozunum kısıtı değeri 0.01 aralıklarla değiştirilerek damgalama işlemi tekrar edilir. P_k^R değerini sağlayan bozunum kısıtı bir sonraki çerçevenin başlangıç bozunum kısıtı olarak saklanır. M damga bit dizisini göstermek üzere, k. çerçeveye ait aradeğerleme hataları E_k elde edildikten sonra damgalama işlemi aşağıdaki gibi yapılır.



Şekil 3.3. Tersinirliğin sağlanması için video çerçevelerinin damgalanma sırası.

E_k'nın $P_{E_k}(x)$, $x \in \{0,1,\dots,B-1\}$ ile belirtilen dağılımı hesaplanır. E_k, her biri K pikselden oluşan g adet örtüşmeyen bloklara ayrılır. E_k'daki i. blok e_i^k olmak üzere, E_k

$$E_k = \{e_i^k \mid i = 1,2,\dots,g\}$$

şeklinde ifade edilebilir. Son blok damga çözme ve çerçeve geri çatımı için gerekli yan bilgileri saklamada kullanılacağından son bloğun boyutu özel olarak hesaplanır. Son bloğun boyutu L_{son} ile gösterilsin. F_k çerçevesinin uzunluğu N olmak üzere, $N=K.(g-1) + L_{son}$ eşitliği sağlanmalıdır. Daha sonra, verilen bozunum kısıtı için E_k'ya eklenebilecek bit sayısını en büyükleyen optimum olasılık geçiş matrisleri $Q_{Y|X}$ ve $Q_{X|Y}$, [41]'deki yöntem yardımıyla hesaplanır. $Q_{Y|X}$, orijinal çerçeve ile damgalı çerçeve arasındaki geçiş olasılıklarını, diğer bir deyişle, optimum çözüm için damgalı çerçevenin dağılımının nasıl olması gerektiğini vermektedir.

Bloklara ayrılan aradeğerleme hatası çerçevesi, M damga dizisindeki bitler ile blok blok damgalama işlemine tabi tutulur. Damga ekleme işlemi, orijinal bloğa ait aradeğerleme hatalarının, $Q_{Y|X}$ olasılık geçiş matrisi yardımıyla damga bit dizisinin ters sıkıştırılmasından elde edilen diziyle değiştirilmesi esasına dayanır. Damga çözmede ise sıkıştırma algoritması ile geri çatılan damga bit dizisi elde edilir. Bu tezde, sıkıştırma ve ters sıkıştırma aritmetik kodlama kullanılarak gerçekleştirilmiştir. Aritmetik kodlayıcı sıkıştırma ve ters sıkıştırma işlemlerini gerçekleştirirken $Q_{Y|X}$ ve $Q_{X|Y}$ olasılık geçiş matrislerini kullanmaktadır. Bir bloğun damgalanması ve bloğun geri oluşturulmasında gerekli yan bilgilerin üretilmesi aşağıda detaylı olarak tartışılmıştır.

M rastgele seçilmiş 0 ve 1'lerden oluşan bir bit dizisidir. E_k çerçevesinin i. bloğu e_i^k, M_i mesajı ile damgalanarak damgalanmış blok y_i^k ve M_{i+1} damga dizisi elde edilecektir. e_i^k bloğunun histogramındaki herhangi bir nokta $e, e \in \{0,1,\dots,B-1\}$ bunun kaç kez tekrarlandığı h_e ile gösterilsin. e_i^k bloğundaki e değerleri damgalanırken, öncelikle M_i mesaj dizisi, $Q_{Y|X}$ olasılık geçiş matrisinde e'ye karşılık

gelen sütun $Q_{Y|e} = \left(P_{Y|X}(0,e), \dots, P_{Y|X}(B-1,e)\right)^{\mathrm{T}}$ kullanılarak ters sıkıştırılmış halinin uzunluğu h_e'ye eşit oluncaya kadar ters sıkıştırılma işlemine tabi tutulur. Ters sıkıştırma sonucunda elde edilen h_e uzunluğundaki $y_{i,e}^k$ alt dizisi orijinal e_i^k bloğunda e görülen yerlere yerleştirilerek damgalama işlemi gerçekleştirilmiş olur. Yukarıda anlatılan işlem e_i^k bloğundaki tüm $e \in \{0,1,\dots,B-1\}$ değerleri için tekrarlandığında e_i^k bloğunun damgalanmış hali y_i^k elde edilmiş olur. Bu işlem aracılığıyla e değeri y değerine $P_{Y|X}(y|e)$ olasılığıyla değiştirilmiştir. $Q_{Y|X}$ matrisinde dağılımının entropisi sıfıra eşit olan semboller damgalama için kullanılmaz. Ters sıkıştırma işleminin tüm blok için genelleştirilmiş hali, TS ters sıkıştırma operatörünü belirtmek üzere aşağıdaki gibi ifade edilebilir.

$$\left(b_i, y_{i,0}^k, \dots, y_{i,B-1}^k\right) = TS\left(M_i, Q_{Y|X}, h_0, \dots, h_{B-1}\right) \tag{3.5}$$

Denklem (3.5) şu anlama gelmektedir. M_i mesajının ilk b_i adet biti ters sıkıştırılarak $y_{i,e}^k$, $0 \le e \le B-1$ alt dizilerinden oluşan y_i^k dizisi elde edilir. e sembolüne karşılık gelen alt dizi anlamına gelen $y_{i,e}^k$ dizisi h_e uzunluğundadır ve $Q_{Y|e}$ yardımıyla elde edilmiştir. Damga çıkartma aşamasında eklenen bitler y_i^k dizisinin $Q_{Y|X}$'e göre sıkıştırılmasıyla geri elde edilebilir.

Damgalama işlemi gerçekleştirildikten sonra tersinirliğin sağlanması için gerekli yan bilginin üretilip damgalanmış y_i^k bloğunda, orjinal e_i^k bloğuna gore değerleri değişen sembollerin e_i^k bloğundaki orjinal değerleri ve konumları damga çözücüye iletilmelidir. Bu amaçla, damgalanmış blokta değerleri değişen sembollerin yerleri tespit edilip bu sembollerin orjinal bloktaki değerleri yan bilgi olarak saklanmalıdır. $Q_{X|Y}$ matrisinde entropisi sıfır olan dağılımlara ait semboller, bu sembollerin değerleri damgalama işlemi sonrasında değişmediğinden bu semboller için yan bilgi oluşturulmaz. Yan bilgi saklamanın en basit yolu e_i^k bloğunu sıkıştırmaktır. Bu çalışmada yan bilgi miktarını azaltan ve aşağıda açıklanan koşullu bir sıkıştırma tekniği kullanılmıştır.

Damgalanmış y_i^k bloğundaki y'ye eşit tüm sembollerin konumları IY_y, $0 \leq y \leq B-1$ dizisi ile ifade edilsin. $[\cdot]$ konum operatörünü göstermek üzere, IY_y yardımıyla $e_{i,y}^k$ alt dizisi $e_{i,y}^k = \{e_i^k[n] \mid n \in IY_y\}$ eşitliğinden oluşturulur. $e_{i,y}^k$'nin olasılık dağılımı, $Q_{X|y}$ geçiş olasılık dağılımı matrisinden elde edilebilr. Sonuç olarak, elde edilen $e_{i,y}^k$ alt dizisi $Q_{X|y}$ dağılımına göre sıkıştırılarak ilgili yan bilgi elde edilmiş olur. Bu işlem, S sıkıştırma operatörünü belirtmek üzere aşağıdaki denklem ile ifade edilebilir:

$$O(e_i^k, y) = S(e_{i,y}^k, Q_{X|y}) \quad , \quad y = 0,1,\dots,B-1$$

$\|$ sembolü art arda eklemeyi belirtmek üzere, e_i^k bloğuna ait yan bilgi

$$O(e_i^k) = O(e_i^k, 0) \,\|\, O(e_i^k, 1) \,\|\, \cdots \,\|\, O(e_i^k, B-1)$$

eşitliğinden oluşturulur. $O(e_i^k)$ yan bilgisi, M_i dizisinin ilk b_i adet bit haricinde geriye kalan bitlerin önüne eklenerek bir sonraki blok e_{i+1}^k'e ait M_{i+1} mesajı elde edilir. M_{i+1}, e_{i+1}^k'e eklenir ve e_{i+1}^k için gerekli yan bilgi M_{i+2}'nin başına eklenir. Bu işlem, (*g-1*). bloğun damgalanmasına kadar devam ettirilir.

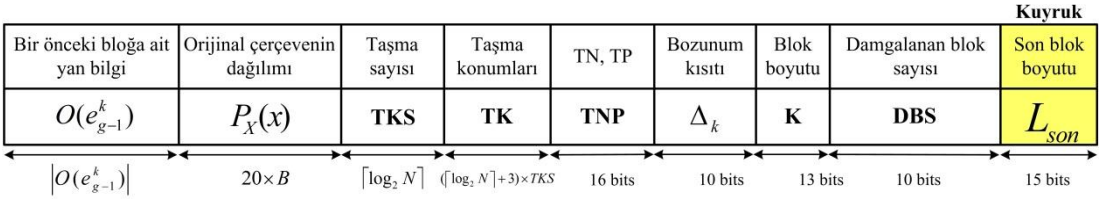
3.5. Taşma Durumlarının Ele Alınması, Yan Bilginin Oluşturulması ve Son Bloğun Boyutunun Belirlenmesi

YHD uygulanırken, damgalanmış piksel değerlerinin 8-bit görüntüler için izin verilen [0-255] aralığının dışında değer alması durumlarının dikkate alınması gereklidir. Gerçekleştirilen simülasyonlarda hareketin çok karmaşık olmadığı video dizileri için kapasitenin çok büyük değerler almadığı durumlarda taşma durumları ile karşılaşılmadığı gözlemlenmiştir. Bununla birlikte, hareketin karmaşıklaştığı ve yüksek kapasitelere çıkıldığı durumlarda az da olsa taşmalar oluşmuştur. Örneğin, hareketin oldukça yoğun olduğu 'Bus' video dizisi için 0,0870 piksel başına bit (BPP) kapasite seviyesinde çerçeve başına ortalama 14,10 adet pikselde taşma yaşanırken, 0,2802 BPP seviyesinde taşma yaşanan ortalama piksel sayısı 21,03

olarak gözlemlenmiştir. Orta düzey hareketliliğe sahip ’Foreman’ dizisinde tüm kapasite seviyeleri için çerçeve başına ortalama 0,18 adet pikselde taşma durumları ile karşılaşılmıştır. Düşük hareketliliğe sahip ‘Paris’ dizisinde ise hiçbir kapasite seviyesinde taşma durumları ile karşılaşılmamıştır. Hareketin fazla ve karmaşık olduğu çerçevelerde ara değerleme hataları yüksek değerler almakta ve bunun sonucu olarak taşma olayları yaşanmaktadır.

Görüldüğü gibi videodaki hareketliliğe bağlı olarak, karşılaşılan taşma durumları oldukça seyrek karşılaşılmasına rağmen, tersinirliğin sağlanması bakımından taşma yaşanan piksel konumları, sayısı ve taşma miktarları damga çözücüye iletilmelidir. Taşma oluşan pikselin orijinal değerinin damga çözücüde elde edilebilmesi için pikselin konumu ve taşma miktarının bilinmesi yeterli olacaktır. Buna ek olarak, bir çerçeve özellikle düşük kapasitelerde damgalanırken hedef kapasiteye ulaşmak için tüm blokların damgalanmasına gerek kalmayabilir. Hatasız bir damga çıkartma yapılabilmesi için hangi blokların damgalandığı bilgisi de damga çözücüye iletilmelidir. Bu amaçla damgalanan blok sayısı (DBS) ilgili çerçevenin yan bilgisine eklenir.

YHD yöntemi, HDÇA hatalarına uygulanmıştır. HDÇA hatalarının histogramı bir T eşik değeri tarafından kırpılmaktadır. h_e, herhangi bir HDÇA hatasının görülme sıklığını temsil etsin. $[T_N, T_P]$ değerleri T’nin seçiminin bir sonucu olmak üzere, sadece $h_e > T, e \in [T_N, T_P]$ ilişkisini sağlayan HDÇA hataları damgalamada kullanılmaktadır. $[T_N, T_P]$ aralığı, damgalanan hataların sayısı B’yi belirlemektedir. Video dizilerinin hareket karakteristiğine göre TN,TP ve B değerleri değişmektedir. Hareketli videolarda bu aralık büyürken daha dik ve dar histogramlı HDÇA hatalarına sahip hareketsiz videolarda $[T_N, T_P]$ aralığı küçülmektedir. Bu çalışmada, tüm video çerçeveleri için sabit TN,TP değerleri yerine, her bir çerçeve için T değerine bağlı olarak ayrı değişken TN,TP değerleri kullanılmıştır. Bu nedenle, her bir çerçeve için damgalamada kullanılan TN,TP ikilisinin de damga çözücüye iletilmesi gerekmektedir.



Şekil 3.4. Damgalı bir çerçevenin geri elde edilmesi için gerekli yan bilgi.

Her çerçevedeki son blok yan bilgileri saklamak için kullanılmaktadır. Bir önceki bloğa ait yan bilgi $O(e_{g-1}^{k})$, orijinal çerçevenin dağılımı $P_X(x)$, bozunum kısıtı Δ_k, blok uzunluğu *K*, damgalanan blok sayısı *DBS*, Taşma yaşanan piksel konumları (*TK*), taşma konumları sayısı (*TKS*) ve (*TN,TP*) parametreleri (*TNP*) son blokta saklanır. $O(e_{g-1}^{k})$'de bir önceki bloğun damgalanmasından sonra, damgalanmış blokta değerleri değişen ve entropisi sıfırdan farklı sembollerin orjinal değerleri tutulmaktadır. Bir çerçevesinin boyutları $W \times H$ olan bir video dizisinde bir çerçevedeki toplam piksel sayısı $N = W \times H$ olmak üzere, TKS $\lceil \log_2 N \rceil$ bit, TK ise $(\lceil \log_2 N \rceil + 3) \times TKS$ bit ile temsil edilebilmektedir. 3 bit taşma değerini saklamak için yeterli olacaktır. B elemanlı $P_X(x)$'i saklamak için genellikle $B \times 20$ bit yeterli olmaktadır. Δ_k için 10 bit, *K* için 13 bit, *TNP* için 16 bit ayrılmaktadır. *DBS* için 10 bit yeterli olmaktadır. Ayrıca, son bloğun son 15 biti, son blok boyutunu (L_{son}) saklamak için kullanılır. Bu durumda bir çerçeve için gerekli yan bilgi miktarı aşağıdaki gibi verilebilir.

$$L_{son} = \left|O(e_{g-1}^{k})\right| + B.20 + \lceil \log_2 N \rceil + (\lceil \log_2 N \rceil + 3).TKS + 10 + 13 + 16 + 10 + 15$$

Bu bilgiler ışığında, son blokta saklanan yan bilgiyi oluşturan bileşenler Şekil 3.4.'te gösterilmiştir. Son blok e_g^k, en az anlamlı bit (LSB) değiştirme yöntemi ile tüm yan bilgileri saklayacak kapasitede damgalanır. Son bloğunun orijinal LSB'leri ise önceki bloklara damga biti olarak eklenir. Son bloğun boyutu bahsedilen bütün yan bilgilerin saklanmasına imkân verecek şekilde seçilir.

3.6. Damga Çözme ve Orijinal Bloğun Geri Çatımı

Tek numaralı çerçevelere damga eklenirken, damgalı çift çerçevelerin aradeğerlemesi kullanılmıştır. Aynı aradeğerleme çerçevesini elde edebilmek için damga çıkartma işlemi öncelikle tek çerçeveler üzerinde gerçekleştirilir. Orijinal tek numaralı çerçeveler elde edildikten sonra çift numaralı çerçeveler damga çıkartma işlemine tabi tutulur.

Damga çıkartma işlemi E_k^w'nin hesaplanması ile başlar. E_k^w'ye yukarıda anlatılan damga ekleme işlemleri tersten yapılarak damga çıkartma işlemi gerçekleştirilir. Önce, y_{i+1}^k'dan elde edilen yan bilgi $O(e_i^k)$ aracılığıyla e_i^k oluşturulur. Daha sonra, y_i^k ve e_i^k bloklarından $Q_{Y|X}$ olasılık geçiş matrisi yardımıyla M_i belirlenir.

Öncelikle E_k^w'nin son 15 pikselinin en az anlamlı bitlerinden son bloğun boyutu L_{son} elde edilmelidir. Daha sonra, en son blok y_g^k'den orijinal işaretin dağılımı $P_X(x)$, bozunum kısıtı Δ_k ve blok boyutu K parametreleri elde edilir. Elde edilen bu parametrelerin yardımıyla $Q_{Y|X}$ ve $Q_{X|Y}$ optimum olasılık geçiş matrisleri hesaplanabilir.

Çöz kelimesi damga çıkartma işlemlerini belirtmek üzere sondan bir önceki bloktan başlayıp $\left(M_{g-1}, e_{g-1}^k\right) = \text{Çöz}(M_g, y_{g-1}^k)$ ile belirtilen damga çözme işlemi aşağıda verilen adımlar izlenerek gerçekleştirilir. y_{g-1}^k bloğunda y'ye eşit olan piksellerin sayısı l_y ile gösterilsin. M_g dizisinin ön kısmında (*g-1*). bloğa ait yan bilgiler bulunduğu için M_g dizisi l_y uzunluğunda bir sembol dizisi verene kadar $Q_{X|y}$ dağılımı altında ters sıkıştırma algoritmasına tabi tutulur. Bu işlem ilgili blokta tüm y=0,1,…,B-1 değerleri için tekrar edilir. Bir blok için ters sıkıştırma

$$\left(c_g, e_{g-1,0}^k, \dots, e_{g-1,B-1}^k\right) = TS(M_g, Q_{X|Y}, l_0, \dots, l_{B-1}) \tag{3.6}$$

şeklinde ifade edilebilir. Denklem (3.6) şu anlama gelmektedir, M_g dizisinin ilk c_g biti ters sıkıştırılarak $e_{g-1,y}^k, 0 \le y \le B-1$ alt dizileri elde edilir. y sembolüne

karşılık gelen $e_{g-1,y}^{k}$ alt dizisi l_y uzunluğundadır ve $Q_{X|y}$ dağılımına göre elde edilmiştir.

Daha sonra, y_{g-1}^{k} bloğundaki tüm y'ler $e_{g-1,y}^{k}$ dizisi ile değiştirilerek orijinal blok e_{g-1}^{k} oluşturulur. Elde edilen e_{g-1}^{k} bloğu ve y_{g-1}^{k} kullanılarak damga çözülür. Bu amaçla, e_{g-1}^{k} bloğundaki tüm 'e' lerin konumları IX_e dizisinde toplanır. Bu konumların y_{g-1}^{k} bloğunda karşılık gelen piksel değerleri $y_{g-1,e}^{k} = \{y_{g-1}^{k}[n] \mid n \in IX_e\}$ ile verilen bir dizide toplanır. $y_{g-1,e}^{k}$ dizisi $Q_{Y|e}$ dağılımına göre sıkıştırılarak ilgili noktalardaki damga çıkartılmış olur. Bu işlem tüm e=0,1,…,B-1 değerleri için tekrarlanarak e_{g-1}^{k} bloğundaki damga çözülür. Elde edilen damga, M_g dizisinden geriye kalan bitlerin önüne katılarak M_{g-1} dizisi oluşturulur.

Aynı şekilde, geriye kalan tüm i=g-2,…,1 blokları için $(M_i, e_i^k) = \text{Çöz}(M_{i+1},\ y_i^k)$ işlemleri tekrar edilerek ilk (g-1) orijinal blok ve M_1 mesajı geri elde edilir. M_1 dizisinden son bloğun orjinal LSB'leri alınarak yerlerine konmak suretiyle orijinal son blok geri elde edilir.

3.7. Sonuçlar

Önerilen TVD algoritmasının performansı literatürde sıklıkla kullanılan dört test video dizisi üzerinde sınanmıştır. Her bir video çerçevesi 352x288 boyutundadır. Eklenecek damga, rastgele oluşturulmuş bit dizisinden ibarettir. Damga, MATLAB™ yazılımında "rand" fonksiyonu yardımıyla oluşturulmuştur. rand fonksiyonu düzgün dağılımlı bir bit dizisi oluşturmaktadır. Bu yüzden damgadaki bitler yaklaşık olarak düzgün dağılımlıdır. Örneğin "Foreman" videosuna eklenen damganın %49,39'u '0', %50,61'i '1''lerden oluşmaktadır. Test videoları seçilirken farklı hareket ve histogram karakteristiğine sahip dizilerin tercih edilmesine özen gösterilmiştir. "Paris" videosunda düşük bir hareketlilik mevcutken, "Foreman" dizisi orta düzey bir hareketliliğe sahiptir, "Hall-Monitor" ve "Bus" videoları ise hareketin çok olduğu videolardır. Her bir test videosu için 30 çerçeve damgalama işlemine tabi tutulmuştur.

Elde edilen sonuçlar, damga ekleme kapasitesi ve görsel kalite bakımlarından [35],[36] ve [37]'deki yöntemlerle karşılaştırılmıştır. Piksel başına bit (BPP) ile verilen damgalama kapasitesi, yan bilgi hariç eklenen toplam bit sayısının toplam piksel sayısına bölümü ile elde edilmektedir. N adet çerçeveden oluşan bir video dizisi için damga ekleme kapasitesi,

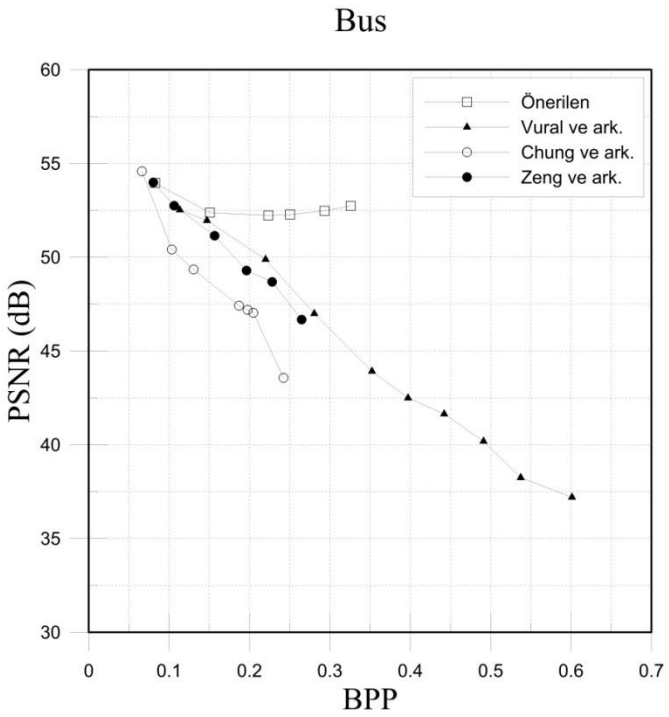
$$BPP = \frac{PL}{N \times W \times H}$$

eşitliğinden hesaplanır. PL, toplam eklenen bit sayısını, W ve H sırasıyla çerçeve genişliği ve yüksekliğini ifade etmektedir. Görsel kalite ise Tepe-İşaret Gürültü Oranı (PSNR) ve Yapısal Benzerlik İndeksi (SSIM) ölçütleri ile ölçülmektedir. PSNR iki sayısal işaret arasındaki bozunumu ölçen en yaygın geleneksel yöntem olmasına rağmen insan görme sistemine uygun olmadığı kabul edilmektedir. Bu eksikliği gidermek ve daha doğru bir ölçüm yapmak adına geliştirilmiş olan SSIM ölçütünün kullanımı son yıllarda gittikçe yaygınlaşmaktadır. Bu yüzden, bu tezde yapılan tüm karşılaştırmalar her iki ölçütü de kullanarak gerçekleştirilmiştir.

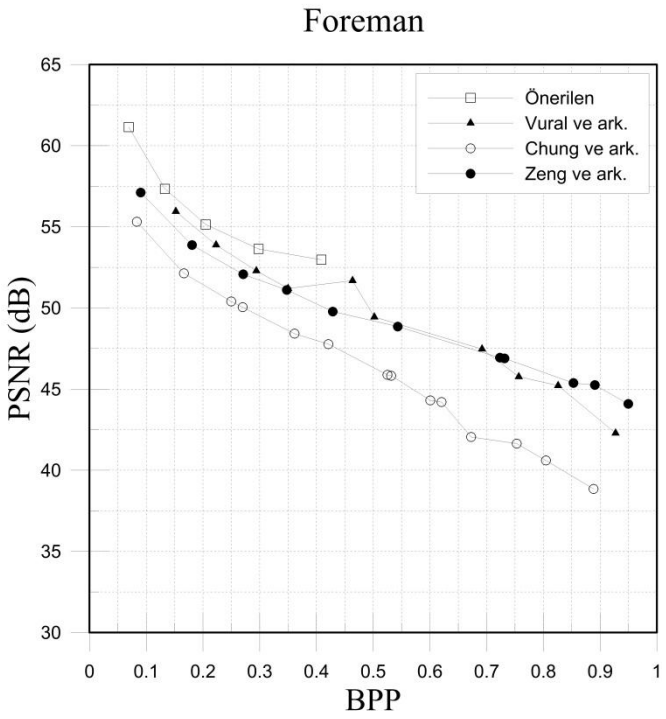
Benzetimler, Intel Core 2 Duo CPU 2.4 GHz işlemcili, 3 GB RAM'e sahip ve MS-Windows 2007 işletim sistemi yüklü bir kişisel bilgisayarda MATLAB™ 2014 geliştirme ortamında gerçekleştirilmiştir. Bir çerçevenin damgalanması için gereken süre [36]'daki yöntem kullanıldığında 0,47 dk., [35]'dekinde 0,43 dk., [37]'dekinde 1,4 dk. olarak gerçekleşirken önerilen yöntem ile bir çerçevenin damgalanması ortalama 1,37 dk. sürmektedir. Önerilen ve [37]'deki yöntemlerin diğerlerine göre daha farklı çalışma sürelerini gerektirmesi aradeğerlemeye dayalı olmalarından kaynaklanmaktadır.

Önerilen yöntemde, HDÇA hatalarının histogramında T eşik değeri ile belirlenen $[T_N, T_P]$ aralığındaki hatalar damgalamada kullanılmaktadır. T küçük seçildiğinde $[T_N, T_P]$ aralığının uzunluğu B büyümekte ve buna bağlı olarak blok uzunluğu K'nın büyük tutulması gerekmektedir. Bunun nedeni şu şekilde açıklanabilir. Küçük blok boyutlarında bloğa ilişkin piksel dağılımı tüm çerçeveye karşılık gelen piksel

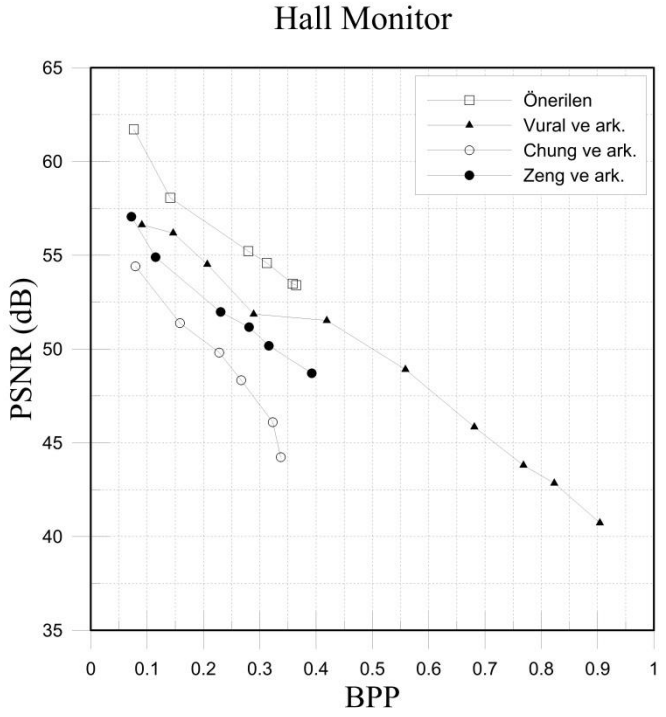
dağılımını tam olarak yansıtmayabilir. Bu nedenle, aritmetik kodlayıcı verimli bir kodlama gerçekleştiremeyebilir. Önerilen yöntemde T = 100, K = 500 alınmıştır. Hareketin daha az olduğu video dizilerinde HDÇA hatalarının histogramı daha dik ve dar olduğundan daha az sayıda HDÇA hatası damgalamada kullanılmaktadır. 'Bus' video dizisi için T = 100 için $[T_N, T_P] = [-44,45]$ ve $B = 90$ olduğu gözlemlenmiştir. Hareketin daha az olduğu 'Paris' videosunda ise $[T_N, T_P] = [-18,17]$ ve $B = 36$ olarak gerçekleşmiştir. Son bloğun boyutu, 'Paris' ve 'Hall Monitor' videoları için $L_{son} = 1200$, 'Foreman' için $L_{son} = 1600$, 'Bus' için ise $L_{son} = 3000$ olarak hesaplanmıştır. Önerilen ve karşılaştırmada kullanılan yöntemlerin kapasitenin fonksiyonu cinsinden sağladıkları görsel kalite Şekil 3.5., Şekil 3.6., Şekil 3.7. ve Şekil 3.8.'de gösterilmiştir.



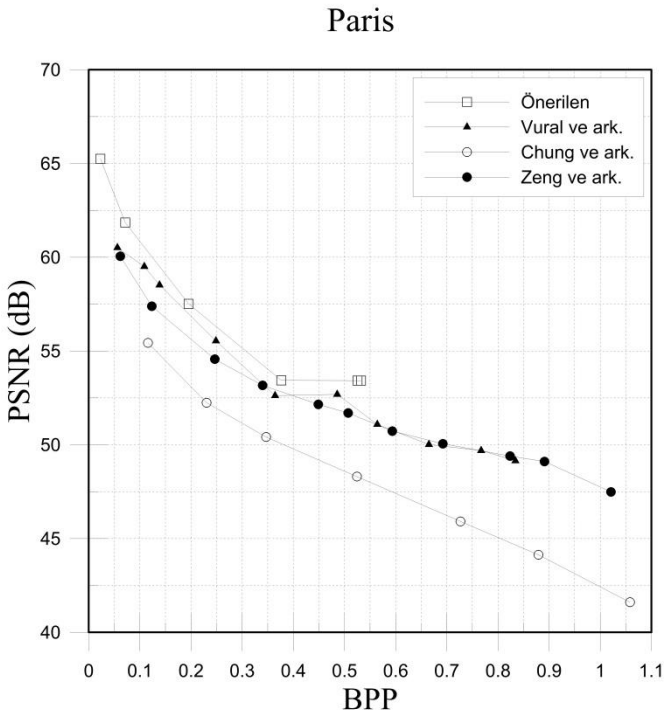
Şekil 3.5. Kapasitenin fonksiyonu olarak önerilen ve mevcut yöntemlerin Bus videosu için sağladıkları görsel kalite.



Şekil 3.6. Kapasitenin fonksiyonu olarak önerilen ve mevcut yöntemlerin Foreman videosu için sağladıkları görsel kalite.



Şekil 3.7. Kapasitenin fonksiyonu olarak önerilen ve mevcut yöntemlerin Hall-Monitor videosu için sağladıkları görsel kalite.

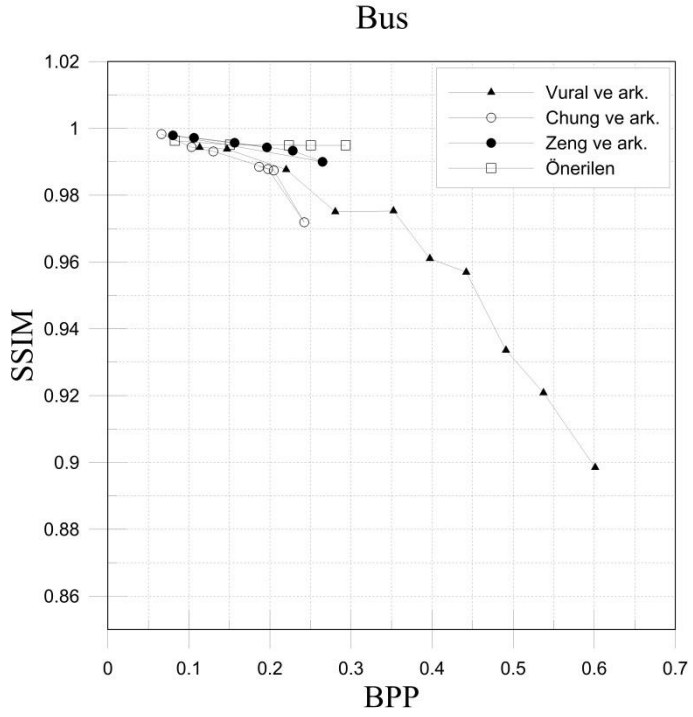


Şekil 3.8. Kapasitenin fonksiyonu olarak önerilen ve mevcut yöntemlerin Paris videosu için sağladıkları görsel kalite.

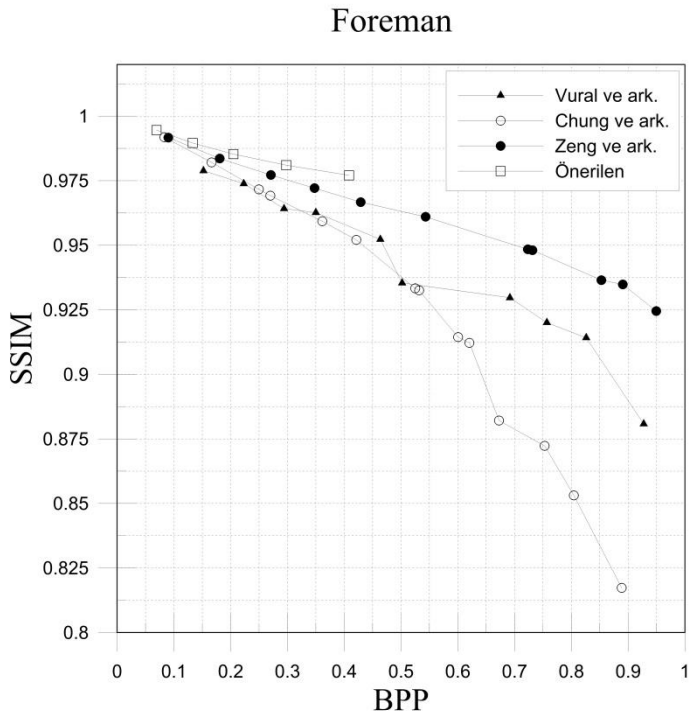
Şekillerden görüldüğü gibi, verilen bir kapasitede mevcut yöntem diğer yöntemlerden daha yüksek görsel kalite vermektedir. Örneğin, "Foreman" videosu için 0,3 BPP damgalama seviyesinde, [36], [35] ve [37]'deki çalışmaların verdiği görsel kalite sırasıyla 34,25 dB, 48,50 dB ve 50,31 dB iken, önerilen yöntem 53,16 dB görsel kalite vermektedir. Hareketin fazla olduğu videolarda daha yüksek aradeğerleme veya öngörü hataları elde edileceğinden bu videolar için tüm yöntemler düşük performansa sahiptir. Daha düşük harekete sahip video dizilerinde ise damgalama yöntemleri daha yüksek kapasite ve PSNR değerleri elde edebilmektedirler. Örneğin, hareketin fazla olduğu "bus " videosu için önerilen yöntem en fazla 0,31 BPP düzeyinde kapasite sağlayabilmekteyken, daha düşük hareketliliğe sahip "paris" videosu için 0,60 BPP seviyesine kadar kapasite sağlayabilmektedir. Ayrıca, önerilen yöntem, 0,20 BPP kapasiteye "bus" dizisinde 52,16 dB PSNR değerinde ulaşırken, "Paris" dizisinde 57,31 dB PSNR değerinde ulaşmaktadır. Bunun nedeni aradeğerleme hatalarının hareketin az olduğu video dizilerinde daha düşük olmasıdır. Bununla beraber önerilen yöntemin, videonun

hareket karakteristiğinden bağımsız bir biçimde mevcut yöntemlerden daha iyi kapasite ve görsel kalite performansına sahip olduğu görülmektedir. Örneğin benzetimlerde kullanılan 4 test videosu için, 0,20 BPP kapasite seviyesinde önerilen yöntem [37]'deki çalışmadan ortalama 2,59 dB, [35]'tekinden 3,75 dB ve [36]'dakinden 21,19 dB daha iyi sonuçlar vermektedir.

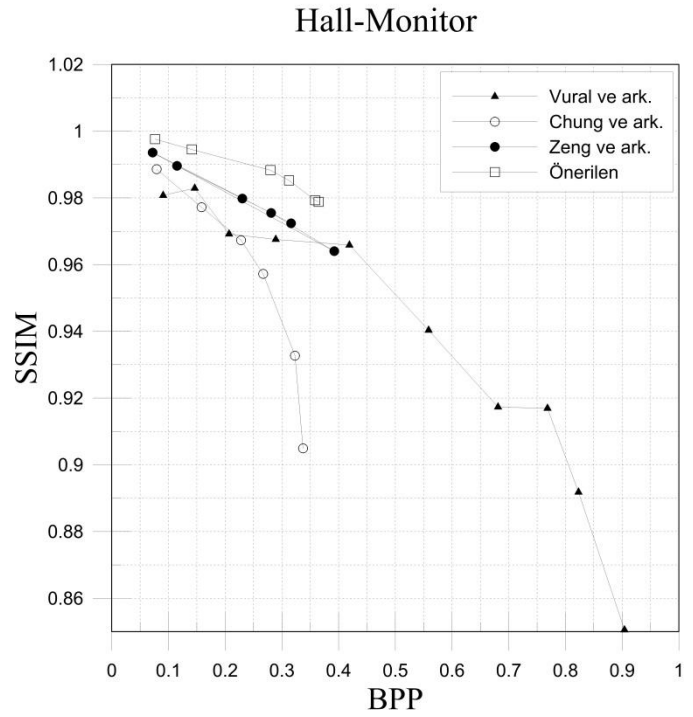
PSNR insan görme sistemi ile uyumlu değildir. Buna karşın, SSIM ölçütü insan görme sistemi dikkate alınarak geliştirilmiştir. Yöntemler SSIM ölçütüne göre de karşılaştırılmıştır. Ancak, ölçütlerin farklı özellikleri dikkate almasından dolayı bir ölçüt bakımından iyi olan bir yöntem diğer ölçüt bakımından iyi olmayabilir. TD yöntemleri üzerinde yapılan araştırmaların çoğunda yapılan iyileştirmeler PSNR ölçütü dikkate alınarak geliştirilmiştir. Video damgalama yöntemlerinde önceki TD algoritmalarından faydalanıldığı için sonuçların SSIM ölçütü ile değerlendirilmesinde aynı iyileştirmeler beklenmemelidir. Bununla birlikte, önerilen yöntemin SSIM cinsinden oluşturulan kapasite-bozunum performansının diğer yöntemlere göre üstün olduğu Şekil 3.9., Şekil 3.10., Şekil 3.11. ve Şekil 3.12.'den görülebilir.



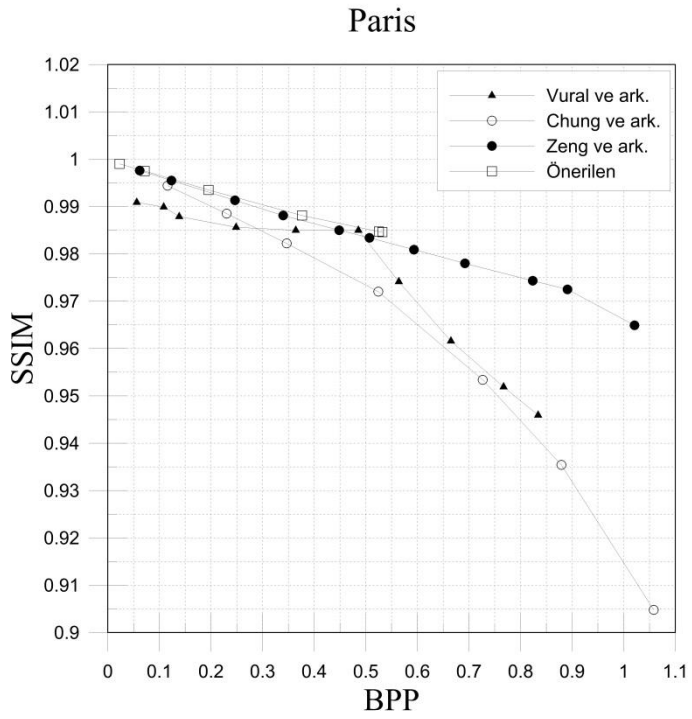
Şekil 3.9. Bus videosu için SSIM cinsinden kapasite-bozunum performansı.



Şekil 3.10. Foreman videosu için SSIM cinsinden kapasite-bozunum performansı.



Şekil 3.11. Hall-Monitor videosu için SSIM cinsinden kapasite-bozunum performansı.

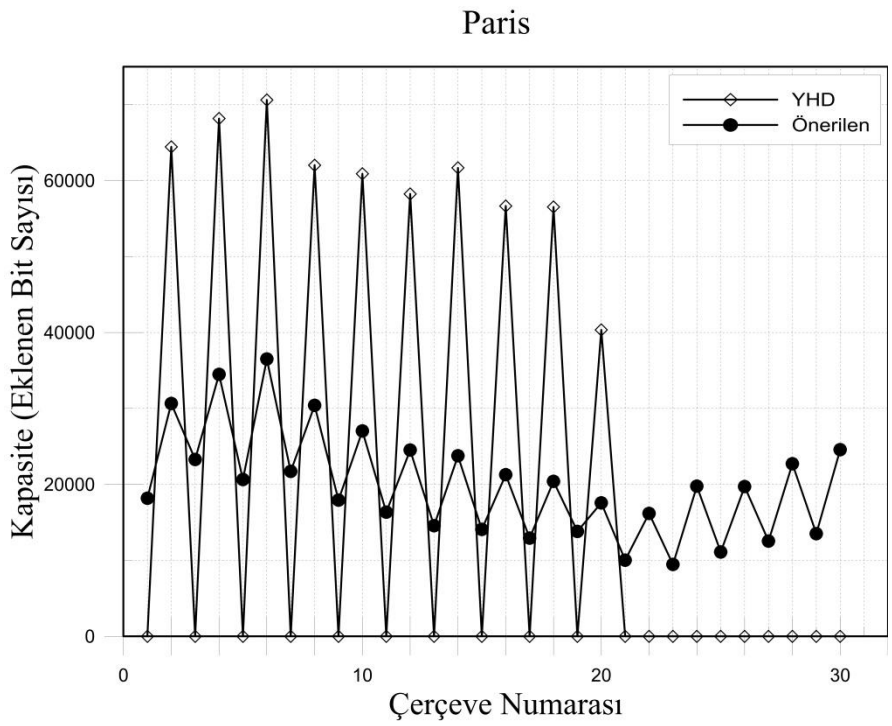


Şekil 3.12. Paris videosu için SSIM cinsinden kapasite-bozunum performansı.

Önerilen yöntemde, bir video dizisi verilen bir hedef kapasite için damgalanırken kapasite ve oluşacak bozunumun tüm çerçevelere homojen bir biçimde dağıtılmasını sağlayan özgün bir video damgalama yaklaşımı geliştirilmiştir. Önerilen yaklaşımın sağladığı görsel kalitenin önemini kavramak amacıyla toplam kapasite çerçevelere eşit olarak dağıtılmadan her bir çerçeve kendi maksimum kapasitesinde YHD yöntemiyle damgalanmış ve sonuçlar önerilen yaklaşımla karşılaştırılmıştır. 30 çerçeveden oluşan Paris videosuna 600.000 bit damga eklenerek benzetim sonuçları elde edilmiştir. Önerilen yaklaşımın kapasite ve bozunum üzerindeki etkileri Şekil 3.13., Şekil 3.14. ve Tablo 3.1.'de gösterilmiştir. İlgili şekiller ve Tablodan önerilen yöntemle bozunumdaki dalgalanmaların azaltıldığı ve kapasitenin tüm çerçevelere homojen bir biçimde dağıtıldığı görülmektedir. Örneğin, önerilen yöntemle kapasite 30 çerçeveye homojen bir biçimde dağıtıldığında ortalama PSNR 57,18 dB olarak ölçülürken diğer yaklaşım kullanıldığında kapasite 10 çerçevede toplanmış ve ortalama PSNR 52,76 dB seviyesinde ölçülmüştür. Önerilen yöntemdeki küçük dalgalanmalar, tek numaralı çerçevelerin damgalanmasında damgalanmış çift numaralı çerçevelerin aradeğerleme hatalarının kullanılmasından kaynaklanmaktadır. Tek numaralı çerçevelerin aradeğerleme hatalarının bulunmasında damgalanmış yani

bozunuma uğramış çerçevelerin kullanılması kapasite ve bozunumda çift numaralı çerçevelere oranla kayıplara neden olmaktadır. Örneğin 7 numaralı çerçeveye 21720 adet bit 56,83 dB görsel kalitede eklenebilirken, 8 numaralı çerçeve 55,42 dB görsel kalitede 30418 bit saklayabilmektedir. Bununla birlikte, önerilen yaklaşımın sağladığı kapasitenin çerçeveler üzerinde eşit dağılımlı olamamasının nedeni her bir çerçevede farklı yan bilgi miktarlarının oluşmasıdır. Şekil 3.13., Şekil 3.14. ve Tablo 3.1.'deki kapasiteler yan bilgi hariç eklenen damga biti sayısını belirtmektedir.

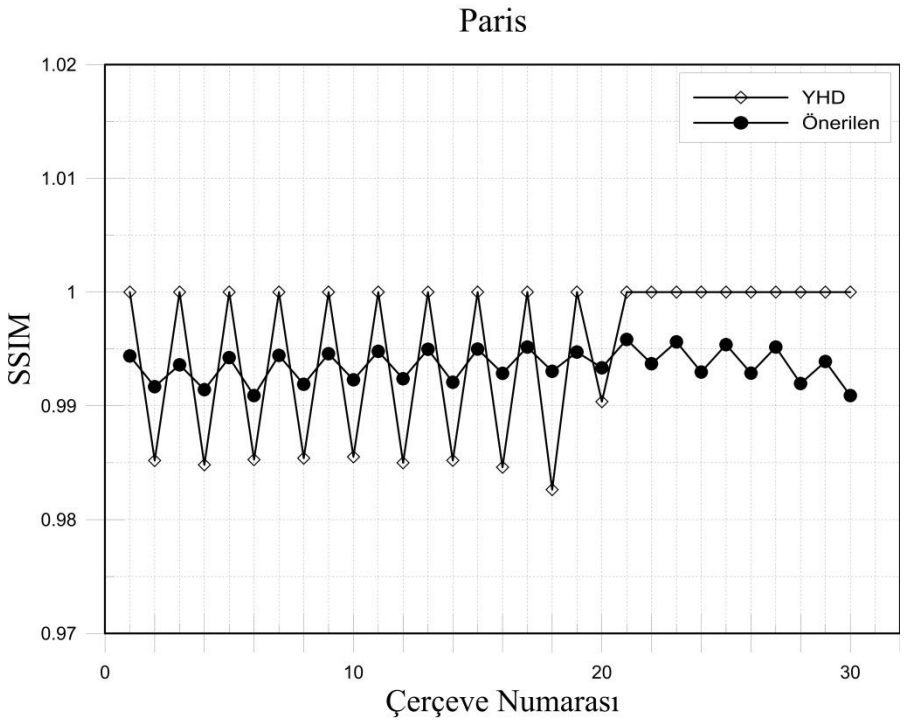
Önerilen yaklaşım, karşılaştırılan yöntemlere göre daha iyi kapasite-görsel kalite karakteristiğine sahip olmasına rağmen diğer yöntemlerdeki yüksek kapasitelere ulaşamamaktadır. Bu eksiklik, blok uzunluğunun sınır seviyelere kadar düşürülmesi veya çok seviyeli damgalama ile çözülebilir.



Şekil 3.13. Paris videosu için her iki yaklaşımda elde edilen çerçeve başına kapasite performansı.

Tablo 3.1. Paris videosu için PSNR, SSIM ve Kapasite değerleri.

Çerçeve No	PSNR		SSIM		KAPASİTE (Bit Sayısı)	
	Önerilen	YHD	Önerilen	YHD	Önerilen	YHD
1	57,3933	∞	0,9944	1	18184	0
2	55,3362	52,5628	0,9917	0,9852	30666	64478
3	56,4258	∞	0,9936	1	23298	0
4	54,9017	52,4030	0,9914	0,9848	34497	68211
5	57,0158	∞	0,9942	1	20645	0
6	54,7278	52,4420	0,9909	0,9853	36518	70662
7	56,8290	∞	0,9944	1	21720	0
8	55,4154	52,5799	0,9919	0,9854	30418	62072
9	57,4278	∞	0,9946	1	17939	0
10	55,8564	52,6802	0,9923	0,9855	27051	60934
11	57,8215	∞	0,9948	1	16355	0
12	56,1669	52,7724	0,9924	0,9850	24529	58289
13	58,0995	∞	0,9950	1	14568	0
14	56,2662	52,9027	0,9921	0,9852	23770	61713
15	58,2730	∞	0,9950	1	14085	0
16	56,7014	52,8385	0,9929	0,9846	21285	56713
17	58,5902	∞	0,9952	1	12948	0
18	56,8570	52,2110	0,9930	0,9826	20413	56560
19	58,3449	∞	0,9947	1	13829	0
20	57,3519	54,2825	0,9934	0,9904	17584	40368
21	59,4136	∞	0,9958	1	10041	0
22	57,6521	∞	0,9937	1	16197	0
23	59,4038	∞	0,9956	1	9475	0
24	56,9269	∞	0,9930	1	19765	0
25	58,9603	∞	0,9954	1	11109	0
26	56,9569	∞	0,9929	1	19723	0
27	58,6261	∞	0,9952	1	12548	0
28	56,4420	∞	0,9920	1	22729	0
29	59,1311	∞	0,9939	1	13525	0
30	56,2356	∞	0,9909	1	24586	0



Şekil 3.14. Paris videosu için her iki yaklaşım için çerçeve başına SSIM cinsinden bozunum performansı.

BÖLÜM 4. ŞİFRELİ VİDEOLAR İÇİN YİNELEMELİ HİSTOGRAM DEĞİŞTİRME TABANLI TERSİNİR VİDEO DAMGALAMA YÖNTEMİ

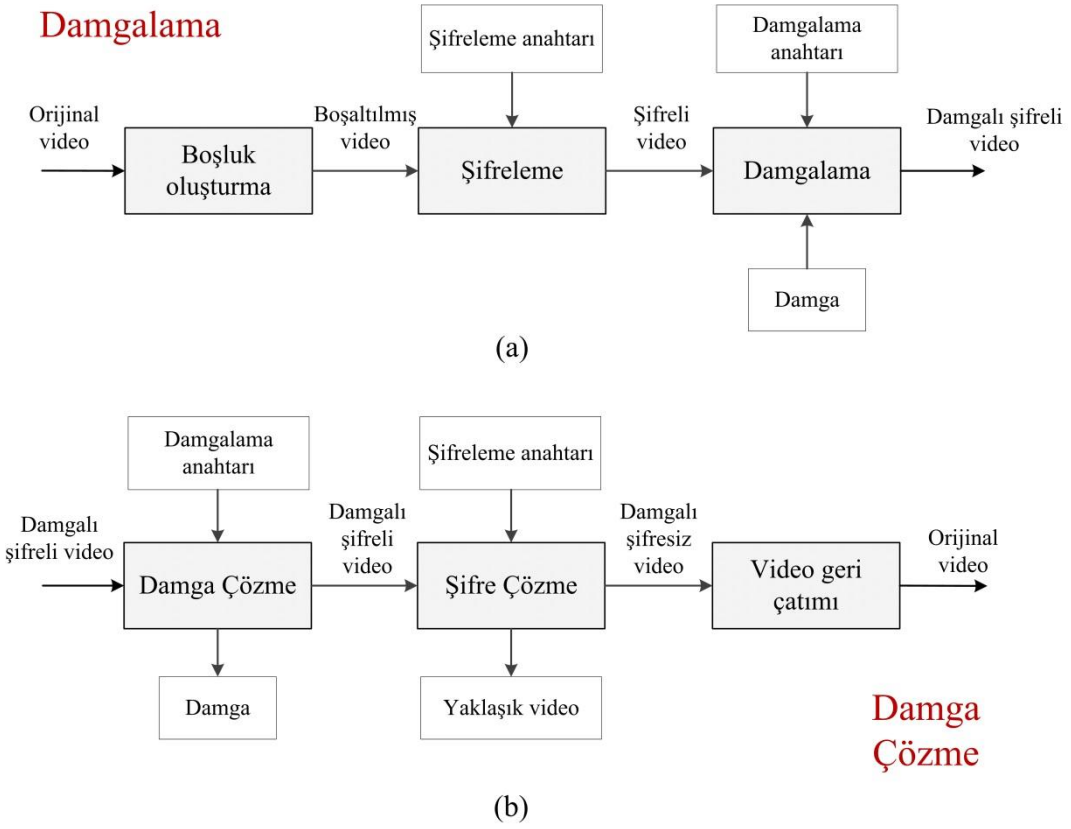
4.1. Giriş

TD konusunda araştırmalar şifresiz veriler üzerinde yoğunlaşmıştır. Bulut teknolojisindeki gelişmeler sonucunda istemci-sunucu tabanlı uygulamaların yaygınlaşmasıyla şifreli verilerin damgalanması ihtiyacı doğmuştur. Şifreli verilerin tersinir damgalanması, şifresiz verilerin damgalanmasında olduğu gibi iki aşamada gerçekleştirilmektedir; (i) orijinal veri üzerinde eklenecek damga için boşluk oluşturulur, (ii) boşluğa damga tersinir bir biçimde eklenir.

Bu bölümde, video işaretleri için ŞÖBO tabanlı *ayrışabilir* bir tersinir şifreli video damgalama (TŞVD) yöntemi açıklanmıştır. Bölüm 4.2.'de önerilen yöntemin damga ekleme adımlarının detayları verilmiştir. Ayrıca, alıcı taraftaki olası kullanım durumu senaryoları ile birlikte damga çıkartma ve video geri çatma adımları açıklanmıştır. Önerilen yöntem ile elde edilen kapasite ve görsel kalite sonuçları Bölüm 4.3.'te tartışılmıştır.

Yöntemde, video çerçeveleri arasındaki ilintiden faydalanabilmek için boşluk oluşturmada HDÇA hataları kullanılmıştır. Yöntemin damgalama kısmı, çerçeveler üzerinde boşluk oluşturma, şifreleme ve damgalama adımlarından oluşmaktadır. Damga çözücü ise damga çıkartma, şifre çözme ve geri çatma adımlarından oluşur. Yöntemin blok diyagramı Şekil 4.1.'de verilmiştir.

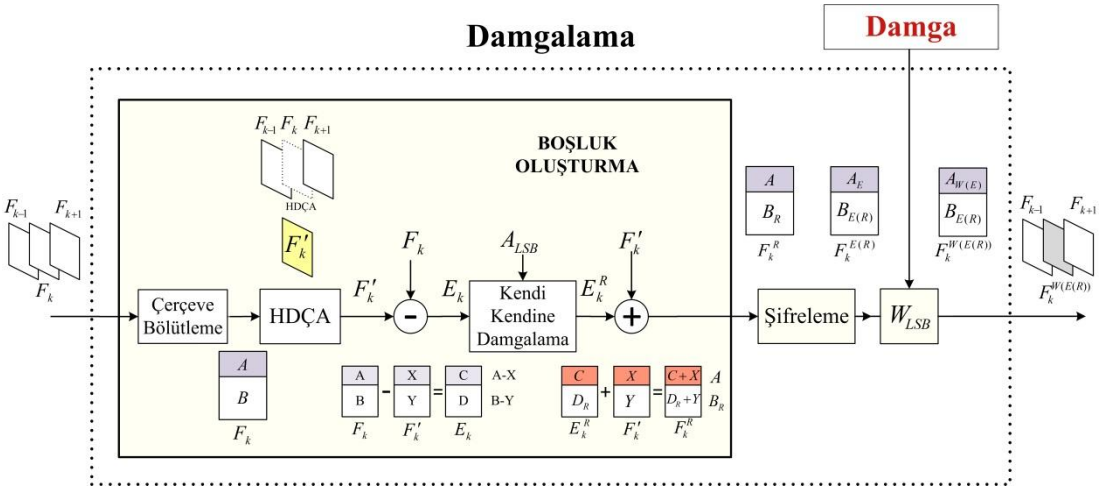
Damga çıkartımı ve çerçeve geri çatımında olası üç durum mevcuttur: (i) sadece şifreleme anahtarının bilinmesi, (ii) Sadece damgalama anahtarının bilinmesi (iii) Şifreleme ve damgalama anahtarının bilinmesi. Ayrışabilir bir TŞD yönteminde,



Şekil 4.1. Önerilen yönteme karşılık gelen blok diyagram. (a) Damga ekleme, (b) Damga çözme.

boşluk oluşturulmuş ve şifrelenmiş video, içerik sağlayıcı tarafından üçüncü parti uygulamalar için dağıtıma hazırdır. Herhangi bir içerik sağlayıcı kendi özel videosunun içeriğinin güvenliğinden emin bir şekilde sadece damgalama anahtarını vererek başka bir servis sağlayıcıya rahatlıkla gönderebilir. Video içeriğinin erişimine izin verilmek istendiğinde ise şifreleme anahtarının gönderilmesi yeterli olmaktadır. Damgalı ve şifreli video, sadece şifreleme anahtarına sahip bir alıcı tarafından ters şifrelenip damgaya erişemeden rahatlıkla kullanılabilir. Bu durumda, orijinal videonun bozunumlu bir versiyonu elde edilmiş olur.

Algoritmanın önemli bir bölümünü oluşturan ve performansını büyük ölçüde etkileyen boşluk oluşturma işlemi, HDÇA ve KKTD adımlarından oluşmaktadır. Tersinirliğin sağlanması için orijinal çerçeve ve HDÇA hataları çerçevesi, tüm çerçeveler için aynı olacak şekilde örtüşmeyen iki bölgeye (üst ve alt bölgeler) ayrılır. Daha sonra, orijinal çerçevedeki üst bölgenin LSB'leri HDÇA hataları



Şekil 4.2. Bir çerçeve için önerilen yöntemin damga ekleme kısmına karşılık gelen detaylı blok diyagram.

çerçevesindeki alt bölgeye uygun bir TD algoritmasıyla saklanır. Bu tezde, KKTD amacıyla oldukça etkin bir yöntem olan YHD tercih edilmiştir.

4.2. Önerilen Yöntem

Bir çerçeve için önerilen yöntemin damga ekleme kısmına karşılık gelen detaylı blok diyagram Şekil 4.2.'de verilmiştir. Damga ekleme boşluk oluşturma, şifreleme ve damgalama olarak belirtilen üç adımdan oluşmaktadır. Kolay anlaşılabilir olunması adına bu bölümde kullanılan bazı önemli notasyonlar Tablo 4.1.'de verilmiştir. Damga eklemenin nasıl yapıldığı aşağıda detaylı bir şekilde açıklanmıştır.

4.2.1. Damga ekleme

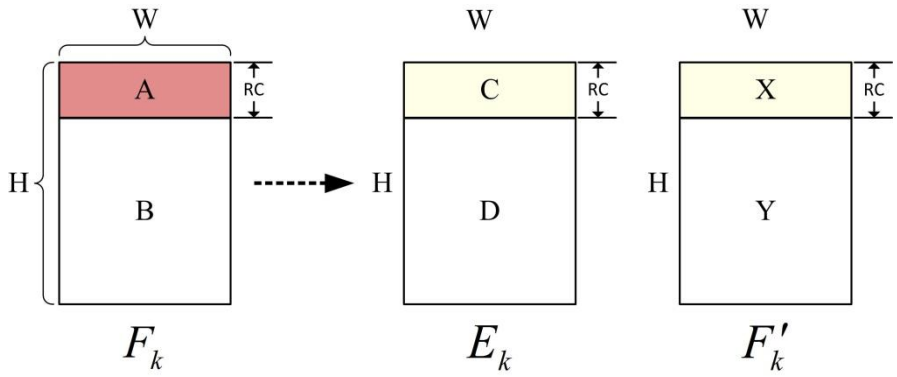
4.2.1.1. Çerçeve bölütleme ve boşluk oluşturma

Damgalanacak video üzerinde şifreleme öncesi boşluk oluşturma için öncelikle video çerçevelerinin kapasiteye göre bölütlenmesi gerekmektedir. F_k bir video dizisindeki k. çerçeveyi temsil etsin. F'_k ve E_k karşılık gelen hareket dengelenmiş aradeğerleme ve hata çerçevelerini belirtmek üzere, çerçeve bölütleme işlemi Şekil 4.3.'te verilmiştir. F_k önceden belirlenen damga kapasitesi miktarına göre A ve B olarak belirtilen iki bölgeye ayrılır. Bölütleme işlemi bir çerçeveye eklenecek bit miktarına

Tablo 4.1. Bazı önemli kısaltmalar

	Damga Ekleme
F_k	k. çerçeve
F'_k	k. HDÇA çerçevesi
E_k	k. çerçeveye ait HDÇA hata çerçevesi
F_k^R	k. boşluk oluşturulmuş çerçeve
E_k^R	k. boşluk oluşturma işlemi sonrası hata çerçevesi
$F_k^{E(R)}$	k. boşluk oluşturulmuş ve şifrelenmiş çerçeve
$F_k^{W(E(R))}$	k. boşluk oluşturulmuş, şifrelenmiş ve damgalanmış çerçeve
	Damga Çıkartma ve Video Geri Çatma
$F_k^{L(R)}$	k. damga çıkartılmış ve şifre çözülmüş çerçeve (F_k^R'den farkı: A bölgesindeki LSB'ler hatalı)
$E_k^{L(R)}$	k. damga çıkartılmış ve şifre çözülmüş HDÇA hata çerçevesi
E_k^L	k. ters boşluk oluşturma sonrası HDÇA hata çerçevesi
F_k^L	k. ters boşluk oluşturma sonrası çerçeve
F_k	k. çerçeve

göre A bölgesinin boyutunun belirlenmesi ile gerçekleştirilir. Herhangi bir video dizisi için toplam kapasite video dizisindeki çerçeve sayısına bölünerek bir çerçevenin hedef kapasitesi belirlenir. Hedef kapasiteye imkan veren A bölgesinin en küçük boyutu hedef kapasitenin çerçeve genişliğine bölünmesiyle elde edilir. $W \times H$ çerçeve boyutlarına sahip bir video dizisi için, toplam kapasite P, çerçeve sayısı FC, bir çerçeveye ait hedef kapasite P_f ve A bölgesindeki satır sayısı RC ile gösterilsin. Hedef kapasite $P_f = \left\lceil \frac{P}{FC} \right\rceil$ eşitliğinden, A bölgesinin boyutu ise $RC = \left\lceil \frac{P_f}{W} \right\rceil$ ilişkisinden hesaplanır. F_k için gerçekleştirilen bölütleme işleminin aynısı uygulanarak E_k, C ve D, F'_k ise X ve Y bölgelerine ayrılır. A, C ve X bölgelerinin aynı boyutta olduğuna dikkat ediniz.

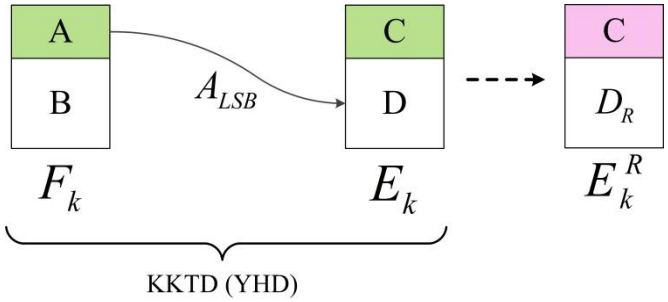


Şekil 4.3. Video dizisinde damgalanacak çerçeve ile karşılık gelen HDÇA hata çerçevelerinin bölütlenmesi.

Damgalama işlemi için gerekli boşluk video çerçeveleri arasındaki ilintiden yararlanılmasına imkân veren HDÇA hataları kullanılarak oluşturulmuştur. Detayları [44]'te verilen HDÇA yöntemi kullanılarak, orijinal F_{k-1} ve F_{k+1} çerçevelerinden F'_k ile belirtilen aradeğerlenmiş çerçeve elde edilir ve daha sonra $E_k = F_k - F'_k$ eşitliğinden HDÇA hataları hesaplanır.

F_k çerçevesinin A bölgesindeki piksellerinin LSB'leri (A_{LSB}) KKTD amacıyla YHD yöntemi kullanılarak E_k çerçevesinin D bölgesine gömülür. Böylelikle A bölgesinin LSB'leri damgalama işlemi için boşaltılmış olur. E_k çerçevesindeki D bölgesine A_{LSB} eklendikten sonra boşluk oluşturma için derğştirilmiş D bölgesi, DR ve boşluk oluşturma işlemi sonrası değiştirilmiş hata çerçevesi E_k^R elde edilmiş olur. Bu işlem Şekil 4.4.'te grafiksel olarak gösterilmiştir. Boşluk oluşturulmuş k. çerçeve $F_k^R = E_k^R + F'_k$ eşitliğinden elde edilir.

Önerilen yöntemde, damga çözücüde damganın çıkartılabilmesi için A bölgesinin boyutunun ve hedef kapasitenin bilinmesi gereklidir. A bölgesinin genişliği çerçeve genişliği ile aynı olduğundan A bölgesindeki satır sayısının (RC) damga çözücüye iletilmesi yeterlidir. RC için 10 bit ve hedef kapasite için 20 bit ayrılır ve 30 bitlik bu damgalama anahtarı Şekil 4.5.'te gösterildiği gibi damganın önüne eklenerek saklanacak veri oluşturulur. A bölgesinin LSB'lerinin tamamı D bölgesinde saklanabilmelidir. YHD tabanlı boşluk oluşturmada çerçeve bozunum kısıtı, kapasiteyi belirlemektedir. Algoritmada, her bir çerçeve için bozunum kısıtı değiştirilerek hedef kapasiteye ulaşılmaya çalışılır. Herhangi bir çerçevede bozunum



Şekil 4.4. Boşluk oluşturma işleminde KKTD sonrası hata çerçevesi.

kısıtının tüm değerleri için A bölgesinin LSB'leri D bölgesinde saklanamıyorsa damgalama yapılmaksızın "verilen damga bu videoya eklenemez" uyarısı verilir.

Boşluk oluşturma işlemi Şekil 4.6.'da gösterildiği sırada tüm çerçeveler için tekrarlanarak boşluk oluşturulmuş video dizisi elde edilir. Tersinirliğin sağlanması için önce çift numaralı çerçevelerde, daha sonra boşluk oluşturulmuş çift numaralı çerçeveler kullanılarak tek numaralı çerçevelerde boşluklar oluşturulur. Çift numaralı çerçevelerde boşluk oluşturulurken aradeğerleme için orijinal tek numaralı çerçeveler kullanılır. Tüm çift çerçevelerde boşluk oluşturulduktan sonra, tek numaralı çerçevelerin aradeğerleme hataları hesaplanırken boşluk oluşturulmuş çift çerçeveler kullanılır. İlk ve son çerçeveler için HDÇA'da kullanılacak iki çerçeve bulunmadığı için, damgalanacak çerçeve sayısı çift bir sayıysa son çerçevenin aradeğerlemesi bir önceki çerçeve, ilk çerçevenin aradeğerlemesi ise boşluk oluşturulmuş ikinci çerçeve kabul edilir. Çerçeve sayısı tek bir sayıysa son çerçevenin aradeğerlemesi bir önceki boşluk oluşturulmuş çerçeve olarak varsayılır.

4.2.1.2. Şifreleme

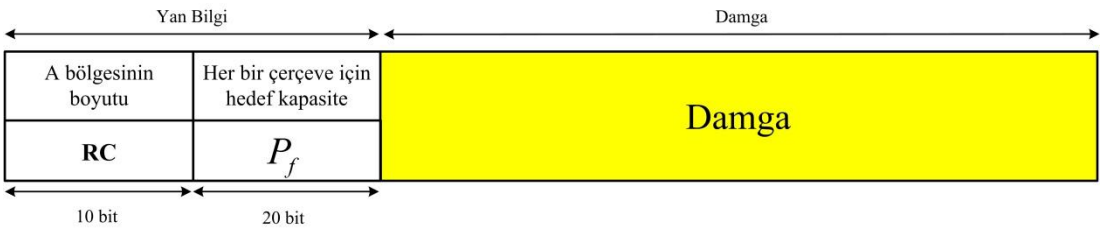
KKTD sonrasında yeniden düzenlenen ve 8 bit ile temsil edilen F_k^R çerçevesindeki herhangi bir pikselin grilik seviyesinin ikilik sayma sistemi temsilinde karşılık gelen bitleri $F_{k,i,j}^R(b) = \left\lfloor \frac{F_{k,i,j}^R}{2^b} \right\rfloor \ mod\ 2, \quad b \in \{0,1,\ldots,7\}$ eşitliğinden hesaplanabilir. Standart bir şifreleme algoritması tarafından şifreleme anahtarı yardımıyla üretilen ve her bir elemanı 8 bit uzunluğunda olan kod matrisi r ile gösterilsin. Şifrelenmiş çerçevede karşılık gelen bitler

$$F_{k,i,j}^{E(R)}(b) = F_{k,i,j}^{R}(b) \oplus r_{i,j}(b) \quad (4.1)$$

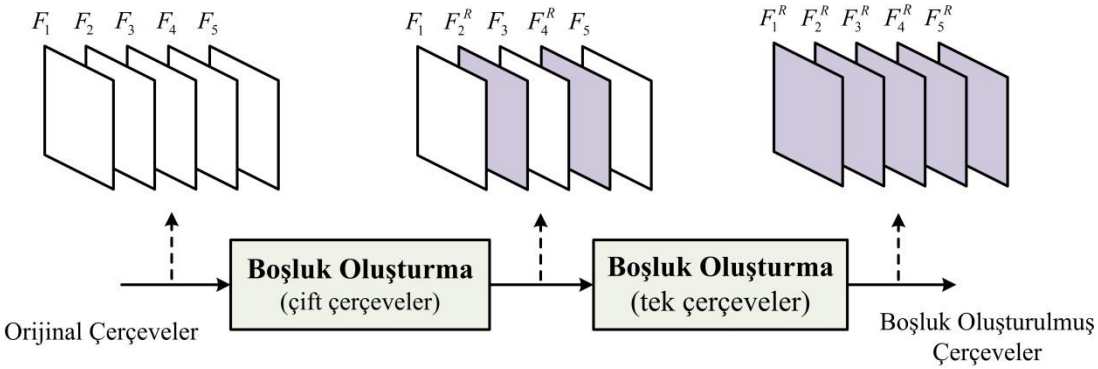
denklemi ile hesaplanır. Denklem (4.1)'de $\oplus$ operatörü, XOR işlemini göstermektedir. $r_{i,j}$ ise şifreleme algoritması tarafından çerçevedeki her bir piksel için özel olarak üretilen 8 bit uzunluğundaki kodu ifade etmektedir. Şifreleme sonrası, damgalayıcı taraf veya üçüncü parti bir servis sağlayıcı şifreleme anahtarı olmaksızın video içeriğine erişemeyecektir. Açıklanan işlemler sonunda bir yandan içerik sağlayıcının mahremiyeti korunurken diğer yandan video, damgalamaya müsait bir hale getirilmiş olur.

4.2.1.3. Damgalama

Boşluk oluşturulmuş ve şifreli $F_k^{E(R)}$ çerçevesindeki şifrelenmiş A bölgesinin LSB'leri damga bit dizisi ile değiştirilerek damgalama işlemi gerçekleştirilir ve $F_k^{W(E(R))}$ ile belirtilen damgalı ve şifreli k. çerçeve elde edilir. Bu işlem tüm çerçeveler için gerçekleştirilerek damgalı ve şifreli video dizisi elde edilmiş olur. A bölgesinin orijinal LSB'leri, boşluk oluşturma aşamasında KKTD ile hata çerçevesindeki D bölgesine saklanmıştır. Böylelikle, damga çözücüde orijinal LSB'lerin kayıpsız geri elde edilmesi garanti altına alınarak tersinirlik sağlanmış olmaktadır. Şifreleme ve damgalama adımları tersinirliğin sağlanmasına engel olan hareket dengelemeli aradeğerleme içermediğinden, damga ekleme işlemi birinci çerçeveden başlayarak çerçeve sırasına göre tüm çerçevelere uygulanarak tamamlanır.



Şekil 4.5. Damgalama anahtarı dahil saklanacak veriyi oluşturan bileşenler.



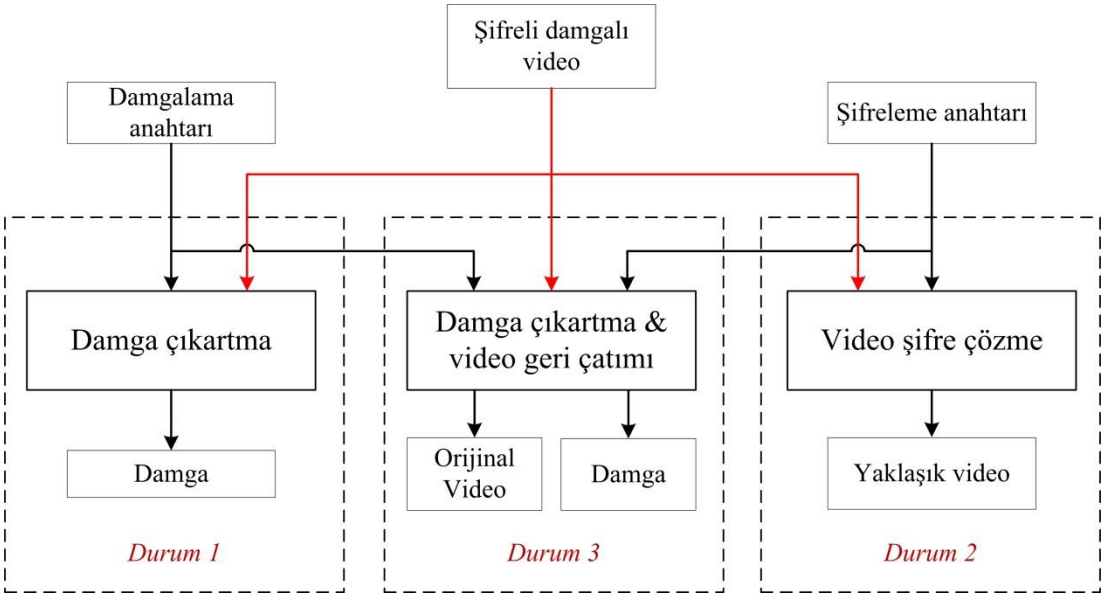
Şekil 4.6. Damgalamanın tersinir olması amacıyla boşluk oluşturma işleminde çerçevelerin işlenme sırası.

4.2.2. Damga çıkartma ve video geri çatma

Damga çözücüde, damgalanmış şifreli video üzerinde yapılabilecek üç işlem vardır. Sadece şifreleme anahtarının bilinmesi halinde, şifre çözme gerçekleştirilebilir ancak orijinal çerçeve hatasız olarak geri çatılamaz onun yerine bozunumlu bir versiyonu elde edilebilir. Sadece damgalama anahtarının mevcut olması durumunda, şifre çözme gerçekleştirilemez fakat damga çıkartılabilir. Son olarak, iki anahtarında mevcut olması durumunda damga çıkartılabilir ve orijinal video geri çatılabilir. Üç olası senaryo için damga çözücüde gerçekleştirilen işlemler blok diyagram olarak Şekil 4.7.'de gösterilmiştir. Damga çıkartımı için sadece damgalama anahtarının bilinmesinin yeterli olduğuna dikkat ediniz. Şekil 4.8.'de bir çerçeve için damga çıkartma, şifre çözme ve çerçeve geri çatma amacıyla yapılan işlemlerin detayları gösterilmiştir.

4.2.2.1. Damga çıkartma

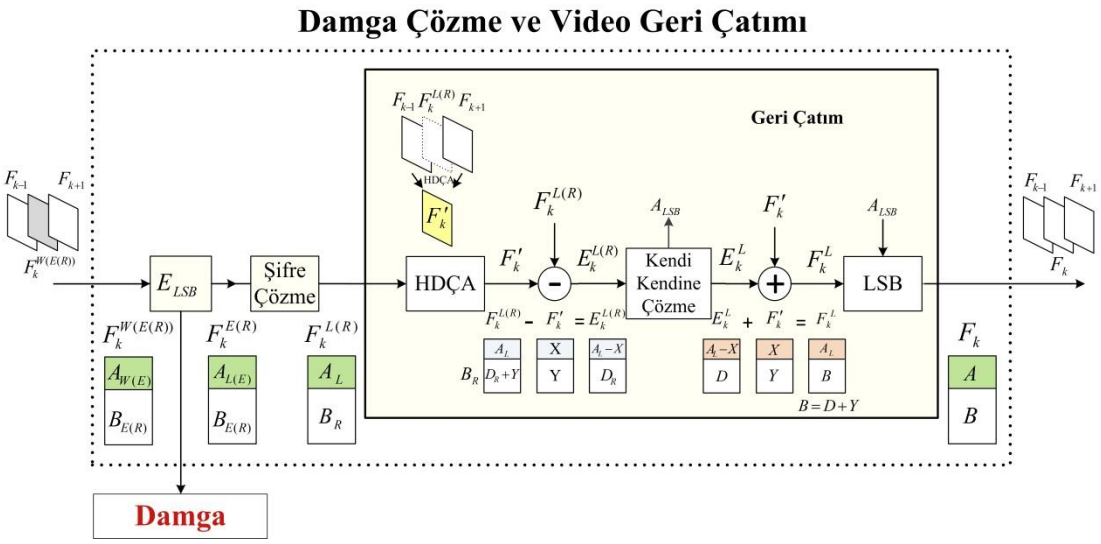
Damgalı ve şifreli videoda çerçevelerin ilk RC satırındaki pikseller damgalıdır. Damga çözücü ilk olarak damgalama anahtarı yardımıyla RC değeri ve bir çerçeveye ilişkin hedef kapasiteyi belirler. Daha sonra, $F_k^{W(E(R))}$ çerçevesinin en üstten RC satırı kadar bölgede (eşdeğer olarak $A_{W(E)}$ bölgesinde) hedef kapasite sayısı kadar LSB toplanarak damga çıkartılır.



Şekil 4.7. Üç olası kullanım senaryosu için damga çözücü adımlarının blok diyagramı.

4.2.2.2. Şifre çözme

Şifreleme işlemi sonrasında gerçekleştirilen damgalama işlemi, çerçevede A bölgesinin pikselllerini değiştirdiğinden, şifre çözme sonrasında A bölgesinin pikselleri hatalı elde edilecektir. Damgalama işlemi şifreli çerçevede piksellerin sadece LSB'sini değiştirmektedir. Şifreleme ve şifre çözme işlemlerinin ikiside bit tabanlı yapıldığından şifre çözme işlemi sonrası A bölgesindeki piksellerin sadece LSB'leri hatalı elde edilecektir. Buna bağlı olarak, tüm çerçevelerdeki damga bilgileri çıkartıldıktan sonra elde edilen $F_k^{E(R)}$ şifreli çerçeve, şifreleme anahtarı yardımıyla şifre çözme işlemine sokularak, hatalı LSB'lere sahip $F_k^{L(R)}$ çerçevesi elde edilir. $F_k^{L(R)}$ çerçevelerinin oluşturduğu video orijinal videoya oldukça yakın bir videodur. Yöntemin bozunum performansı bu elde edilen video ile orijinal videonun karşılaştırılması ile değerlendirilir. Şifre çözme işlemi, öncesinde damga çıkartımının yapılıp yapılmadığına bakılmaksızın her iki durum için de benzer şekilde yapılır. Şifre çözme ve damga çıkartma işlemlerinin bu şekilde birbirinden bağımsız gerçekleştirilebilmesi nedeniyle yöntem ayrışabilir TŞVD sınıfında yer almaktadır.



Şekil 4.8. Bir çerçeve için önerilen yöntemin damga çözme ve çerçeve geri çatma kısmına karşılık gelen detaylı blok diyagram.

4.2.2.3. Damga çıkartma ve video geri çatma

Önerilen yöntemde orijinal videonun hatasız geri çatılabilmesi için damga çıkartma ve şifre çözme işlemlerinin yukarıda anlatıldığı gibi sırayla gerçekleştirilmesi gerekir. Damga çıkartımından A bölgesinin boyutu, şifre çözme işleminden değiştirilmiş çerçeveler elde edilir. Tüm video dizisi için $F_k^{L(R)}$ çerçeveleri elde edildikten sonra geri çatma işlemine geçilir. Geri çatma işleminde, öncelikle damga ekleme adımındakine benzer şekilde bir çerçevenin iki komşusundan aradeğerlemesi elde edilir. Tek numaralı çerçeveler için boşluk oluşturulurken, boşluk oluşturulmuş çift çerçevelerin aradeğerlemesi kullanılmıştır. Damgalama adımında kullanılan aradeğerleme çerçevelerini elde edebilmek için geri çatma işlemi öncelikle tek çerçeveler üzerinde gerçekleştirilir. Orijinal tek numaralı çerçeveler elde edildikten sonra çift numaralı çerçeveler geri çatma işlemine tabi tutulur. F_k' aradeğerlenmiş çerçeve, boşluk oluşturma işlemi esnasında elde edilen HDÇA çerçevesinin aynısı olacağından tersinirlik sağlanmış olur.

$F_k^{L(R)}$ ve F_k' çerçeveleri kullanılarak, $E_k^{L(R)} = F_k^{L(R)} - F_k'$ ifadesinden değiştirilmiş HDÇA hata çerçevesi elde edilir. $E_k^{L(R)}$'nin D_R bölgesinde orijinal çerçevenin A bölgesinin LSB'leri saklıdır. D_R bölgesinden, ters YHD yöntemi ile orijinal hatalar D

ve orijinal çerçevenin A bölgesinin LSB'leri elde edilir. $E_k^{L(R)}$ çerçevesinde D_R bölgesi, elde edilen orijinal D bölgesi hataları ile değiştirilerek E_k^L çerçevesi elde edilir. Daha sonra, $F_k^L = E_k^L + F_k'$ eşitliğinden A bölgesinin LSB'leri hariç orijinal çerçeve elde edilir. Ters YHD'den elde edilen A bölgesinin LSB'leri yerine konarak orijinal F_k çerçevesi kayıpsız bir şekilde geri çatılır.

4.3. Sonuçlar

Önerilen TVD algoritmasının performansı literatürde sıklıkla kullanılan, farklı hareket ve histogram karakteristiğine sahip dört test video dizisi üzerinde sınanmıştır. 352x288 boyutlarındaki her bir test videosu için 25 çerçeve damgalama işlemine tabi tutulmuştur. Eklenecek damga, rastgele oluşturulmuş bit dizisidir. Benzetimler, Intel Core 2 Duo CPU 2.4 GHz işlemcili, 3 GB RAM bellekli ve Windows 2007 işletim sistemi yüklü kişisel bir bilgisayarda MATLAB™ yazılım geliştirme ortamı kullanılarak gerçekleştirilmiştir. Bir çerçevenin damgalanma süresi önerilen yöntem ile ortalama 4,13 dk sürmektedir. Çalışma süresinin uzun olması YHD yönteminde kullanılan aradeğerleme algoritmasından kaynaklanmaktadır. Adil bir karşılaştırma yapabilmek adına boşluk oluşturma aşamasında kullanılan YHD yönteminin parametreleri tüm video dizileri için sabit tutulmuştur.

Sadece şifreleme anahtarına sahip bir alıcı orijinal video değil, onun bozunumlu bir kopyasını elde edebilmektedir. Elde edilen bu yaklaşık videonun kalitesi, PSNR ve SSIM ölçütleri kullanılarak değerlendirilmiştir.

Şifreli ve damgalı videodan damgalama anahtarı yardımıyla yalnızca damga çıkartılabilmektedir. Alıcı tarafta sadece damgalama anahtarı bilindiğinde video içeriğine erişilememektedir. Böyle bir kullanım senaryosunda yöntemin kapasitesi önemli bir rol oynamaktadır. Dört test videosu için önerilen yöntemin erişebildiği en büyük damgalama kapasiteleri Tablo 4.2.'de verilmiştir. Önerilen yöntemin kapasitesini boşluk oluşturma işleminde kullanılan KKTD yöntemi belirlemektedir. Hareketin daha az olduğu videolarda, HDÇA hata çerçevesi daha düzgün ve dik bir

Tablo 4.2. Test videoları için önerilen yöntemin sağladığı en büyük kapasite değerleri

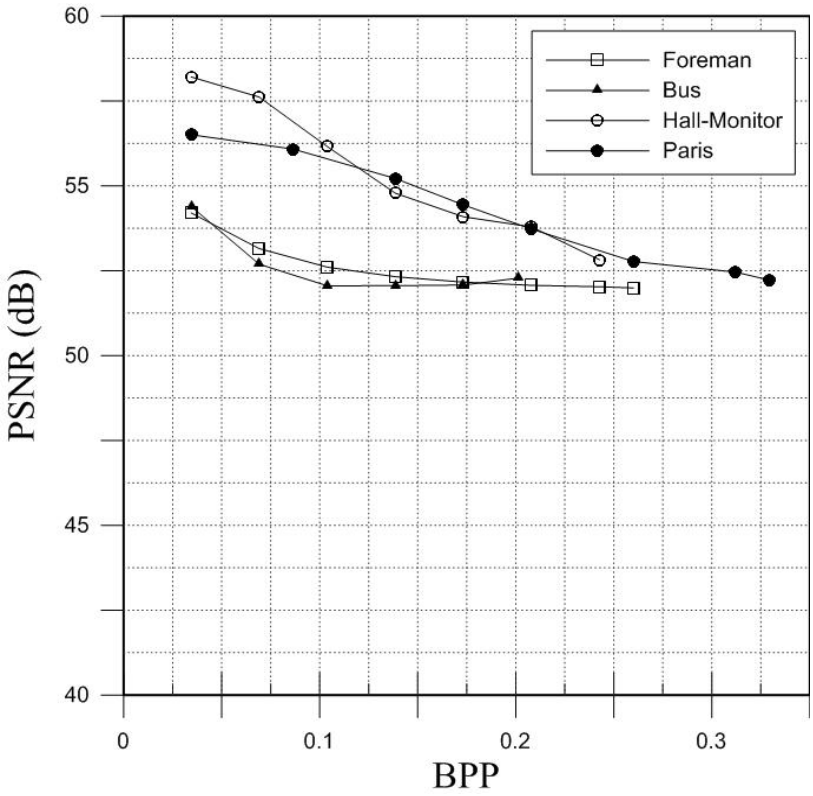
	BUS	FOREMAN	HALL MONITOR	PARIS
Kapasite (BPP)	0,2080	0,2601	0,2428	0,3261

dağılıma sahip olduğundan YHD algoritması daha yüksek veri ekleme kapasitesine erişebilmektedir. Örneğin "paris" videosunda 0,3296 BPP kapasitesine ulaşılabilirken "foreman" ve "hall-monitor" videolarında sırasıyla 0,2601 BPP ve 0,2428 BPP, "bus" videosu içinse 0,2011 BPP kapasite seviyesi elde edilebilmektedir.

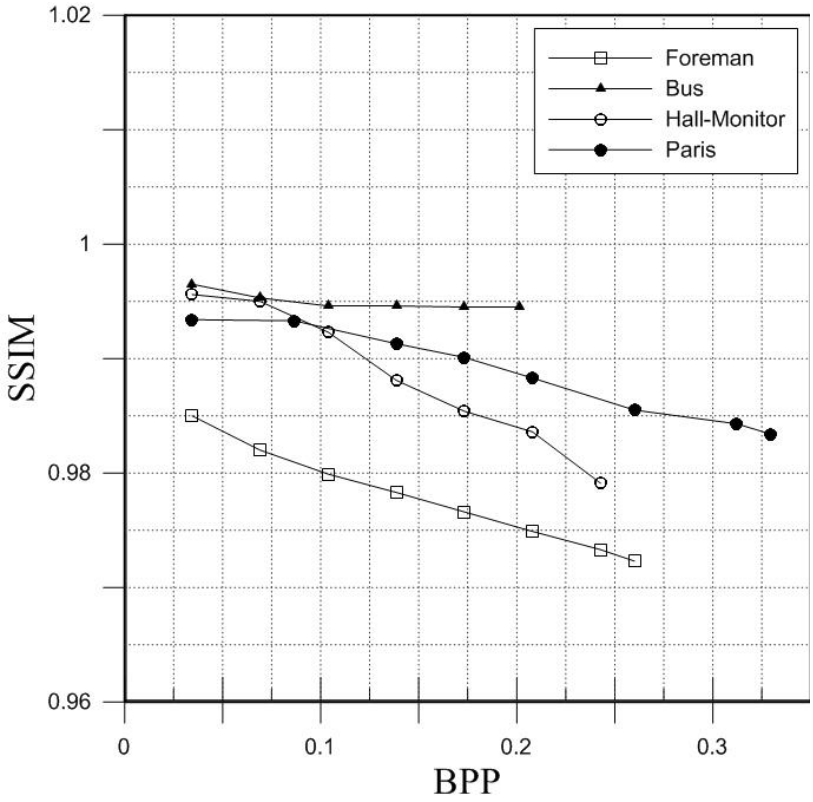
Yalnızca şifreleme anahtarı ile damgaya erişilemez ancak şifre çözme işlemi ile video içeriğine erişilebilir. Test videoları için elde edilen görsel kalite kapasitenin fonksiyonu cinsinden Şekil 4.9., Şekil 4.10. ve Tablo 4.3.'te gösterilmektedir. Şekil 4.9.'da görüldüğü gibi, yöntem hareketin daha az olduğu video dizilerinde daha iyi görsel kalite-kapasite performansı vermektedir. Örneğin 0,2080 BPP kapasite seviyesinde hareketin görece daha az olduğu "paris" videosu için 53,73 dB görsel kaliteye ulaşılırken orta seviye hareketliliğe sahip "foreman" videosunda 52,07 dB görsel kalite elde edilmektedir. Çok hareketli "bus" video dizisinde ise bu kapasite seviyesine ulaşılamamaktadır. Bunun nedeni, videodaki hareketliliğin artmasıyla HDÇA hatalarının büyümesi ve buna bağlı olarak KKTD sonucunda çerçeve üzerinde oluşan bozunum miktarının artmasıdır. Önerilen yöntem yüksek kapasitelere çıkamamaktadır. Bununla birlikte, en karmaşık hareketliliğe sahip video dizisinde bile 0,2 BPP kapasitede damgalama yapılabilmektedir. 0,2 BPP kapasitede 30 saniyelik bir video damgalanmak istendiğinde yaklaşık 1,74 MB veri videoya saklanabilmektedir. Bu damgalama kapasitesi de giriş bölümünde açıklanan tüm uygulama alanları için yeterlidir.

Tablo 4.3. Test videoları için kapasite-bozunum sonuçları.

BPP	Bus		Hall-Monitor		Paris		Foreman	
	PSNR	**SSIM**	**PSNR**	**SSIM**	**PSNR**	**SSIM**	**PSNR**	**SSIM**
0,0344	54,3941	0,9965	58,2082	0,9956	56,5039	0,9934	54,2022	0,9850
0,0691	52,6924	0,9953	57,6146	0,9950	56,2248	0,9935	53,1512	0,9820
0,1039	52,0555	0,9946	56,1544	0,9923	55,8813	0,9926	52,6043	0,9799
0,1386	52,0589	0,9946	54,7817	0,9881	55,2160	0,9913	52,3224	0,9783
0,1733	52,0776	0,9945	54,0886	0,9854	54,4397	0,9901	52,1641	0,9766



Şekil 4.9. Test videoları için PSNR cinsinden önerilen yöntemin sağladığı görsel kalite ve kapasite sonuçları.



Şekil 4.10. Test videoları için SSIM cinsinden önerilen yöntemin sağladığı görsel kalite ve kapasite sonuçları.

Şekil 4.10. ve Tablo 4.3. incelendiğinde, SSIM ölçütü ile de benzer bulguların elde edildiği görülebilir. "foreman", "hall-monitor" ve "paris" videoları için SSIM cinsinden elde edilen kapasite bozunum eğrileri, Şekil 4.9.'da PSNR cinsinden verilen eğrilerle benzerlik göstermektedir. Bununla birlikte "bus" videosu için PSNR ölçütü ile yapılan benzetimlerde elde edilen sonuçlar diğer test videolarına göre daha kötü olmasına rağmen, SSIM ölçütü ile yapılan benzetimlerde daha iyi sonuçlar elde edilmiştir. "bus" videosunda elde edilen bu farklı sonuçlar, video çerçevelerindeki köşe- kenar sayıları ve çerçeveler arası hareketin karakteristiğine bağlı olarak PSNR ve SSIM ölçütlerinin farklı büyüklükleri ölçmesi ile açıklanabilir.

Şifreleme ve damgalama anahtarlarının ikisinin de bilindiği durumlarda hem orijinal video hem de damga kayıpsız geri elde edilebilmektedir. Her iki anahtara sahip bir damga çözücü damgalama ve şifreleme anahtarlarını ayrı ayrı kullanıp damgayı çıkartabileceği veya yaklaşık videoyu geri çatabileceği gibi ikisini birlikte kullanarak orijinal videoyu da geri çatabilmektedir. Şekil 4.9. ve Şekil 4.10.'da açık olduğu

üzere, yöntem görece daha düşük hareketliliğe sahip Hall-Monitor ve Paris videolarında daha iyi kapasite-görsel kalite performansı vermektedir. Hareketin daha fazla olduğu Foreman ve Bus videolarında ise daha düşük kapasite-görsel kalite performansı sağlamaktadır. İlave olarak, hareketin karmaşık olduğu videolarda yüksek kapasitelerde damgalama işlemi yapılamamaktadır. Düşük hareketliliğe sahip videolarda erişilebilecek en yüksek kapasite artmaktadır.

Literatürde bu bölümde önerilen yöntem hariç olmak üzere, şifreli videolar için geliştirilmiş şifreleme öncesi boşluk oluşturmaya dayalı herhangi bir TVD yöntemi henüz mevcut değildir. Buna bağlı olarak, elde edilen sonuçlar başka bir yöntemle elde edilenlerle karşılaştırılamamıştır. Ancak, önerilen yöntem aracılığıyla görsel kalite ve kapasite bakımından tatmin edici bir damgalama gerçekleştirilebilmiştir.

BÖLÜM 5. TARTIŞMA VE SONUÇ

Bu tezde, biri şifreli diğeri şifresiz videolar için iki TVD yöntemi sunulmuştur. Görüntü damgalama için geliştirilen YHD yöntemi şifreli ve şifresiz video dizileri için uyarlanarak algoritmalar geliştirilmiştir. YHD yönteminin video işaretlerine uyarlanmasında karşılaşılan problemler için özgün çözümler önerilmiştir. Geliştirilen yöntemler litaratürde sıklıkla kullanılan test videoları üzerinde denenmiştir. Yöntemlerde kullanılan test videolarının örnek birer çerçevesi Şekil 5.1.'de verilmiştir. Önerilen yöntemlerin damgalama kapasitesi ve görsel kalite bakımından mevcut yöntemlere olan üstünlüğü bilgisayar benzetimleriyle gösterilmiştir.

Yöntemlerin bozunum performansının değerlendirilmesinde literatürde yaygınlıkla kullanılan PSNR ölçütünün yanında SSIM ölçütü de kullanılmıştır. Her iki ölçütle yapılan değerlendirmelerde de tatmin edici sonuçlar elde edilmiştir. Yöntemlerin kapasite karşılaştırmaları ise, gerekli yan bilgi miktarlarının toplam kapasiteden çıkartılmasıyla elde edilen saf kapasite değerleri üzerinden yapılmıştır. Her iki yöntem de HDÇA hatalarını kullandığından hareketin görece daha düşük olduğu videolarda daha iyi görsel kalite - kapasite eğrileri elde edilmiştir. Daha hareketli videolarda ise aradeğerleme hataları yüksek değerler aldığından yöntemlerin kapasite-bozunum performansları düşmektedir. Bununla birlikte, benzetimlerde kullanılan dört test videosu için önerilen yöntemler literatürdeki yöntemlerden daha iyi sonuçlar vermiştir.

Bu tezde önerilen yöntemler YHD tabanlıdır. Şifresiz videolar için geliştirilen yöntemde HDÇA hatalarına YHD uyarlanarak damgalama gerçekleştirilmiştir. Şifreli videolar için geliştirilen ikinci yöntemde ise YHD boşluk oluşturma işlemi için kullanılmıştır. YHD yöntemi tersinir görüntü damgalama yöntemleri arasında kapasite ve görsel kalite bakımından en iyi sonucu vermektedir. YHD, verilen bir

Şekil 5.1. Test videoları için örnek çerçeveler

bozunum kısıtı için damgalanmış işaretin sahip olması gereken dağılımı hesaplayarak bu dağılımı sağlayacak bir damgalama gerçekleştirmektedir. YHD yöntemi, damgalanmış görüntünün dağılımı belirlenen hedef dağılıma eşit olacak bir damgalamayı aritmetik kodlama yardımıyla yapmaktadır. Bloklara ayrılan görüntüde her bir blok için orijinal işaretin dağılımı ve bozunum kısıtı kullanılarak [41]'de verilen yöntem yardımıyla damgalanmış bloğun hedef dağılımı hesaplanmaktadır. Damga bit dizisi, dağılımına hedef dağılıma eşit olması için aritmetik kod çözücü algoritması ile ters sıkıştırılarak orijinal blokla aynı alfabe ve aynı uzunluğa sahip bir sembol dizisi elde edilir. Elde edilen sembol dizisi ile orijinal blok yer değiştirilerek damgalama gerçekleştirilir.

Önerilen birinci yöntem, şifresiz videolar için geliştirilmiştir. Bu yöntemde video çerçeveleri arasındaki zamansal ilinti değerlendirilerek yüksek kapasiteli ve düşük bozunumlu bir damgalama gerçekleştirilmiştir. Tersinirliğin sağlanması için video dizisinde önce çift numaralı çerçevelerin damgalanması gerçekleştirilmiş, sonra damgalı çift numaralı çerçeveler kullanılarak tek numaralı çerçevelerin damgalanması tamamlanmıştır. Damga çözme tarafında ise ters işlemler tek numaralar çerçeveler ile başlatılarak tersinirlik garanti altına alınmıştır. Çerçeveler arasındaki zamansal ilinti, YHD yönteminin video çerçeveleri arasındaki HDÇA hatalarına uygulanması ile değerlendirilmiştir. Aradeğerleme hatalarının kullanılması ile hareket vektörlerinin damga çözücüye iletilmesi ihtiyacı ortadan kaldırılarak yöntemin yan bilgi miktarı azaltılmış, buna bağlı olarak yüksek damgalama kapasitesi elde edilmiştir. Ayrıca, toplam kapasitenin tüm çerçevelere eşit bir biçimde dağıtılması YHD yöntemindeki bozunum kısıtı parametresi kullanılarak sağlanmıştır. Bunun sonucunda damgalama sonrası oluşacak bozunum tüm videoya orantılı bir biçimde dağıtılmış ve görsel kalitenin korunması sağlanmıştır. Bir video

damgalanırken öncelikle damgalamada kullanılacak toplam bit sayısı çerçeve sayısına bölünerek bir çerçeve için hedef kapasite belirlenmiştir. Daha sonra bozunum kısıtı yinelemeli bir biçimde arttırılıp veya azaltılarak hedef kapasiteye ulaşılması sağlanmıştır.

Önerilen birinci yöntemi çeşitli bakımlardan iyileştirmek mümkündür. Yöntemde HDÇA hataları damgalamada kullanmıştır. Yöntemin, hareket dengelemeli öngörü hatalarına damgalanması durumunda sağlayacağı performans incelenebilir. Ayrıca, video çerçeveleri tek seviyeli bir damgalamaya tabi tutulmuşlardır. Bununla birlikte, YHD yönteminin çok seviyeli damgalamada nasıl sonuçlar vereceği araştırılabilir. İlave olarak, çerçevelerin hedef kapasitesine ulaşılmasında, eşik değeri T'nin ve bozunum kısıtı Δ'nın birlikte, aynı iterasyon adımında, adaptif bir algoritmayla değiştirilerek daha üstün görsel kalite ve kapasite performansı elde edebilecek etkin bir damgalama yöntemi geliştirilebilir.

İkinci yöntem şifreli videolar için geliştirilmiştir. Yöntemde YHD yönteminden damgalama amacıyla değil boşluk oluşturma amacıyla yararlanılmıştır. Önerilen TVD yönteminde, damga ekleme için gerekli boşluk şifreleme işlemi öncesinde oluşturulmaktadır. Bununla birlikte, damga çıkartma ve şifre çözme işlemleri birbirinden bağımsız gerçekleştirilebilmektedir. Bu yüzden yöntem *ayrışabilir* ve ŞÖBO tabanlı TŞVD sınıfındadır. Herhangi bir çerçeve için boşluk oluşturma işlemi, çerçevenin belirli piksellerinin LSB'lerinin ilgili çerçeveye ait HDÇA hatalarına YHD yöntemi kullanılarak saklanmasıyla gerçekleştirilmektedir. Önerilen yöntemde damgalama işlemi ise şifreli video üzerinde gerçekleştirilmektedir. Boşluk oluşturma aşamasında LSB'leri saklanan piksellerin LSB'leri damgalanacak bitler ile değiştirilmesiyle damgalama işlemi gerçekleştirilir. Kapasite ve görsel kalite bakımından önerilen yöntemin günümüz uygulamalarının gereksinimlerini rahatlıkla karşılayabileceği bilgisayar benzetimlerinde elde edilen sonuçlar ile gösterilmiştir.

Şifrelenmiş ve damgalanmış video dizisi için damga çözücüde damgalama ve şifreleme anahtarlarının yalnızca birinin veya her ikisinin bulunduğu durumları kapsayan muhtemel üç senaryo ile karşılaşılabilmektedir. Sadece şifreleme

anahtarına sahip bir alıcı damgalanmış şifreli video üzerindeki şifreyi kaldırıp orijinal videoya yakın bir videoyu elde edebilir. Böyle bir durumda, alıcı videoya eklenen damgadan habersiz bir biçimde video içeriğine bozunumlu da olsa erişebilmektedir. Sadece damgalama anahtarına sahip bir çözücü, sadece ters damgalama işlemleri yardımıyla damgaya ulaşabilmekteyken video içeriğine erişemez. Her iki anahtara sahip bir çözücü ise damgalama anahtarı yardımıyla damgaya, şifreleme anahtarı yardımıyla da video içeriğine kayıpsız bir biçimde erişebilmektedir. Bu üç kullanım durumundan bir veya birkaçı uygulama amacına göre kullanılabilmektedir.

Önerilen yöntem birçok yönden iyileştirilmeye açıktır. Her bir çerçevenin kendine ait en yüksek kapasitede damgalanmasına dayalı bir yaklaşım kapasitede artış sağlayabilir. Ayrıca, YHD yönteminin parametrelerini her bir çerçeve veya video dizisi için özel olarak belirleyen uyarlamalı bir yaklaşımın sonuçlar üzerindeki etkisi incelenebilir. Son olarak, birden fazla LSB'nin damgalama için kullanılmasının kapasite-bozunum performansı üzerindeki etkileri araştırılabilir.

KAYNAKLAR

[1] Podilchuk, C.I., Delp, E.J., "Digital watermarking: algorithms and applications", IEEE Signal Processing Magazine, 18(4), 33-46, DOI: 10.1109/79.939835, 2001.

[2] Wolfgang, R.B., Podilchuk, C.I., Delp, E.J., "Perceptual watermarks for digital images and video", Proceedings of the IEEE, 87(7), 1108-1126, DOI: 10.1109/5.771067, 1999.

[3] Hernandez, J.R., Perez-Gonzalez, F., "Statistical analysis of watermarking schemes for copyright protection of images", Proceedings of the IEEE, 87(7), 1142-1166, DOI: 10.1109/5.771069, 1999.

[4] Su, K., Kundur, D., Hatzinakos, D., "Statistical invisibility for collusion-resistant digital video watermarking", IEEE Transactions on Multimedia, 7(1), 43-51, DOI: 10.1109/TMM.2004.840617, 2005.

[5] Wang, X., Wu, J., Niu, P., "A new digital image watermarking algorithm resilient to desynchronization attacks", IEEE Transactions on Information Forensics and Security, 2(4), 655-663, DOI: 10.1109/TIFS.2007.908233, 2007.

[6] Doerr, G., Dugelay, J.L., "A guide tour of video watermarking" Elsevier Signal Processing: Image Communication, 18(4), 263-282, http://dx.doi.org/10.1016/S0923-5965(02)00144-3, 2003.

[7] Langelaar, G.C., Setyawan, I., Lagendijk, R.L., "Watermarking digital image and video data a state-of-the-art overview", IEEE Signal Processing Magazine, 17(5), 20-46, DOI: 10.1109/79.879337, 2000.

[8] Moulin, P., Koetter, R., "Data-hiding codes", Proceedings of the IEEE, 93(12), 2083-2126, DOI: 10.1109/JPROC.2005.859599, 2005.

[9] Wu, M., Liu, B., "Data hiding in image and video .I. fundamental issues and solutions", IEEE Transactions on Image Processing, 12(6), 685-695, DOI: 10.1109/TIP.2003.810588, 2003.

[10] Feng, J.B., Lin, I.C., Tsai, C.S., Chu Y.P., “Reversible watermarking: Current status and key issues”, International Journal of Network Security, 2(3), 161-171, 2006.

[11] Thodi, D.M., Rodriguez, J.J., “Expansion embedding techniques for reversible watermarking”, IEEE Transactions on Image Processing, 16(3), 721-730, DOI: 10.1109/TIP.2006.891046, 2007.

[12] Celik, M.U., Sharma, G., Tekalp, A.M., Saber, E., “Lossless generalized-LSB data embedding”, IEEE Transactions on Image Processing, 14(14), 253-266, DOI: 10.1109/TIP.2004.840686, 2005.

[13] Wu, M., Yu, H., Liu, B., “Data hiding in image and video .II. Designs and applications”, IEEE Transactions on Image Processing, 12(6), 696-705, DOI: 10.1109/TIP.2003.810589, 2003.

[14] De Vleeschouwer, C., Delaigle, J.F., Macq, B., “Invisibility and application functionalities in perceptual watermarking an overview”, Proceedings of the IEEE, 90(1), 64-77, DOI: 10.1109/5.982406, 2002.

[15] Zhang, X., “Reversible data hiding in encrypted images”, IEEE Signal Processing Letters, 18(4), 255–258, DOI: 10.1109/LSP.2011.2114651, 2011.

[16] Hong, W., Chen, T.S., Wu, H.Y., “An improved reversible data hiding in encrypted images using side match”, IEEE Signal Processing Letters, 19(4), 199–202, DOI: 10.1109/LSP.2012.2187334, 2012.

[17] Zhang, X., “Separable reversible data hiding in encrypted image,” IEEE Transactions on Information Forensics and Security, 7(2), 826–832, DOI: 10.1109/TIFS.2011.2176120, 2012.

[18] Qian, Z., Zhang, X., “Reversible data hiding in encrypted image with distributed source encoding”, IEEE Transactions on Circuits and Systems for Video Technology, 26(4), 636-646, DOI: 10.1109/TCSVT.2015.2418611, 2016.

[19] Zhou, J., Sun, W., Dong, L., Liu, X., Au, O.C., Tang, Y.Y., “Secure reversible image data hiding over encrypted domain via key modulation”, IEEE Transactions on Circuits and Systems for Video Technology, 26(3), 441-452, DOI: 10.1109/TCSVT.2015.2416591, 2016.

[20] Ma, K., Zhang, W., Zhao, X., Yu, N., Li, F., “Reversible data hiding in encrypted images by reserving room before encryption”, IEEE Transactions on Information Forensics and Security, 8(3), 553-562, DOI: 10.1109/TIFS.2013.2248725, 2013.

[21] Tian, J., “Reversible data embedding using a difference expansion”, IEEE Transactions on Circuits and Systems for Video Technology, 13(8), 890–896, DOI: 10.1109/TCSVT.2003.815962, 2003.

[22] Alattar, A.M, “Reversible watermark using difference expansion of a generalized integer transform”, IEEE Transactions on Image Processing, 13(8), 1147–1156, DOI: 10.1109/TIP.2004.828418, 2004.

[23] Kim, H.J., Sachnev, V., Shi, Y.Q., Nam, J., Choo, H.G., “A novel difference expansion transform for reversible data embedding”, IEEE Transactions on Information Forensics and Security, 3(3), 456–465, DOI: 10.1109/TIFS.2008.924600, 2008.

[24] Lin, C.C., Yang, S.P., Hsueh, N.L., “Lossless data hiding based on difference expansion without a location map” , 2008 Congress on Image and Signal Processing, 2, 8–12, DOI: 10.1109/CISP.2008.64, 2008.

[25] Hu, Y., Lee, H.K., Li, J., “DE-based reversible data hiding with improved overflow location map”, IEEE Transactions on Circuits and Systems for Video Technology, 19(2), 250–260, DOI: 10.1109/TCSVT.2008.2009252, 2009.

[26] Luo, L., Chen, Z., Chen, M., Zeng, X., Xiong, Z., “Reversible image watermarking using interpolation technique”, IEEE Transactions on Information Forensics and Security, 5(1), 187-193, DOI: 10.1109/TIFS.2009.2035975, 2010.

[27] Ni, Z., Shi, Y.Q., Ansari, N., Su, W., “Reversible data hiding”, IEEE Transactions on Circuits and Systems for Video Technology, 16(3), 354–362, DOI: 10.1109/TCSVT.2006.869964, 2006.

[28] Hwang, J., Kim, J.W., Choi, J.U., “A reversible watermarking based on histogram shifting”, Proceedings International Workshop on Digital Watermarking, Lecture Notes in Computer Science, Jeju Island, Korea, 4283, 348–361, DOI: 10.1007/11922841_28, 2006.

[29] Lin, C.C., Hsueh, N.L., “A lossless data hiding scheme based on three-pixel block differences”, Pattern Recognition, 41(4), 1415–1425, http://dx.doi.org/10.1016/j.patcog.2007.09.005, 2008.

[30] Kim, K.S., Lee, M.J., Lee, H.Y., Lee, H.K., “Reversible data hiding exploiting spatial correlation between sub-sampled images”, Pattern Recogniton, http://dx.doi.org/10.1016/j.patcog.2009.04.004, 42(11), 3083–3096, 2009.

[31] Zhang, X., "Reversible data hiding with optimal value transfer", IEEE Transactions on Multimedia, DOI: 10.1109/TMM.2012.2229262, 15(2), 316-325, 2013.

[32] Qin, C., Chang, C.C., Huang, Y.H., Liao, L.T., "An inpainting-assisted reversible steganographic scheme using a histogram shifting mechanism", IEEE Transactions on Circuits and Systems for Video Technology, 23(7), 1109-1118, DOI: 10.1109/TCSVT.2012.2224052, 2013.

[33] Qin, C., Chang, C.C., Hsu, T.J., "Reversible data hiding scheme based on exploiting modification direction with two steganographic images", Multimedia Tools and Applications, DOI: 10.1007/s11042-014-1894-5, 74(15), 5861-5872, 2015.

[34] Zhang, W., Hu, X., Li, X., Yu, N., "Recursive histogram modification: establishing equivalency between reversible data hiding and lossless data compression", IEEE Transactions on Image Processing, 22(7), 2775-2785, DOI: 10.1109/TIP.2013.2257814, 2013.

[35] Chung, K.L., Yang, W.J., Chang, T.C., Liao, H.Y.M., "Efficient multilevel reversible data hiding for video sequences using temporal and spatial approach", Proceedings of 2009 APSIPA Annual Summit and Conference, Sapporo-Japan, 573-582, 2009.

[36] Zeng, X., Chen, Z.Y., Chen, M., Xiong, Z., "Reversible video watermarking using motion estimation and prediction error expansion", Journal of Information Science and Engineering - JISE , 27(2), 465-479, 2011.

[37] Vural, C., Baraklı, B., "Reversible video watermarking using motion-compensated frame interpolation error expansion", Signal, Image and Video Processing, 9(7), 1613-1623, DOI: 10.1007/s11760-014-0618-7, 2015.

[38] Vural, C., Baraklı, B., "Adaptive reversible video watermarking based on motion-compensated prediction error expansion with pixel selection", Signal, Image and Video Processing, 10(7), 1225-1232, DOI: 10.1007/s11760-016-0881-x, 2016.

[39] Lin, S.J., Chung, W.H., "The scalar scheme for reversible information-embedding in gray-scale signals: Capacity evaluation and code constructions", IEEE Transactions on Information Forensics and Security, 7(4), 1155–1167, DOI: 10.1109/TIFS.2012.2197614, 2012.

[40] Willems, F., Maas, D., Kalker, T., "Semantic lossless source coding", Proc. 42nd Annu. Allerton Conf. Communication, Control and Computing, Monticello, 2004.

[41] Hu, X., Zhang, W., Hu, X., Yu, N., Zhao, X., Li, F., “Fast estimation of optimal marked-signal distribution for reversible data hiding”, IEEE Transactions on Information Forensics and Security, 8(5), 779–788, DOI: 10.1109/TIFS.2013.2256131, 2013.

[42] Langdon, G.G., “An introduction to arithmetic coding”, IBM Journal of Research and Development, 28(2), 135-149, 1984.

[43] Witten, I.H., Neal, R.M., Cleary, J.G., “Arithmetic coding for data compression”, Communications of the ACM, 30(6), 520-540, DOI=http://dx.doi.org/10.1145/214762.214771, 1987.

[44] Zhai, J., Yu, K., Li, J., Li, S., “A low complexity motion compensated frame interpolation method”, 2005 IEEE International Symposium on Circuits and Systems, 5, 4927–4930, DOI: 10.1109/ISCAS.2005.1465738, 2005.

ÖZGEÇMİŞ

İbrahim YILDIRIM, 12.05.1980'de Erzurum'da doğdu. İlköğrenimine Erzurum'da başladı ve orta öğrenimiyle birlikte Kocaeli'de tamamladı. 1998 yılında Sakarya Arifiye Anadolu Öğretmen Lisesi'nden mezun oldu. 1998 yılında başladığı Kocaeli Üniversitesi Elektronik ve Haberleşme Mühendisliği Bölümü'nü 2002 yılında bitirdi. 2003 yılında başladığı Kocaeli Üniversitesi Elektronik ve Haberleşme Mühendisliği Bölümü'ndeki yüksek lisans eğitimini 2006 yılında bitirdi. 2005 yılından beri çeşitli özel ve kamu kuruluşlarında mühendis ünvanıyla görev yaptı.