



YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

kesirli fourier dönüşümü analiziyle..

Yüksek Lisans Tezi

uğur ustaoglu

YILDIZ TEKNİK ÜNİVERSİTESİ
KÜTÜPHANE VE DOKÜMANTASYON
DAİRE BAŞKANLIĞI

Yer No (DDC): R.373-156

Kayıt No : 4291
Geldiği Yer : Fen Bil. End.
Tarih : 27/10/08
Fiyat : 3,30 -
Fatura No :
Analiz No : 42
Ek :

**YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

İÇİNDEKİLER

Sayfa

SİMGE LİSTESİ

X-66

KISALTMA LİSTESİ

ŞEKİL LİSTESİ

ÇİZELGE LİSTESİ

ÖNSÖZ

**KESİRLİ FOURIER DÖNÜŞÜMÜ ANALİZİYLE İLİNTİ TABANLI
İKİ BOYUTLU DAMGALAMA**

ABSTRACT

GİRİŞ

1.1 Damgalama İşlemi

2 DİJİTAL DAMGALAMA VE DİJİTAL DÖNÜŞÜMLER

2.1 Ayrık Fourier Transformasyonu
2.2 Ayrık Kosinüs Transformasyonu
2.3 Kesikli Fourier Dönüşümü

2.4 FBE Elektronik ve Haberleşme Anabilim Dalı Elektronik Programında
Hazırlanan

3 DİJİTAL DAMGALAMA VE DİJİTAL DÖNÜŞÜMLER

3.1 Gözetli Kapalı Damgalama
3.1.1 Gözetli Kapalı Damgalama
3.1.2 Gözetli Kapalı Damgalama

3.2 Trigonometrik Karşılıklı Dönüşüm

3.3 Sabit İlişki Karşılıklı Dönüşüm

3.4 Düşük Frekanslı ve Yüksek Frekanslı Dönüşüm

3.5 Wavelet Dönüşümü

4 KFD ÜZAYI

4.1 KFD Üzayı

4.2 Algortima

4.3 İlişki Tabanı

4.4 İlişki Tabanı

5 ALGORİTMA

5.1 Algortima

5.2 Sıralama

Elektronik Müh. Uğur USTAOĞLU

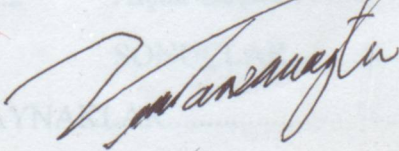
YÜKSEK LİSANS TEZİ

Tez Danışmanı : Yrd.Doç.Dr. Lütfiye DURAK

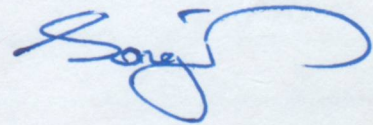
Yrd. Doç. Dr. Lütfiye DURAK

L. Durak

Prof. Dr. Vedat TAVŞANÖĞLU



Yrd. Doç. Dr. Songül ALBAYRAK



İÇİNDEKİLER

Sayfa

SİMGE LİSTESİ.....	iv
KISALTMA LİSTESİ.....	v
ŞEKİL LİSTESİ.....	vi
ÇİZELGE LİSTESİ.....	vii
ÖNSÖZ	viii
ÖZET	ix
ABSTRACT	x
1. GİRİŞ.....	1
1.1 Damgalama İşlemi.....	2
2. DİJİTAL DAMGALAMADA KULLANILAN DÖNÜŞÜMLER.....	3
2.1 Ayırık Fourier Dönüşümü	3
2.2 Ayırık Kosinüs Dönüşümü	4
2.3 Kesirli Fourier Dönüşümü.....	6
2.4 Dalgacık Dönüşümü	11
3. DİJİTAL DAMGALAMA KAVRAMLARI VE ALGORİTMALARI	12
3.1 Gözü Kapalı Damgalama, Doğrusal İlintiliyle Algılama	12
3.1.1 Gözü Kapalı Damgalama, Doğrusal İlintiliyle Algılama Algoritması.....	13
3.1.2 Gözü Kapalı Damgalama, İlinti Katsayısıyla Algılama Algoritması.....	14
3.2 Trellis Algoritması Kullanarak Damgalama Yöntemi	14
3.3 Sabit İlinti Katsayıları Yöntemi	16
3.4 Bilgili Kodlama ve Kirli Kağıt Algoritması.....	18
3.5 Watson'un AKD Tabanlı Görsel Modeli	19
4. KFD UZAYINDA İLİNTİ TABANLI DAMGALAMA ALGORİTMALARI ...	24
4.1 KFD uzayında Kare İşaret Kullanarak İlinti Tabanlı Damgalama.....	25
Algoritması Uygulaması.....	25
4.2 İlinti Tabanlı, Damga İşareti Olarak Çörp Kullanılarak Damgalama	27
4.3 İlinti Tabanlı, 2-Boyutlu Dalga Katarıyla Damgalama Algoritması.....	32
5. ALGORİTMALARIN PERFORMANSLARI.....	34
5.1 Algılanabilirlik ve Hata Sayıları.....	34
5.2 Saldırılar	35
5.2.1 Gürültü Saldırısı	36
5.2.2 Alçak Geçiren Filtre Saldırısı.....	41
6. SONUÇLAR	46
KAYNAKLAR.....	48

EKLER	51
Ek1 Çalıştırılan Algoritmalar.....	51
Ek2 Algoritmalarda Kullanılan Fonksiyonlar.....	51
ÖZGEÇMİŞ	66

SİMGE LİSTESİ

c	Gelen İşaret
c_0	Orijinal İşaret
c_w	Damgalanmış İşaret
m	Mesaj İşareti
T_s	Örnekleme Periyodu
v_0	Resim Blokları
v_w	Damgalanmış Resim Bloğu
w_a	Eklenen Damga
w_r	Referans Örüntüsü
w_m	Damgalanan Resim
W_0	Damga Kümesi
z_{cc}	İlinti
α	Damga Gücü

KISALTMA LİSTESİ

AFD	Ayrık Fourier Dönüşümü	1
AGF	Alçak Geçiren Filtre	10
AKD	Ayrık Kosinüs Dönüşümü	1
AZFD	Ayrık Zaman Fourier Dönüşümü	15
DD	Dalgacık Dönüşümü	16
FD	Fourier Dönüşümü	17
İT-ÇD	İlinti Tabanlı, Damga İşareti Olarak Çörp Kullanılarak Damgalama	1
İT-DKD	İlinti Tabanlı, Damga İşareti Olarak Çörp Kullanılarak Damgalama	20
KFD	Kesirli Fourier Dönüşümü	20
KZFD	Kısa Zamanlı Fourier Dönüşümü	21
POCS	Dışbükey Kümelere İzdüşüm	1
WD	Wigner Dağılımı	26
Şekil 4.3	Kara ileme ile damgalanmış Lena resmi	27
Şekil 4.4	Perforasyon sonucu	27
Şekil 4.5	Seçilmiş Selekson 6, Buile frekansı 7 olan çörp işareti ve frekansları 1000 çevrilmiş	30
Şekil 4.6	Çörp ve frekansları 1000 çevrilmiş çörp ile damgalanmış resmi	30
Şekil 4.7	Çörp işaretini seçilmiş çörp	31
Şekil 4.8	İkinci işaretli çörp ile damgalanmış resmi	32
Şekil 4.9	Yüksek frekanslı damga işaretleri	33
Şekil 4.10	Mikrofonun dinledi damgalama damgalanmış resmi	33
Şekil 5.1	İT-ÇD ve İT-DKD algoritmaları gördüğü gözetim hızı sayıları karşılaştırması	39
Şekil 5.2	İT-ÇD ve İT-DKD gördüğü gözetim hızı sayıları karşılaştırması	39
Şekil 5.3	Çörpün gözetim hızı 5 ile sınırlanmış Lena resmi	40
Şekil 5.4	Çörpün gözetim hızı 10 ile sınırlanmış Lena resmi	40
Şekil 5.5	İT-ÇD ve İT-DKD algoritmaları AGF gözetim hızı sayıları karşılaştırması	44
Şekil 5.6	İT-ÇD ve İT-DKD algoritmaları AGF gözetim hızı sayıları karşılaştırması	44
Şekil 5.7	AGF gözetim hızı 5 ile sınırlanmış Lena resmi	45
Şekil 5.8	AGF gözetim hızı 10 ile sınırlanmış Lena resmi	45

ŞEKİL LİSTESİ

Şekil 1.1 Basit damgalama sistemi.....	1
Şekil 2.1 Basamak fonksiyonu.....	10
Şekil 3.1 Gözü kapalı damgalama, doğrusal ilintiyle algılama.....	1
Şekil 3.2 Trellis algoritması yapısı.....	15
Şekil 3.3 Normalize ilinti (Cox vd., 2002).....	16
Şekil 3.4 Gram-Schmidt dikgenleştirmenin kullanıldığı durumda (Cox vd., 2002).....	17
Şekil 3.5 Costa'nın kanalı.....	1
Şekil 3.6 Gözün uzamsal frekans tepkisi (Cox vd., 2002).....	20
Şekil 3.7 Gözün ışık tayfı frekanslarına tepkisi (Cox vd., 2002).....	20
Şekil 3.8 Gözün zamansal frekansa tepkisi (Cox vd., 2002).....	21
Şekil 4.1 Damgalama algoritması.....	1
Şekil 4.2 Kare damga işaretleri.....	26
Şekil 4.3 Kare işaretle damgalanmış Lena resmi.....	27
Şekil 4.4 Performans analizi.....	27
Şekil 4.5 Başlangıç frekansı 6, bitiş frekansı 7 olan çörp işareti ve frekansları ters çevrilmiş eşi.....	30
Şekil 4.6 Çörp ve frekansları ters çevrilmiş çörp ile damgalanmış resim.....	30
Şekil 4.7 Çörp ve eksi işaretli çörp.....	31
Şekil 4.8 Eksi işaretli çörp ile damgalanmış resim.....	32
Şekil 4.9 Yüksek ilintili damga işaretleri.....	32
Şekil 4.10 Maksimum ilintili damgalarla damgalanmış resim.....	33
Şekil 5.1 İT-ÇD ve İT-DKD algoritmaları gürültü gücü- hata sayısı karşılaştırması.....	39
Şekil 5.2 İT-ÇD ve İT-DKD gürültü gücü- algısal uzaklık karşılaştırması.....	39
Şekil 5.3 Gürültü gücü 5 ile saldırılmış Lena resmi.....	40
Şekil 5.4 Gürültü gücü 10 ile saldırılmış Lena resmi.....	40
Şekil 5.5 İT-ÇD ve İT-DKD algoritmaları AGF gücü- hata sayısı karşılaştırması.....	44
Şekil 5.6 İT-ÇD ve İT-DKD algoritmaları AGF gücü- algısal uzaklık miktarı karşılaştırması.....	44
Şekil 5.7 AGF gücü 5 ile saldırılmış Lena resmi.....	45
Şekil 5.8 AGF gücü 10 ile saldırılmış Lena resmi.....	45

ÇİZELGE LİSTESİ

Çizelge 3.1 Frekans – Hassaslık çizelgesi (Watson, 1993).....	22
Çizelge 4.1 Çörp performansları	29
Çizelge 4.2 Çörp ve eksi işaretli çörp kullanarak damgalama algoritması	31
Çizelge 5.1 İT-ÇD için hata sayıları.....	34
Çizelge 5.2 İT-ÇD için algısal uzaklıklar.....	35
Çizelge 5.3 İT-DKD için hata ve algı bozulma miktarı sayıları	35
Çizelge 5.4 Gürültü gücü 5 için İT-ÇD algoritması hata sayısı	36
Çizelge 5.5 Gürültü gücü 5 için İT-ÇD algoritması algısal uzaklık miktarı	37
Çizelge 5.6 Gürültü gücü 10 için İT-ÇD algoritması hata sayısı	37
Çizelge 5.7 Gürültü gücü 10 için İT-ÇD algoritması algısal uzaklık miktarı	38
Çizelge 5.8 Gürültü gücü 5 ve 10 için İT-DKD algoritması hatalı bit sayısı ve algısal uzaklık miktarı.....	38
Çizelge 5.9 AGF 5 için İT-ÇD algoritması hata sayısı	41
Çizelge 5.10 AGF gücü 5 için İT-ÇD algoritması algısal uzaklık miktarı	42
Çizelge 5.11 AGF gücü 10 için İT-ÇD algoritması hata sayısı	42
Çizelge 5.12 AGF gücü 10 için İT-ÇD algoritması algısal uzaklık miktarı	43
Çizelge 5.13 AGF gücü 5 ve 10 için İT-DKD algoritması hatalı bit sayısı ve algısal uzaklık miktarı	43

ÖNSÖZ

Bilgiye erişebilmenin en kolay yollarından biri bilginin dijital ortamda saklanmasıdır. Ancak önemli ve değerli bilgilerin dijital ortama aktarılması, bilgilerin güvenliğinin tehlikeye girmesine yol açmıştır. Dijital işaretler kolayca çoğaltılabildiği için dijital bilginin telif haklarının korunması büyük bir sorun haline gelmiştir. Bu tezde dijital ortamdaki resimlerin görsel olarak değiştirmeden içine bilgi gömülebilmesini sağlayacak yeni bir damgalama algoritması önerilmiştir.

Gerçekleştirilen çalışmada bilgileri ve desteğiyle her zaman yanımda olan Sayın Yrd. Doç. Dr. Lütfiye Durak hocama ve özellikle dijital işaret işleme ve dijital görüntü işleme dersleriyle beni işaret işleme konusuyla tanıştıran Sayın Prof. Dr. Vedat Tavşanoğlu hocama teşekkürü borç bilirim. Her zaman olduğu gibi öğrencilik yaşamımda da yanımda olan ve beni destekleyen sevgili anne ve babama da teşekkür ederim.

ÖZET

Günümüzde bilgiye daha kolay erişebilmek için dijital ortam kullanılmaktadır. Ancak önemli ve değerli bilgilerin dijital ortama aktarılması bu bilgilerin güvenliğinin tehlikeye girmesine yol açmıştır. Bilgilerin kime ait olduğu, telif haklarının korunmabilmesi açısından çok önemli hale gelmiştir. Dijital işaretlerin algısal özelliklerini kaybetmeden gizli bilgi taşımasına dijital damgalama denir.

Bu tezde dijital resimler için yeni bir dijital damgalama algoritması önerilmiştir. Algoritma, kesirli Fourier uzayında blok tabanlı gerçekleştirilmiştir. Damga işareti olarak 2-boyutlu çörp işareti ve 2-boyutlu dalga katarı kullanılmıştır. Alıcı kısmında mesaj ilintiyle tespit edilmiştir. Uygulamada 128×128 piksellik resimler kullanılmış ve her bir resme toplam 64 bit gömülmüştür. Algoritmanın performansı, alçak geçiren filtreleme ve gürültü saldırılarından sonra resimdeki hata sayılarındaki değişimle ölçülmüştür. Ayrıca damgalama işleminin resmi algısal olarak ne kadar değiştirdiği de tespit edilmiştir.

Performans analizleri sonucunda algoritmada damga olarak 2-boyutlu çörp kullanıldığında alçak geçiren filtre gürbüzlüğünün daha fazla, damga olarak 2-boyutlu dalga katarı kullanıldığında görsel farklılığın daha düşük olduğu gösterilmiştir.

Anahtar Kelimeler: Dijital damgalama, kesirli Fourier dönüşümü, 2-Boyutlu çörp, veri saklama.

digital works without changing their perceptual properties is called digital watermarking.

the watermarking process.

pass filtering attacks and wave train watermark is perceptually more invisible.

1. GİRİŞ

Günümüzde dijital sistemlerin gelişmesi, dijital ortamlardaki resim, müzik ve video gibi bir, iki ve üç boyutlu işaretlerin yaygınlaşmasını sağlamıştır. Ancak bu gelişme beraberinde, telif haklarının korunması ve güvenliğinin artırılması gibi konuları gündeme getirmiştir. Bu noktada günümüzden yaklaşık 800 yıl önce kullanılmaya başlanan damgalama işleminin dijital işaretler için de kullanılabileceği düşüncesi belirmiştir. İlk olarak 1282 yılında İtalya'da kullanılan damgalama işlemi o zamanlar kâğıt kalıplarının içine yerleştirilen küçük tel desenlerinden oluşmaktaydı. On sekizinci yüzyılda Avrupa ve Amerika'da telif hakları için yaygın olarak kullanılan bu metot, daha sonraları para güvenliğinin artırılabilmesi için de kullanılmaya başlanmıştır. Dijital damgalama terimi ilk olarak 1988 yılında Komatsu ve Tominaga tarafından kullanılmıştır (Komatsu ve Tominaga, 1988). Ancak yaygın olarak dijital damgalama konusunda çalışmalar 1995'li yıllarda başlamıştır. Günümüzde dijital damgalama denilince orijinal işaretin algılanabilir olarak değiştirilmeden içine o çalışmayla ilgili bilginin yerleştirilmesi anlaşılmaktadır. Bu konuda çeşitli algoritmalar geliştirilirken, ticari anlamda da uygulamalar yapılmaktadır.

Dijital damgalama algoritmalarında en genel anlamda iki ana kısımdan bahsedilebilir. Bunlar:

- Damgalayıcı Bölümü,
- Alıcı Bölümü

olarak adlandırılırlar. Damgalayıcı bölümünde orijinal işaret (c_0) belli bir algoritma ile değiştirilir. Değiştirme işlemindeki amaç c_0 'daki değişimin insana ait algısal sistemler tarafından en az şekilde fark edilmesidir. Değiştirilen işarete damgalanmış işaret (c_w) denir.

Alıcı bölümünde c_w 'dan damga işareti çıkartılır ve yorumlanır. Alıcılar da kendi aralarında iki ana tipe ayrılabilir. Bunlar:

- Gözü Kapalı Alıcılar,
- Bilgili Alıcılar'dır.

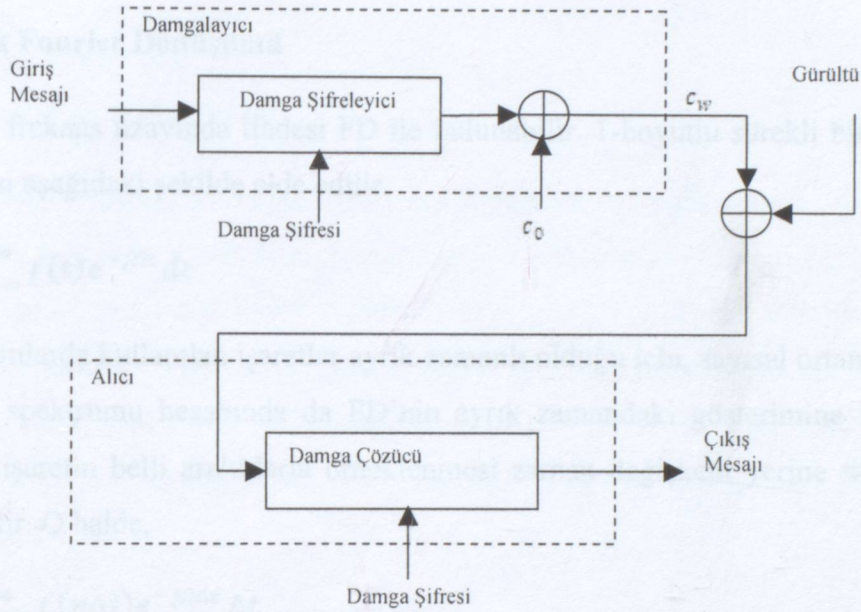
Gözü kapalı alıcılarda damgalama sisteminin alıcı kısmında c_0 bilinmezken, bilgili alıcılarda bilinmektedir.

Literatürde damgalama algoritmalarının bazıları, resmin bloklara ayrılmasıyla ilinti tabanlı (Miller vd., 2000), Trellis kanal kodlaması kullanılarak (Miller vd., 2002), yan bilgi kullanılarak (Cox ve Miller, 2004), algısal model kullanarak (Li ve Cox, 2005), kesirli fourier bölgesinde (Djurovic vd., 2000) ve kesirli fourier bölgesinde çörp işaretleri kullanılarak

(Zhang vd., 2006) gerçekleştirilmiştir. Bu algoritmalar ilerleyen bölümlerde detaylandırılacaktır.

1.1 Damgalama İşlemi

Bir damgalama sistemi en basit haliyle Şekil 1.1’de gösterildiği gibidir.



Şekil 1.1 Basit damgalama sistemi.

Giriş mesajı, c_0 'ın içine eklenen ve çoğunlukla dijital işaretin telif haklarını içeren bilgidir. Bu mesaj, damga şifreleyici tarafından c_0 'ın içine yerleştirilebilecek damga işaretlerine dönüştürülür ve c_0 , içinde damga işaretlerini taşıyacak şekilde değiştirilir. Damgalanan işaret iletim ortamında çeşitli saldırılardan ya da gürültülerden etkilenebilir. İletim ortamında meydana gelebilecek etkiler Şekil 1.1’de gürültü olarak modellenmiştir. Burada bahsedilen gürültü, iletilen işarete bağlı bazı işaret işleme tekniklerini kapsar. Örneğin, dijital işaret bir resim ise, resmin alıcıya ulaşmadan belli bir kısmının kesilmesi ya da resmin döndürülmesi de bir gürültü modelidir. Bunun yanında özellikle damgayı yok etmeye yönelik saldırılar da Şekil 1.1’de gürültü olarak gösterilmiştir.

Bölüm 2’de damgalama algoritmalarında kullanılan matematiksel dönüşümler incelenmektedir. Bölüm 3’te literatürdeki bazı damgalama algoritmaları incelenip, Bölüm 4’te kesirli Fourier dönüşümü (KFD) ile gerçekleştirilen damgalama algoritmaları tanıtılmaktadır. Son olarak, Bölüm 5’te performans analizleri yapıp, sonuçlar karşılaştırılmaktadır.

2. DİJİTAL DAMGALAMADA KULLANILAN DÖNÜŞÜMLER

Literatürde dijital damgalama en sık Ayırık Fourier Dönüşümü (AFD), Ayırık Kosinüs Dönüşümü (AKD), Dalgacık Dönüşümü (DD) ve Kesirli Fourier Dönüşümü (KFD)'yle gerçekleştirilmiştir. Bu tezde AFD, AKD ve KFD kullanılmıştır.

2.1 Ayırık Fourier Dönüşümü

Bir işaretin frekans uzayında ifadesi FD ile bulunabilir. 1-boyutlu sürekli bir işaretin sürekli zaman FD'si aşağıdaki şekilde elde edilir.

$$F(j\Omega) = \int_{-\infty}^{\infty} f(t)e^{-j\Omega t} dt \quad (2.1)$$

Sayısal ortamlarda kullanılan işaretler ayırık zamanlı olduğu için, sayısal ortamlarda yapılacak bir frekans spektrumu hesabında da FD'nin ayırık zamandaki gösterimine ihtiyaç duyulur. Sürekli bir işaretin belli aralıklarla örneklenmesi zaman değişkeni yerine $n\Delta t$ yazılmasıyla elde edilebilir. O halde,

$$F(j\Omega) \cong \sum_{-\infty}^{\infty} f(n\Delta t)e^{-j\Omega \Delta t} \Delta t \quad (2.2)$$

olarak yazılabilir. Ayırık işaretler için Δt ifadesi örnekleme periyodunu (T_s) göstereceğinden aşağıdaki eşitlik yazılabilir.

$$F(j\Omega) \cong \sum_{-\infty}^{\infty} f(nT_s)e^{-j\Omega T_s n} \quad (2.3)$$

Sürekli frekans işaretinin örnekleme periyoduyla çarpılmış halini, ayırık frekans olarak gösterirsek, $\Omega T_s = \omega$ eşitliğiyle (2.4) yazılabilir.

$$F(e^{j\omega}) \cong \sum_{-\infty}^{\infty} f(n)e^{-j\omega n} \quad (2.4)$$

Bu ifade ayırık zaman Fourier dönüşümü(AZFD) olarak tanımlanır ve ayırık işaretlerin sürekli Fourier uzayında ifade edilmesi için kullanılır. $F(e^{j\omega})$ ifadesi sadece sonlu sayıda da hesaplanabilir. Bunun için ifadenin sonlu sayıda örneklenmesi yeterlidir. Fonksiyondan alınacak eşit aralıklı örnek sayısı N ile gösterilebilir. $F(e^{j\omega})$ ifadesi 2π ile periyodik olacağı için örneklerin sadece bir periyot için alınması yeterlidir. O halde ayırık Fourier dönüşümü (AFD) şu şekilde ifade edilir.

$$F\left(e^{j\frac{2\pi}{N}k}\right) = \sum_{n=0}^{N-1} f(n)e^{-j\frac{2\pi}{N}kn} \quad (2.5)$$

Benzer şekilde iki boyutlu AFD'yi de şu şekilde yazmak mümkündür.

$$F[k, l] = \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} f[n, m] e^{-j\frac{2\pi}{N}kn} e^{-j\frac{2\pi}{N}lm} \quad (2.6)$$

Eğer $f(n, m) = f_1(n)f_2(m)$ şeklinde yazılabiliyorsa, yani ayrıştırılabilir bir fonksiyonsa, iki boyutlu fourier dönüşümü şu şekilde hesaplanabilir:

$$F[k, l] = F_1\left(e^{j\frac{2\pi}{N}k}\right) F_2\left(e^{j\frac{2\pi}{N}l}\right) \quad (2.7)$$

(Tavşanoğlu, Ders notları).

2.2 Ayırık Kosinüs Dönüşümü

AFD, karmaşık işareti karmaşık uzayda dönüştürür. Ancak işaret gerçel sayılardan oluşuyorsa, elde edilen dönüşümün yarısı kendini tekrar eder (İşaretin sanal kısmı sıfır ise spektrumda gerçel ve sanal kısım birbirinin aynı olur). Gerçel bir dönüşüm olarak Ayırık Kosinüs Dönüşümü (AKD), gerçel işareti gerçel spektruma dönüştürür ve AFD'deki tekrar probleminden kurtulur. AKD, AFD'den çıkarılabilir ve AFD'nin bütün özellikleri AKD için geçerlidir. Gerçel bir dizinin $\{x[m], (m = 0, \dots, N-1)\}$ AKD'si bulmak için öncelikle yeni bir işaret oluşturulur.

$$x'[m] \triangleq \begin{cases} x[m] & (0 \leq m \leq N-1) \\ x[-m-1] & (-N \leq m \leq -1) \end{cases} \quad (2.5)$$

Oluşturulan işaret, kendini $(-N \leq n \leq N-1)$ aralığı dışında da tekrar edecek şekilde tanımlanır. Başka bir deyişle, yeni dizi $x'(m)$, $2N$ periyoduyla periyodiktir ve $m = -\frac{1}{2}$ için simetriktr.

$$x'[m] = x'[-m-1] = x'[2N-m-1] \quad (2.6)$$

Eğer $x'(m)$ işareti, sağa ya da sola doğru $\frac{1}{2}$ kadar kaydırılırsa $x'(m) = x'(m' - \frac{1}{2})$ ifadesi $m' = 0$ için simetrik olur. $x'(m)$, $x(m)$ olarak ifade edilecek olunursa, $2N$ noktalı simetrik ifadenin AFD'si şu şekilde bulunur:

$$X[n] = \frac{1}{\sqrt{2N}} \sum_{m'=-N+\frac{1}{2}}^{N-\frac{1}{2}} x[m' - \frac{1}{2}] e^{-j2\pi m' n / 2N} \quad (n = 0, \dots, 2N-1) \quad (2.7)$$

m' ile $m + \frac{1}{2}$ yer değiştirilirse:

$$X[n] = \frac{1}{\sqrt{2N}} \sum_{m=-N+\frac{1}{2}}^{N-\frac{1}{2}} x[m'] - \frac{1}{2} \cos\left(\frac{2\pi m' n}{2N}\right) = \frac{1}{\sqrt{2N}} \sum_{m=0}^{N-1} x[m] \cos\left(\frac{(2m+1)n\pi}{2N}\right) = \sqrt{\frac{2}{N}} \sum_{m=0}^{N-1} x[m] \cos\left(\frac{(2m+1)n\pi}{2N}\right) \quad (n = 0, \dots, N-1) \quad (2.8)$$

elde edilir. (2.8)'in kosinüsler cinsinden yazılablmesinin sebebi $x(m)$ 'in gerçel ve simetrik olmasıdır. Aynı sebepten dönüşümden elde edilen değerlerin sadece ilk kısmı saklanır, ikinci kısmı simetrik olduğu için atılır.

Sonuç olarak elde edilen dönüşüm (2.9) gibi ifade edilebilir.

$$\sqrt{\frac{2}{N}} \sum_{m=0}^{N-1} x[m] \cos\left(\frac{(2m+1)n\pi}{2N}\right) = \sum_{m=0}^{N-1} c[n, m] x[m] \quad (n = 0, \dots, N-1) \quad (2.9)$$

Burada, m . dereceden satır ve n . dereceden sütun için dönüşüm matrisi:

$$c[n, m] \triangleq \sqrt{\frac{2}{N}} \cos\left(\frac{(2m+1)n\pi}{2N}\right) \quad (2.10)$$

olarak tanımlanır. Bu AKD matrisinin sütun vektörlerinin hepsi dikgen ve ilki hariç normaldir:

$$\sqrt{\sum_{m'=0}^{N-1} c^2[n, m]} = \sqrt{\sum_{m'=0}^{N-1} \frac{2}{N} \cos^2\left(\frac{(2m'+1)n\pi}{2N}\right)} = \begin{cases} \sqrt{2} & n = 0 \\ 1 & n = 1, 2, \dots, N-1 \end{cases} \quad (2.9)$$

O halde AKD dönüşümünü birim dikgen hale getirecek katsayı şu şekilde tanımlanabilir.

$$a(n) = \begin{cases} \sqrt{1/N} & n = 0 \\ \sqrt{2/N} & n = 1, 2, \dots, N-1 \end{cases} \quad (2.10)$$

Sonuç olarak dikgen AKD dönüşümü (2.13) gibi elde edilir:

$$X(n) = a(n) \sum_{m=0}^{N-1} x[m] \cos\left(\frac{(2m+1)n\pi}{2N}\right) = \sum_{m=0}^{N-1} x[m] c[n, m] \quad (n = 0, \dots, N-1) \quad (2.11)$$

Ters dönüşümü ise:

$$x(m) = \sum_{n=0}^{N-1} a(n)X(n) \cos\left(\frac{(2m+1)n\pi}{2N}\right) = \sum_{m=0}^{N-1} X[n]c[n, m]$$

$$(n = 0, \dots, N - 1)$$

(2.12)

olarak elde edilir (1).

2.3 Kesirli Fourier Dönüşümü

KFD, FD'nin bir genellemesi olan doğrusal bir dönüşümdür. KFD, n . dereceden FD'de n sayısının tamsayı olmaması durumu olarak düşünülebilir. Bu dönüşümle fonksiyon zaman – frekans uzayında döndürülür.

KFD birkaç farklı yoldan tanımlanabilir. Bu tanımlardan herhangi biri çıkış noktası olarak alınırsa öteki tanımları özellik olarak çıkarmak mümkündür. Bu tezde doğrusal integral dönüşüm tanımı gösterilmektedir. Çeşitli farklı tanımlar Özaktaş (2001) tarafından gösterilmiştir.

Derecesi α olan bir KFD, $f_\alpha(u)$ veya $F^\alpha f(u)$ şeklinde ifade edilir. Burada F^α , f işaretine etkiyen kesirli Fourier operatörüdür ve sonuç u uzayında gösterilir:

$$f_\alpha(u) \equiv F^\alpha f(u) \equiv (F^\alpha f)(u) \equiv F^\alpha[f](u) \equiv (F^\alpha[f])(u) \quad (2.13)$$

İkinci olarak $F^\alpha f(u)$ ifadesinde F^α , $f(u)$ işaretine etkiyen operatördür ve (2.15) gibi gösterilir:

$$f_\alpha(u) \equiv F^\alpha f(u) \equiv F^\alpha[f(u)](u) \equiv (F^\alpha[f(u)])(u) \quad (2.14)$$

İkinci tanım F^α operatörünün sistem ya da dönüşüm olmasından bağımsız olarak doğrudur ancak ilk tanım sadece F^α bir sistem gösterdiğinde doğrudur.

Doğrusal integral dönüşümü tanımında, KFD'nin doğrusal dönüşüm çekirdeği (kernel) tanımlanır. α . dereceden KFD, (2.16)'de gösterilen doğrusal işlemdir

$$f_\alpha(u) \equiv \int_{-\infty}^{\infty} K_\alpha(u, u') f(u') du'$$

$$K_\alpha(u, u') = A_\alpha e^{[j\pi(\cot\alpha \cdot u^2 - 2\csc\alpha \cdot uu' + \cot\alpha \cdot u'^2)]}$$

$$A_\alpha \equiv \sqrt{1 - j\cot\alpha}, \emptyset \equiv \frac{\alpha\pi}{2} \quad (2.15)$$

Dönüşüm tanımsal olarak doğrusaldır fakat fonksiyon sadece u ve ya u' değişkenlerini içermeyen için ötelemeyle değişmeyen (shift invariant) değildir ($\alpha = 4k$ dışında). α bir tamsayıyı ve k keyfi bir tamsayıyı belirtiyorsa, tanımsal olarak F^{4k} ve $F^{4k \pm 2}$ sırasıyla

özdeşlik operatörü (identity operator) I 'ya ve eşlenik operatörü (parity operator) P 'ye eşit olur. Mesela $\alpha = 1$ ise $\emptyset = \frac{\pi}{2}$ ve $A_\emptyset = 1$ bulunur, o halde:

$$f_1(u) = \int_{-\infty}^{\infty} e^{-j2\pi u u'} f(u') du' \quad (2.16)$$

olur. Burada $f_1(u)$ 'nin $f(u)$ fonksiyonunun FD'sine ($F(u)$) eşit olduğu görülür. Benzer bir şekilde $f_{-1}(u)$ 'in de $f(u)$ fonksiyonunun ters FD'sine eşit olduğu görülebilir. (2.16)'da $\emptyset = \frac{\alpha\pi}{2}$ trigonometrik fonksiyonların argümanlarında görüldüğü için, tanım α (ya da $\emptyset$) için periyodu 4 (ya da $\frac{\pi}{2}$) olacak şekilde periyodiktir. Bu sebepten dolayı genel olarak işlemlerde $\alpha \in (-2,2]$ (ya da $\emptyset \in (-\pi,\pi]$) ya da $\alpha \in [0,4)$ (ya da $\emptyset \in [0,2\pi)$) aralığı dikkate alınır. İfade, operatör notasyonu ile şu şekilde gösterilebilir:

$$F^0 = I, \quad (2.17)$$

$$F^1 = F \quad (2.18)$$

$$F^2 = P, \quad (2.19)$$

$$F^3 = FP, \quad (2.20)$$

$$F^4 = F^0 = I, \quad (2.21)$$

$$F^{4k+\alpha} = F^{4k'+\alpha}, \quad (2.22)$$

Burada k ve k' rastgele değişkenlerdir.

(2.16)'ya göre, tanımsal olarak $f(u)$ 'nin 0. dereceden dönüşümü kendisine eşittir. Benzer şekilde ± 2 . dereceden dönüşümü de $f(-u)$ 'ya tanımsal olarak eşittir. Bu parça parça tanımlar α 'nın bütün değerleri için süreklilik gösterir. Dönüşümün çekirdeği incelenecek olunursa α , 2'nin bir tamsayı katına yakın olmadığı durumlarda, α 'nın değerindeki ufak bir değişimin $f_\alpha(u)$ 'da da ufak bir değişime neden olacağı kolaylıkla görülebilir. Bunun doğruluğunu gözlemlemek için α 'yı 2'nin bir tamsayı katına yakın olduğu durumlarda, çekirdeğin $\alpha \rightarrow 0$ durumu dikkate alınır. Sonsuz küçük $|\alpha| > 0$ için çekirdek şu şekilde bir daha yazılabilir:

$$K_\alpha(u, u') = \frac{e^{-j\pi \sin(\emptyset)/4}}{\sqrt{|\alpha|}} e^{[j\pi(u-u')^2/\emptyset]} \quad (2.23)$$

Limit $\alpha \rightarrow 0$ durumu için bu ifade (2.16)'ya yakınsar.

$$K_{\alpha}(u) = \frac{e^{-j\pi \operatorname{sgn}(\alpha)/4}}{\sqrt{|\alpha|}} e^{[j\pi u^2/\alpha]}. \quad (2.24)$$

Buradan $f_{\alpha}(u)$, $K_{\alpha}(u) * f(u) = f_{\alpha}(u)$ şeklinde bulunur. $K_{\alpha}(u)$ 'nin FD'si, $\alpha \rightarrow 0$ için $e^{[j\pi \alpha u^2]} \rightarrow 1$ olacağından delta fonksiyonuna yakınsayacaktır. Yani dönüşümün tanımı $\alpha = 0$ civarında α ile doğrusal değişmektedir. Benzer bir şekilde ikinin diğer tamsayı katlarına yakınsandığında benzer özellik gözlemlenir.

Kesirli Fourier Dönüşümünün bir diğer önemli özelliği indeks toplama özelliğidir. Özellik, (2.26)'de alternatif gösterim şekilleriyle gösterilmiştir.

$$\begin{aligned} F^{\alpha_1}[F^{\alpha_2}[f]] &= F^{\alpha_1+\alpha_2}[f] = F^{\alpha_2}[F^{\alpha_1}[f]], \\ F^{\alpha_1}F^{\alpha_2}f &= F^{\alpha_1+\alpha_2}f = F^{\alpha_2}F^{\alpha_1}f, \\ F^{\alpha_1}F^{\alpha_2} &= F^{\alpha_1+\alpha_2} = F^{\alpha_2}F^{\alpha_1}. \end{aligned} \quad (2.25)$$

(2.16) ile bu eşitlikleri ispat etmek mümkündür. Gösterilmesi gereken ifade;

$$\int K_{\alpha_1}(u, u'') K_{\alpha_2}(u'', u') du'' = K_{\alpha_1+\alpha_2}(u, u') \text{ 'dir.} \quad (2.26)$$

Mesela bir fonksiyonun 0,3. dereceden kesirli Fourier dönüşümünün 0,6. dereceden kesirli Fourier dönüşümü 0,9. dereceden kesirli Fourier dönüşümüne eşit olur. İndeks toplama özelliği sayesinde α . dereceden bir KFD'nin tersi $(F^{\alpha})^{-1}$, $F^{-\alpha}$ 'ya eşit olacaktır ($F^{-\alpha}F^{\alpha} = I$). Yani α parametresini sanki FD operatörü F 'in katıymış gibi kullanılabilir.

KFD'nin bir diğer özelliği de Wigner dağılımıyla olan ilişkisidir. Bu noktada kısa zamanlı Fourier dönüşümü ve Wigner dağılımından da bahsetmek gerekir.

FD, bir işareti içinde barındırdığı frekans değerlerine göre frekans düzleminde gösterir. Ancak bu gösterimde bir frekansın hangi zamanda ortaya çıktığı bilgisi kaybolur. Kısa zamanlı Fourier dönüşümünde (KZFD) işaret, kısa zaman aralıklarına bölünerek Fourier dönüşümü alınır. Bulunan frekans dağılımları frekans-zaman ekseninde gösterilir ve zaman bilgisi kaybedilmemiş olur.

Bir $s(t)$ işaretinin, t zamanına ayarlı pencere fonksiyonu $h(t)$ ile filtrelenmesi şu şekilde gerçekleşir.

$$s_{\tau}(\tau) = s(\tau)h(\tau - t) \quad (2.28)$$

Pencere fonksiyonun kısa zaman aralığında $s(\tau)$ işaretini değiştirmemesi ve pencerenin dışındaki yerlerde ise 0 olması gerekir.

$$s_t(\tau) \cong \begin{cases} s(\tau) & t \text{ civarında} \\ 0 & t'den \text{ uzakta} \end{cases} \quad (2.29)$$

Pencerelenmiş işaret, işaretin t zamanındaki parçası olacağından bu işaretin fourier dönüşümü de bu zamandaki frekans dağılımını gösterecektir.

$$S_t(\omega) = \frac{1}{\sqrt{2\pi}} \int e^{-j\omega\tau} s_t(\tau) d\tau = \frac{1}{\sqrt{2\pi}} \int e^{-j\omega\tau} s(\tau) h(\tau - t) d\tau \quad (2.30)$$

Enerji yoğunluk fonksiyonu ise şu şekilde elde edilir.

$$P_{SP}(t, \omega) = |S_t(\omega)|^2 = \left| \frac{1}{\sqrt{2\pi}} \int e^{-j\omega\tau} s(\tau) h(\tau - t) d\tau \right|^2 \quad (2.31)$$

Her t değeri için farklı bir spektrum elde edilir ve bu spektrumlar zaman-frekans dağılımını meydana getirir (Cohen, 1994). Ancak KSFD'de kullanılan pencere fonksiyonu işarete bozulmaya yol açacaktır.

Wigner dağılımı (WD) 1932 yılında Wigner tarafından quantum mekaniği uygulamaları için önerilmiştir ve daha sonraları zaman-frekans analizi için güçlü bir araç olarak kabul edilmiştir. Wigner dağılımı işaretin enerjisinin zaman frekans uzayında dağılımı olarak yorumlanabilir. Claasen ve Mecklenbrauker (1980) Wigner dağılımının hem zaman işareti $f(t)$,

$$W(t, \omega) = \frac{1}{2\pi} \int s^*(t - \frac{1}{2}\tau) s(t + \frac{1}{2}\tau) e^{-j\tau\omega} d\tau \quad (2.32)$$

Hem de işaretin FD'sinden, elde edilebileceğini göstermişlerdir.

$$W(t, \omega) = \frac{1}{2\pi} \int S^*(\omega + \frac{1}{2}\theta) S(\omega - \frac{1}{2}\theta) e^{-j\theta t} d\theta \quad (2.33)$$

WD'de işaretin belirli bir zamandaki özelliklerine karar verebilmek için işaretin bu zamandan önceki ve sonraki kısmı karşılaştırılır. Eğer karşılaştırılan kısımlarda örtüşen noktalar varsa bu noktaların incelenen zamanda mevcut olacağı düşünülür (Cohen, 1994).

WD'de bulanıklaşma/ yayılma etkisinin az olacağı ve KZFD'ne göre çok daha iyi işaret zaman-frekans özelliklerini vereceği gösterilebilir (Cohen, 1966), (Cohen ve Margeneau, 1967). KZFD'de yüksek frekans çözünürlüğü ancak zaman çözünürlüğünden ödün verilerek elde edilirken, WD'de bu dezavantaj ortadan kaldırılmıştır. Ancak WD'de fonksiyonların iç çarpımından oluşan çapraz terim sorunu vardır.

Bir işaretin WD'si ile KFD'si arasında yakın bir ilişki vardır (Özaktaş vd., 2001). KFD açısı $\alpha = 0$ olan bir işaretin WD'si, işaretin WD'sinin α açısı kadar döndürülmüş haline eşittir.

Bu özellik tersi için de düşünülebilir; işaretin KFD'si, işaretin WD'nın $\emptyset$ açısı kadar döndürülmüş eksene olan integral izdüşümüdür.

KFD'nin hesaplanmasıyla ilgili, literatürde çeşitli çalışmalar mevcuttur. Özaktaş vd. (1996) tarafından hızlı KFD algoritması geliştirilmiş ve Kutay ve O'Neill tarafından farklı farklı gerçekleştirilmiştir. Bunun yanında Candan (1998) ve Pei vd. (1999) ayrık KFD algoritmaları geliştirilmiştir.

Tezde kullanılan algoritma Bultheel'in (2004, 2007) kullandığı ve (4)'de olan algoritmadır.

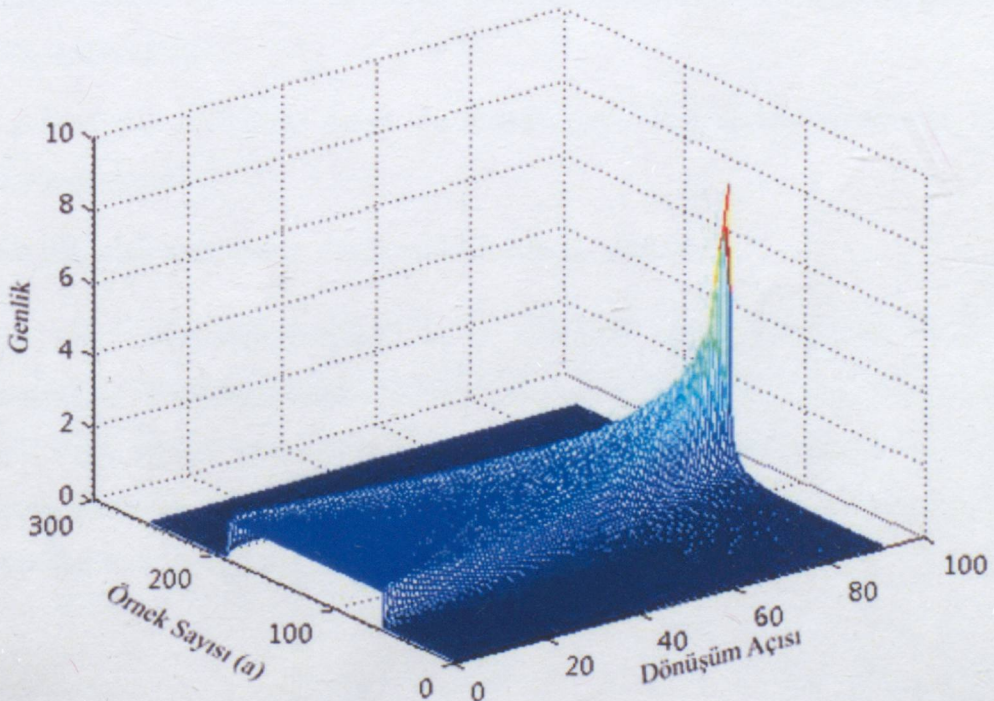
2-boyutlu KFD hesabı için KFD'nin aşağıdaki özelliği kullanılmıştır (Özaktaş, 2001).

$$F^{a_u, a_v} f(u, v) = \iint K_{a_u, a_v}(u, v; u', v') f(u', v') du' dv'$$

$$K_{a_u, a_v}(u, v; u', v') = K_{a_u}(u, u') \cdot K_{a_v}(v, v') \quad (2.34)$$

2-boyutlu KFD aynı zamanda (4), Bultheel (2007) ve (5)'de de aynı şekilde hesaplanmıştır.

Bir işaretin KFD ile kesirli Fourier dönüşüm uzayında α değiştikçe nasıl değiştiği Şekil 2.1'de gösterilmiştir. Basamak fonksiyonun $0 \leq \alpha \leq 1$ aralığında KFD'sine dikkat edilecek olunursa, $\alpha = 0$ için zaman uzayında, $\alpha = 1$ için Frekans uzayında olduğu görülür. $\emptyset$, 0'dan 1'e doğru değişirken işaret, zaman uzayından frekans uzayına doğru dönüşür.



Şekil 2.1 Basamak fonksiyonu.

2.4 Dalgacık Dönüşümü

DD'de işaret farklı frekanstaki bileşenlerine ayrılır ve her bir bileşen üzerinde ayrı olarak işlem yapılır. Buradaki amaç, yüksek ve düşük frekans bileşenlerini ayrı ayrı inceleyip, zaman-frekans çözünürlüğünü yüksek tutmaktır (1). Sürekli dalgacık dönüşümü:

$$\Psi_x^\Psi(\tau, s) = \frac{1}{\sqrt{|s|}} \int x(t) \Psi^*\left(\frac{t-\tau}{s}\right) dt \quad (2.35)$$

ile ifade edilir. (2.38) eşitliğinde görülen Ψ_x^Ψ dönüştürülmüş işaretinde τ ve s sırasıyla öteleme ve ölçekleme parametreleridir. Ana dalgacık olarak isimlendirilen $\Psi(t)$ fonksiyonu ise, dönüştüren fonksiyondur. Öteleme parametresi, KSFD'de kullanılan ve pencerenin yerini belirleyen parametreye benzer ve dönüşüm uzayında zaman bilgisine karşılık gelir. Ölçekleme parametresi, matematiksel olarak bir işareti ya sıkıştırır ya da yayar. Yüksek ölçekleme değerleri yayılmış, düşük ölçekleme değerleri ise sıkıştırılmış işaretlere karşılık gelmektedir (Strang ve Nguyen, 1996).

Bu tez kapsamında KFD kullanılarak damgalama yapılmış, gürültü ve alçak geçiren filtrelemeye karşı gürbüzlük performans analizleri irdelenmiştir.

3. DİJİTAL DAMGALAMA KAVRAMLARI VE ALGORİTMALARI

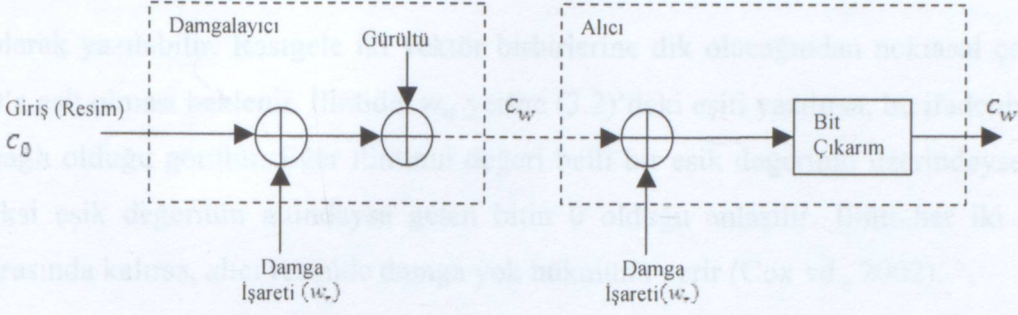
Dijital damgalamada, orijinal işaretin duyusal özellikleriyle, saldırılara karşı gürbüzlüğü ters orantılıdır. Dijital işaret, 2-boyutlu bir resim ise algısal olarak değişime uğramadan maksimum değiştirilebilmesini sağlayacak kuantalama matrisi Watson (1993) tarafından gösterilmiştir. Bu sebepten damgalama algoritmaları aslında bir optimizasyon problemidir. Amaç, c_0 'ı algısal olarak minimum değiştirip, saldırılara karşı maksimum gürbüzlüğü sağlamaktır.

Damgalama işleminde çeşitli senaryolar mümkün olabilir. Mesela, c_0 bir film ise ve film dağıtıcıları aracılığıyla dağıtılıyorsa, filmin yasadışı bir şekilde çoğaltılması durumunda, üretici firma hangi dağıtıcının filmi çoğalttığını öğrenmek isteyebilir. Böyle bir senaryoda her bir dağıtıcı filmini kendi damgasıyla damgalaması gerekir. Bir haber ajansı bir resmin telif hakkını korumak istiyorsa resmi kendi damgasıyla damgalamalıdır. Eğer damgalanacak işaret kâğıt üzerine basılacak bir resim ise damganın gürbüzlüğü çok önemli olacaktır. Damga, bir müzik grubunun dijital ortamdaki kaydına konacak ve yüksek sadakat teknolojisi ile kaydedilecekse damganın algısal olarak fark edilebilme özelliğinin minimum olması istenebilir. Farklı senaryolar Barni ve Bartolini'nin (2004) çalışmalarında bulunabilir. Çeşitli durumlarda çeşitli damgalama stratejileri belirlenmelidir. İyi bir damgalama algoritmasının çeşitli parametreleriyle oynanarak saldırılara karşı gürbüzlüğü ya da görsel bozulmalara karşı dayanıklılığı ayarlanabilir.

İlerleyen bölümlerde literatürde sıkça rastlanan temel damgalama kavramları ve çeşitli algoritmalar incelenecektir.

3.1 Gözü Kapalı Damgalama, Doğrusal İlintiliyle Algılama

Damgalama sisteminde alıcı tarafında c_0 'ın bilinmesi algılama işlemini büyük ölçüde kolaylaştırmaktadır. Basit bir çıkarma işlemiyle alınan resimle c_0 'ın farkına bakmak mümkün olur. Ancak çoğu zaman alıcı tarafta c_0 bilinmez. Bu şekilde gerçekleştirilen damgalama algoritmasına gözü kapalı damgalama denilir. Damganın ve resmin birbirinden bağımsız olduğu durumlarda damga ilintiyle algılanabilir (Cox vd., 2002).



Şekil 3.1 Gözü kapalı damgalama, doğrusal ilintiyle algılama.

3.1.1 Gözü Kapalı Damgalama, Doğrusal İlintiliyle Algılama Algoritması

Bu algoritmada bir bitlik mesaj resme yerleştirilir. Şifre olarak bir biti temsil eden, resmin büyüklüğünde normal dağılımlı bir gürültü oluşturulur. Bu gürültü referans örüntüsü (w_r) olarak adlandırılır. Mesaj bitinin değerine göre w_r , artı ya da eksiyle çarpılarak resme eklenir. Damgalanan resim (w_m) ile gösterilir. Buna göre:

$$w_m = \begin{cases} w_r & m = 1 \\ -w_r & m = 0 \end{cases} \quad (3.1)$$

olarak bulunur ve eğer mesaj işareti (m), işarete eklenen damga (w_a), damgalama gücü α ile gösterilirse damgalama işlemi şu şekilde gerçekleştirilir.

$$w_a = \alpha \cdot w_m \quad (3.2)$$

$$c_w = c_0 + w_a \quad (3.3)$$

Alıcı kısmına gelen c aslında,

$$c = c_0 + w_a + n \quad (3.4)$$

şeklinde ifade edilebilir. Alıcı kısmında mesaj, c 'den doğrusal ilinti yöntemiyle çıkarılır. Bu yöntemde alıcıya gelen resim, referans örüntüsüyle noktasal çarpılarak iki işaret arasındaki ilintiye bakılır.

$$z_{lc}(c, w_r) = \frac{1}{N} c \cdot w_r = \frac{1}{N} \sum_{x,y} c[x,y] \cdot w_r[x,y] \quad (3.5)$$

Eşitlik 3.4'de kullanıldığında, ilinti ifadesi

$$z_{lc}(c, w_r) = \frac{1}{N}(c_0 \cdot w_r + w_a \cdot w_r + n \cdot w_r) \quad (3.6)$$

olarak yazılabilir. Rastgele iki vektör birbirlerine dik olacağından noktasal çarpımlarının da 0'a eşit olması beklenir. İlintide, w_a yerine (3.2)'deki eşiti yazılırsa, bu ifadenin sadece w_r 'ye bağlı olduğu görülür. Eğer ilintinin değeri belli bir eşik değerinin üzerindeyse gelen bitin 1, eksi eşik değerinin altındaysa gelen bitin 0 olduğu anlaşılır. İlinti her iki eşik değerinin arasında kalırsa, alıcı resimde damga yok hükmünü verir (Cox vd., 2002).

$$m_n = \begin{cases} 1 & z_{lc}(c, w_r) > \tau_{lc} \\ \text{damga yok} & -\tau_{lc} \leq z_{lc}(c, w_r) \leq \tau_{lc} \\ 0 & z_{lc}(c, w_r) < -\tau_{lc} \end{cases} \quad (3.7)$$

3.1.2 Gözü Kapalı Damgalama, İlinti Katsayısıyla Algılama Algoritması

Bu alitmada gömme işleminden önce resim ikişer boyutlu, boyutları eşit bloklara (v) ayrılır. Elde edilen bu bloklar bir vektör uzayı oluşturur. Her bir blok, bu blok büyüklüğündeki bir w_r ile gözü kapalı gömme algoritmasında olduğu gibi damgalanır ve bloklar birleştirilerek damgalanmış resim elde edilir. Bu şekilde birden fazla damga aynı resim içine gömülebilir.

Algoritmanın alıcı kısmında ilinti katsayısı yöntemi kullanılır. Gelen resim (c) öncelikle damgalayıcı kısmında olduğu gibi boyutları eşit 2-boyutlu bloklara ayrılır. İlinti katsayısı yönteminde her bir bloğun ve bu bloğa karşı düşen w_r 'nin ilinti lerine bakılmadan önce aritmetik ortalaması kendilerinden çıkarılır ve normalize edilir (3.8). Bu işlem, iletim hattında meydana gelebilecek genlik değişimlerinden algoritmanın etkilenmemesini sağlayarak, gürbüzlüğü artırır. Buna göre ilinti katsayısı;

$$z_{cc}(v, w_r) = \frac{\tilde{v} \cdot \tilde{w}_r}{\sqrt{(\tilde{v} \cdot \tilde{v})(\tilde{w}_r \cdot \tilde{w}_r)}} \quad (3.8)$$

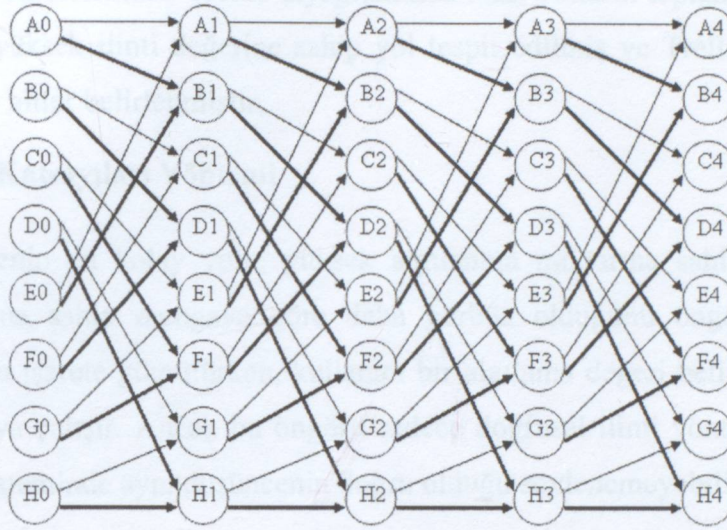
ile ifade edilir. Burada, $\tilde{v} = (v - \bar{v})$ ve $\tilde{w}_r = (w_r - \bar{w}_r)$ 'dir.

Normalizasyon işlemi yapıldığından, elde edilen ilinti miktarı $-1 \leq z_{cc}(v, w_r) \leq 1$ ile sınırlı kalacaktır. Bit değeri uygun eşik değerleri kullanılarak (3.7)'de olduğu gibi tespit edilir (Cox vd., 2002).

3.2 Trellis Algoritması Kullanarak Damgalama Yöntemi

Birden fazla bit aynı resim bloğunun içine gömülerek ilinti temelli algılayıcılarla tespit edilebilir. Her bir w_r , rastgele elde edilmiş gürültü işareti olduğu için bu işaretlerin birbirine

dik olması ve alıcıda birbirini etkilememesi beklenir. Bir bit dizisi kanaldan gönderilirken Trellis algoritması kullanılarak bu dizinin gürültüye karşı gürbüzlüğü artırılabilir.



Şekil 3.2 Trellis algoritması yapısı.

Trellis kodlaması, sınırlı kanal kapasitesine sahip kanallarda yüksek verimle veri iletimi sağlar (3). Kodlamanın özelliği, kanalda oluşabilecek hataların alıcı tarafından fark edilebilmesi ve düzeltilmesidir. Üst üste eklenen damga işaretleri her ne kadar rastgele üretildikleri için birbirlerine dik olması beklense de, gerçekte birbirlerini etkilemektedirler. Trellis kodlaması oluşabilecek hataların alıcıda düzeltilmesi için kullanılmıştır (Cox vd., 2002, 2004).

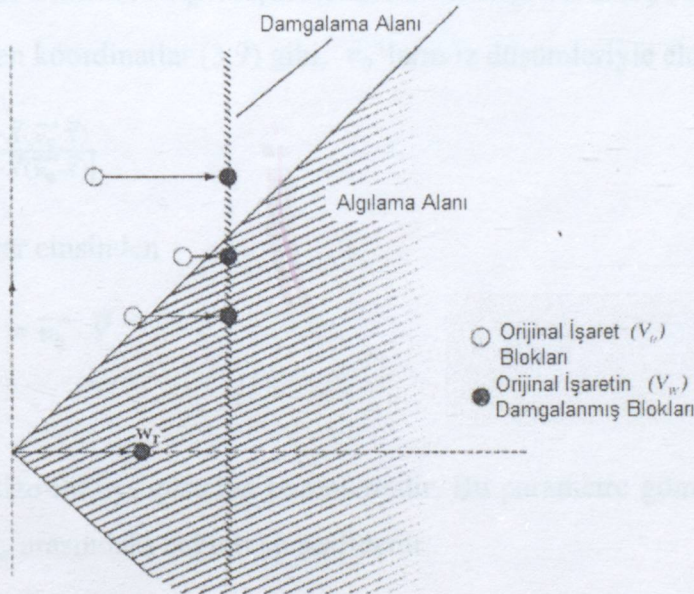
Trellis kodlamasında her bir bit Şekil 3.2’de gösterilen bir okla ifade edilir. Daireyle gösterilmiş bölümler bit dizisinin o anki durumunu gösterir. Açık renkli oklar 0 bitini koyu renkli oklar 1 bitini temsil eder. Örneğin mevcut durum C1 ise ve gelen bit 0 ise sonraki durum E2, gelen bit 1 ise sonraki durum F2 olacaktır. Trellis algoritması damgalama işlemine şu şekilde uyarlanmıştır. Her bir yol için ayrı bir w_r yaratılır. Mesaja göre uygun w_r ’ler seçilip toplanarak damga işareti elde edilir. Mesajdaki bit değeri 0 ise Şekil 3.2’de açık renkli oklara karşı düşen w_r ’ler, bit değeri 1 ise koyu renkli oklara karşı düşen w_r ’ler seçilir. Uygulamada koyu ve açık renkli oklara karşı düşen ve aynı durumdan çıkan oklar için seçilecek w_r ’ler, aralarındaki vektörel uzaklığın maksimum olmasını sağlamak amacıyla birbirlerinin eksilisi olacak şekilde seçilmiştir.

Uygulamada ilk durum A0 olarak seçilmiştir. Gelen bit 0 ise ikinci durum A1 ve bu yol için seçilen damga A0’dan A1 durumuna geçen w_r ’yi temsil eden w_{rA0A1} olmuştur. Eğer gelen bit 1 ise sonraki durum B1 ve seçilen yola karşı gelen damga işareti w_{rA0B1} olmuştur.

Alıcı tarafında mesajın çözümlenmesi Viterbi algoritması tabanlı gerçekleşir. Durumlar arası her bir olası ilinti hesaplanmış ve bu ilinti değerleri $z[A..H]$ ilinti vektöründe saklanmıştır. Elde edilen ilinti vektörlerinden Trellis diyagramında olası yolların toplam ilinti büyüklükleri bulunmuştur. En yüksek ilinti değerine sahip yol tespit edilmiş ve Trellis diyagramında bu yola karşılık gelen bitler belirlenmiştir.

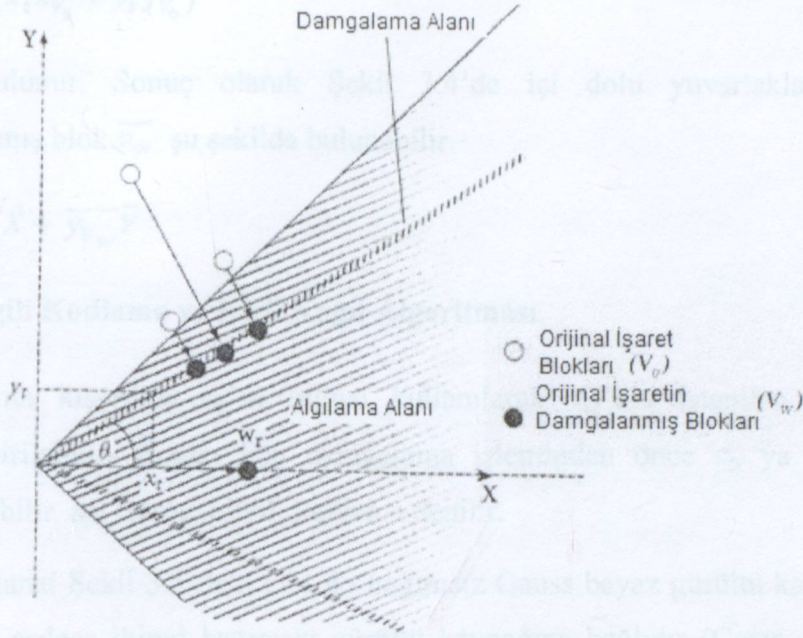
3.3 Sabit İlinti Katsayıları Yöntemi

Gümbüzlüğü ölçmenin en kolay yolu, yüksek algılanma miktarına sahip damganın düşük algılanma miktarına sahip damgaya göre daha gümbüz olduğunu öngörmektir. Sabit bir gümbüzlükle damga işarete gömülürken, kullanıcı bir algılama değeri belirler ve damgalayıcı bu değere ulaşmaya çalışır. Ancak bu öngörü sadece doğrusal ilinti yöntemi için geçerlidir. Normalize ilinti sisteminde aynı düşüncenin doğru olduğu söylenemeyebilir.



Şekil 3.3 Normalize ilinti (Cox vd., 2002).

Doğrusal ilinti modelinde (3.6)'dan ötürü her bir v_w ve w_r arası ilinti sabit bir değere eşit olacaktır. Bu sebepten damgalama alanı dar bir banda denk gelir. Fakat normalize ilinti alıcısının algılama alanı taralı kısımdır. Şekil 3.3'te de görülebildiği üzere her bir v_w uygun bir şekilde algılayıcıda algılanamamaktadır. Bu problemi çözmek için Gram-Schmidt dikgenleştirme kullanılarak damgalanmamış vektörler (v_0) için iki dikgen koordinat eksenı bulunur. Gömme işlemi daha sonra koordinatlara ve uygun gömme gücüne ve gömme açısına göre gerçekleştirilir. Bu gömme stratejisi Şekil 3.4'de gösterilmiştir.



Şekil 3.4 Gram-Schmidt dikgenleştirilmesinin kullanıldığı durumda (Cox vd., 2002).

Her bir v_0 için dikgen koordinatlar (3.9) gibi, v_0 'ların iz düşümleriyle elde edilebilir:

$$\vec{X} = \frac{w_r}{|w_r|} \text{ ve } \vec{Y} = \frac{\vec{v}_0 - \vec{X}(\vec{v}_0 \cdot \vec{X})}{|\vec{v}_0 - \vec{X}(\vec{v}_0 \cdot \vec{X})|} \quad (3.9)$$

Her v_0 bu koordinatlar cinsinden

$$x_{v_0} = \vec{v}_0 \cdot \vec{X} \text{ ve } y_{v_0} = \vec{v}_0 \cdot \vec{Y} \quad (3.10)$$

gibi ifade edilebilir.

t_{nc} istenilen normalize ilintiyi gösteren parametredir. Bu parametre gömme alanının açısıyla alakalıdır. Açıyla t_{nc} arasındaki bağıntı şu şekildedir:

$$\theta = \cos^{-1}(t_{nc}) \quad (3.11)$$

O halde;

$$x_t = t_{nc} \text{ ve } y_t = \sqrt{1 - t_{nc}^2} \quad (3.12)$$

olur. Birim vektörler olarak x_t ve y_t ile tanımlanmış doğruda $\vec{v}_0$ 'a en yakın nokta,

$$x_{V_w} = x_t(x_t x_{V_0} + y_t y_{V_0})$$

$$y_{V_w} = y_t(x_t x_{V_0} + y_t y_{V_0}) \quad (3.13)$$

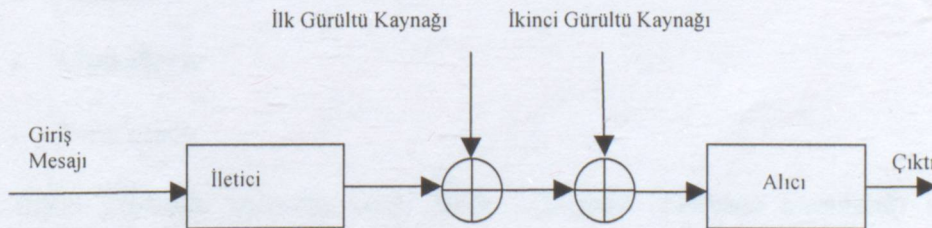
olarak bulunur. Sonuç olarak Şekil 3.4'de içi dolu yuvarlaklar olarak gösterilmiş damgalanmış blok $\vec{v}_w$ şu şekilde bulunabilir.

$$\vec{v}_w = \vec{x}_{V_w} \vec{X} + \vec{y}_{V_w} \vec{Y} \quad (3.14)$$

3.4 Bilgili Kodlama ve Kirli Kağıt Algoritması

Damgalayıcı kısmında c_0 'ın bilgisi kullanılarak c_w 'nin istenilen özellikleri sağlaması gerçekleştirilebilir. Bunun için damgalama işleminden önce c_0 ya da kullanılan damga değiştirilebilir. Bu işleme bilgili kodlama denilir.

Eğer bir kanal Şekil 3.5'deki gibi iki bağımsız Gauss beyaz gürültü kaynağına sahipse kanal kapasitesi sadece ikinci bağımsız gürültü kaynağına bağlıdır (Costa, 1983). Şekil 3.5'deki yapı damgalayıcı yapısına oldukça benzemektedir. İlk gürültü kaynağı c_0 olarak düşünülürse, damgalama işleminde c_0 'ın bilinmesi kanal kapasitesinde hiçbir azalmaya yol açmayacaktır. Costa bu kanalı kirli bir kâğıt üzerine yazı yazmaya benzetmiştir. Kâğıt üzerindeki kir normal dağılımlı bir örüntü olsun. Bu kir, gönderici tarafından incelenebilen ilk gürültü kaynağıdır. Göndericinin bu kâğıt üzerine not yazdığı ve alıcıya gönderdiği düşünüldüğünde, kâğıt yol boyunca ikinci gürültü kaynağına denk gelen bir kirden etkilenebilir. Costa, alıcının, kiri ve mürekkebi birbirinden ayırt edememesi durumunda ilk gürültü kaynağının kanal kapasitesinde hiçbir etkisi olmadığını göstermiştir. İspatı, her bir mesajın, kod tablosunda yer alan çeşitli farklı işaretlere denk gelmesine dayanır. Bu analogiye referansla böyle bir kod tablosuna kirli-sayfa kodları denilmektedir (Miller, 2001).



Şekil 3.5 Costa'nın kanalı.

Kirli sayfa kodları kullanılarak mesaj gönderilmesi için, gönderici aşağıdaki adımları izler. Kod kümesi (W) olmak üzere,

- Mesajı temsil eden kod tablosu bulunur. $W_m \subset W$
- W_m içinde c_0 'a en yakın olan işaret bulunur ve ilk gürültü kaynağı olarak eklenir.

- c_w iletilir.

Alıcı tarafında ise şu adımlar izlenir:

- Bütün kod tablosu aranarak en yakın kod bulunur.
- Kodun temsil ettiği mesaj alınan işaret olarak belirlenir.

Moulin ve O'Sullivan, (2000) Costa'nın çalışmasını daha gerçekçi damgalama algoritmaları için geliştirmişlerdir.

Uygulamada 0 ve 1 bitlerini sembolize eden iki w_r kümesi yaratılmıştır. Kümelere sırasıyla W_0 ve W_1 denilmiştir. Her bir w_r rastgele üretilmiştir. Resim bloğuna 0 biti gömülmesi durumunda, gömücü ilgili resim bloğuyla W_0 kümesi elemanlarının ilintilerine sırasıyla bakar. En yüksek ilintiye sahip w_r seçilerek damgalama işlemi gerçekleştirilir. 1 biti için de aynı işlemler geçerli olup tek fark w_r 'nin W_1 kümesinden seçilecek olmasıdır.

Alıcı kısmında, damgalanan blok ile bütün W_0 ve W_1 kümelerinde bulunan w_r 'lerin ilintilerine bakılır. En yüksek ilintiye sahip w_r seçilir ve bu w_r 'nin ait olduğu küme belirlenir. Eğer w_r , W_0 kümesindeyse gelen bit 0, W_1 kümesindeyse gelen bit 1 olduğuna karar verilir.

3.5 Watson'un AKD Tabanlı Görsel Modeli

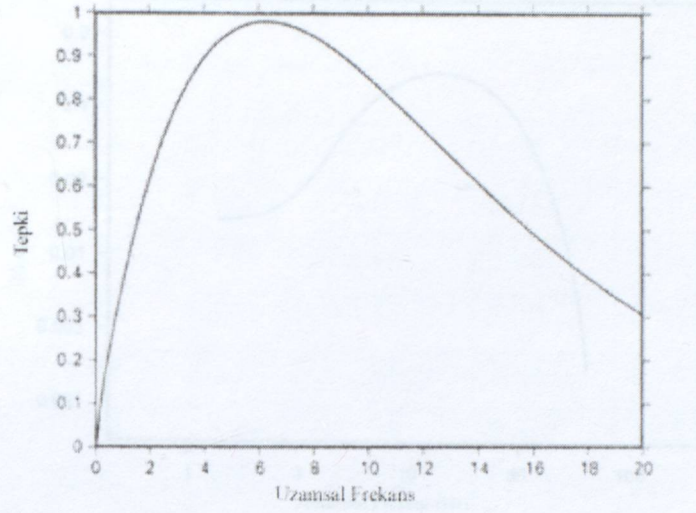
İnsan görsel sistemi uzamsal frekans, parlaklık ve renk girdilerine değişik tepkiler verir. Algısal model, bir damganın her bölgesinin aynı ölçüde algılanabilir olmadığını gösterir.

Algısal modelle genel olarak üç temel etki incelenir. Bu etkileri şu şekilde sıralamak mümkündür:

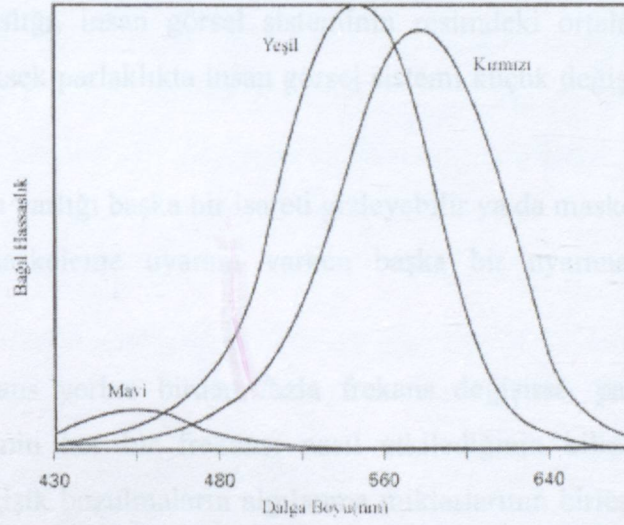
- Hassaslık
- Maskeleye
- Grup etkisi

Hassaslık; insan gözünün uyarana karşı direk tepkisidir. Frekans hassaslığı ve parlaklık hassaslığı olarak iki bölümde incelenebilir.

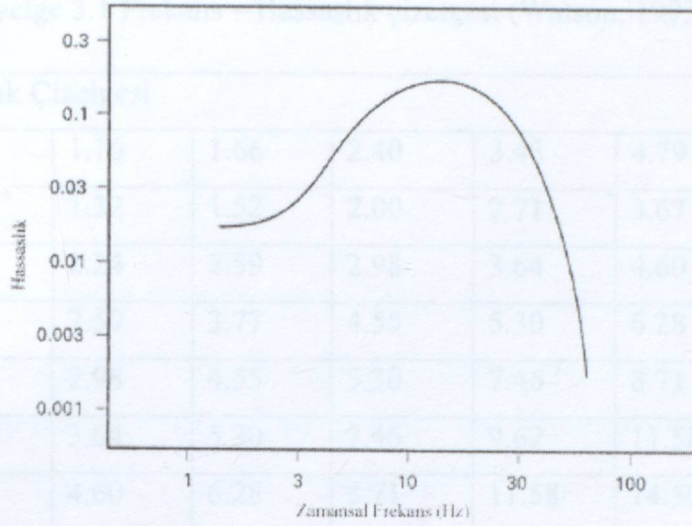
Frekans hassaslığı denildiğinde görmede üç çeşit frekans cevabı düşünülebilir. Biri uzamsal frekans; hassaslık ya da parlaklık maskeleyesi, biri renkler olarak algılanan ışık tayfi frekansı ve biri de hareket ya da titreme olarak algılanan zamansal frekanstır. Bu üç uyarana karşı gözün tepkisi sırasıyla Şekil 3.6, Şekil 3.7 ve Şekil 3.8'da gösterilmiştir.



Şekil 3.6 Gözün uzamsal frekans tepkisi (Cox vd., 2002).



Şekil 3.7 Gözün ışık tayfı frekanslarına tepkisi (Cox vd., 2002).



Şekil 3.8 Gözün zamansal frekansa tepkisi (Cox vd., 2002).

Şiddet parlaklık hassaslığı, insan görsel sisteminin resimdeki ortalama parlaklığına göre değişmesidir. Daha yüksek parlaklıkta insan görsel sistemi küçük değişimleri daha kolay fark eder.

Maskeleme: Bir işaretin varlığı başka bir işareti gizleyebilir ya da maskeleyebilir. Maskeleme, insan gözünün bir maskeleme uyarını varken başka bir uyarana olan cevabı olarak tanımlanabilir.

Grup Etkisi: Bir frekans yerine birden fazla frekans değişirse, parlaklık hassasiyetiyle, maskeleme hassasiyetinin her bir frekansı nasıl etkilediğinin bilinmesi gerekir. Algısal uzaklık modelinde, değişik bozulmaların algılanma miktarlarının birleştirilmesiyle resimdeki genel algısal değişimi gösteren genel tek bir değer elde edilebilir.

Watson'un AKD Tabanlı Görsel Modelinde c_0 , 8×8 'lik bloklara ayrılarak $c[i, j, k]$ 'lar elde edilir. Burada "i" ve "j" piksel numarasını, "k" ise blok numarasını göstermektedir. Bu modelde hassaslıkla ilgili bir frekans hassaslık çizelgesi tanımlanmıştır. Çizelge 3.1'de, genlik değerinin düşük olduğu hücreler, gözün o frekansa daha hassas olduğunu göstermektedir.

Çizelge 3.1 Frekans – Hassaslık çizelgesi (Watson, 1993)

Frekans Hassaslık Çizelgesi							
1.40	1.01	1.16	1.66	2.40	3.43	4.79	6.56
1.01	1.45	1.32	1.52	2.00	2.71	3.67	4.93
1.16	1.32	2.24	2.59	2.98	3.64	4.60	5.88
1.66	1.52	2.59	3.77	4.55	5.30	6.28	7.60
2.40	2.00	2.98	4.55	5.30	7.46	8.71	10.17
3.43	2.71	3.64	5.30	7.46	9.62	11.58	13.51
4.79	3.67	4.60	6.28	8.71	11.58	14.50	17.29
6.56	4.93	5.88	7.60	10.17	13.51	17.29	21.15

Parlaklık Maskelemesi için 8×8'lik blokların ortalama parlaklıklarına bakılır. Fazla parlaklıkta, algısal olarak bozulmadan AKD katsayıları daha fazla değiştirilebilir. Bu yüzden Watson modeli hassaslık tablosunu her bir bloğun DC terimini hesaba katarak yeniden hesaplar. Parlaklık maskelemesi eşik değeri bu durumda şu şekilde bulunabilir:

$$t_L[i, j, k] = t[i, j] \times \left(\frac{C_0[0,0,k]}{C_{00}} \right)^{\alpha_T} \quad (3.15)$$

Burada α_T bir sabittir ve önerilen değeri 0.649'dur. C_{00} resimdeki AKD katsayılarının ortalamasıdır.

Watson modelinde kontrast maskelemesi tanımlanır. Parlaklık maskelemesi eşiği kontrast maskeleme eşiğini etkiler. Kontrast maskeleme ile birlikte bir maskeleme eşiği elde edilir. Maskeleme eşiği($s[i, j, k]$) şu şekilde elde edilir:

$$s[i, j, k] = \max\{t_L[i, j, k], C_0[i, j, k]^{w[i, j]} t_L[i, j, k]^{1-w[i, j]}\} \quad (3.16)$$

Burada $w[i, j]$, 0 ile 1 arasında bir sabittir ve her bir frekans katsayısı için farklı olabilir. Watson her bir i ve j değeri için $w[i, j] = 0,7$ kullanmıştır.

Grup Etkisinde orijinal resim ile bozulmuş (damgalanmış) resmi karşılaştırmak için bu iki resmin AKD katsayıları arasındaki fark hesaplanır.

$$e[i, j, k] = C_w[i, j, k] - C_0[i, j, k] \quad (3.17)$$

Resmin algısal bozulmasının uzaklığı,

$$d[i, j, k] = \frac{s[i, j, k]}{s[i, j, k]} \quad (3.18)$$

olarak hesaplanır. Tek bir algısal bozulma değeri de şu şekilde elde edilir:

$$D_{wat}(c_0, c_w) = \left(\sum_{i,j,k} |d[i, j, k]|^p \right)^{\frac{1}{p}} \quad (3.19)$$

Eşitlikte tavsiye edilen p değeri 4'dür (Watson, 1993).

Uygulamada algısal model gözü kapalı damgalama algoritmasına uyarlanabilir. (3.2)'deki α parametresinin görsel bir bozulmaya sebep olmadan değişmesi istendiğinde, doğrusal bir dönüşüm olan AKD için (3.20)'yi yazmak mümkün olur.

$$c_w = c_0 + \alpha W_m \quad (3.20)$$

Grup etkisi fonksiyonuna göre algısal uzaklığı şu şekilde bulmak mümkündür.

$$\begin{aligned} D_{wat}(c_0, c_w) &= \sqrt[4]{\sum_{i,j,k} \left(\frac{c_w[i,j,k] - c_0[i,j,k]}{s[i,j,k]} \right)^4} = \sqrt[4]{\sum_{i,j,k} \left(\frac{\alpha W_m[i,j,k]}{s[i,j,k]} \right)^4} = \\ &\alpha \times \sqrt[4]{\sum_{i,j,k} \left(\frac{W_m[i,j,k]}{s[i,j,k]} \right)^4} = \alpha \times D_{wat}(c_0, c_0 + w_m) \end{aligned} \quad (3.21)$$

Damgalama için uygun α parametresi şu şekilde elde edilir.

$$\alpha = \frac{D_{target}}{D_{wat}(c_0, c_0 + w_m)} \quad (3.22)$$

4. KFD UZAYINDA İLİNTİ TABANLI DAMGALAMA ALGORİTMALARI

Uzamsal uzaydaki damgalama algoritmaları daha çok görsel bozulmaya yol açmaktadır. Bu sebepten literatürde kullanılan çoğu damgalama algoritmasında damgalama işlemi AFD uzayında gerçekleştirilmekte ve damga alçak-orta frekanslardaki katsayılarla yerleştirilmektedir (Solachidis ve Pitas 2007), (Ping vd. 2005), (Wang vd. 2005, 2006).

Resmin alçak frekanslarının damgalanması görsel açıdan resmi fazla bozmaktadır. Yüksek frekanslara yapılan damgalama işlemleri de alçak geçiren filtreleme gibi saldırılara karşı oldukça savunmasızdır. Frekans uzayındaki bu sınırlamalar farklı dönüşüm uzaylarında ne kadar verim alınabileceği sorusunu gündeme getirir.

Literatürde son yıllarda geliştirilmiş az sayıda KFD tabanlı damgalama çalışmaları yer almaktadır. Djurovic vd. (2000) damgalanacak görüntünün KFD katsayılarını belli bir α_x ve α_y değeri için belirlemiş ve bu katsayıları frekans uzayında sıralamıştır. Orta frekans aralığında seçilen KFD katsayıları, damga olarak seçilen Gauss biçimli beyaz gürültü tarafından damgalanmış ve damga algılaması bilgili algılama modeliyle belirlenmiştir. Stankovic vd. (2001) doğrusal frekans değişimi olan bir çörp işaretini damga olarak kullanmıştır. Bu tip bir işaretin her türlü filtreleme tekniğine karşı gürbüz olduğunu düşünülmüştür. Damgalanmış görüntünün Wigner dağılımının projeksiyonlarında damganın maksimum değeri, orijinal resimden kaynaklanan dağılımdan daha yüksek olacağını düşünülmüştür. Çalışmada iki türlü damgalama önerilmiştir. İlk türde iki çörp işareti arasındaki uzaklık farkı bilinmekte ve her türlü doğrusal görüntü işleme uygulamasından sonra bu farkın sabit kalacağı öngörülmektedir. Damga bu iki işaretin bağlı uzaklığı olarak tanımlanmaktadır. Öteki uygulamada farklı genliklerdeki ve farklı yerlerdeki çeşitli çörp işaretleriyle damgalama gerçekleştirilmiştir. Bu algoritmanın alıcı kısmında her çörp işaretinin yeri ve genlik bilgisi gerekmektedir. Altun vd., (2006) dışbükey kümelerle izdüşümü (Projection onto convex sets, POCS) tabanlı yeni bir bilgili damgalama sistemini KFD kullanılarak damgalama işlemi için özelleştirmiştir. KFD kullanılmasıdaki ana neden, $\emptyset$ parametresinin ekstra bir serbestlik derecesi sağlaması ve saldırılara karşı daha çok gürbüzlük elde edebilmek. Yu vd., (2006) çörp işaretini damga olarak kullanmış, belli bir tarama hızına sahip çörp işaretini orijinal işarete eklemiştir. Alıcıda KFD açısı, çörp tarama hızıyla aynı seçilmiş ve çörp işaretin yüksek bir çıkış üreteceği düşünülmüştür.

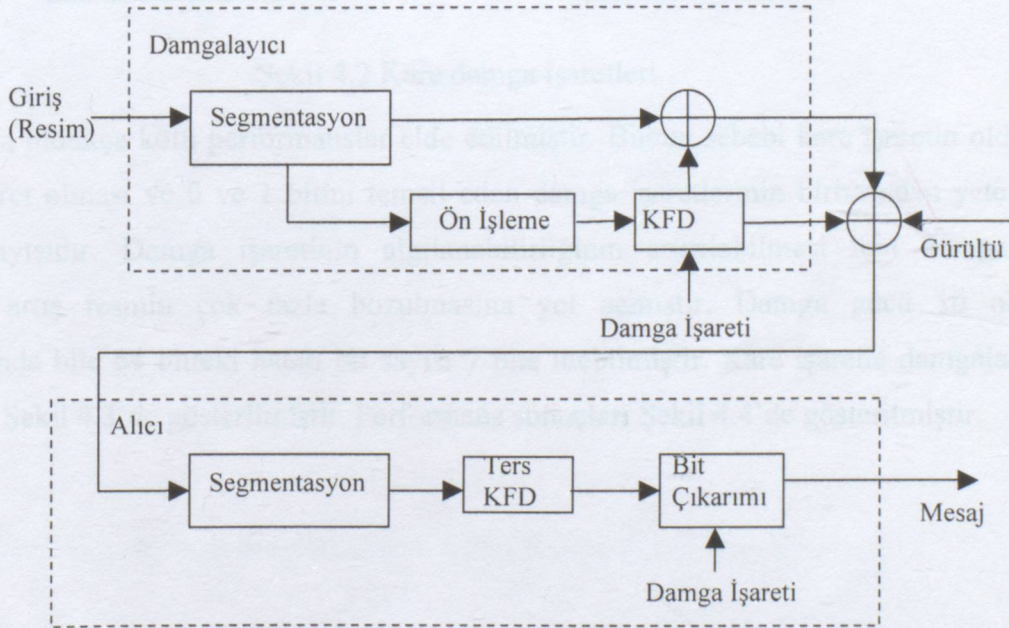
Bu tezde KFD kullanılarak blok tabanlı damgalama çalışmaları yapılmıştır. Damga olarak 2-boyutlu kare, çörp ve dalga katarı kullanılmıştır. Geliştirilen damgalama algoritmalarındaki

ana düşünce, damga işaretini belli bir KFD ile görsel olarak daha az fark edilebilir hale getirmek ve resme eklemektir. Alıcı kısmında dönüştürme ters çevrilerek, damganın daha kolay fark edilebilmesi sağlanmıştır.

4.1 KFD uzayında Kare İşaret Kullanarak İlinti Tabanlı Damgalama Algoritması Uygulaması.

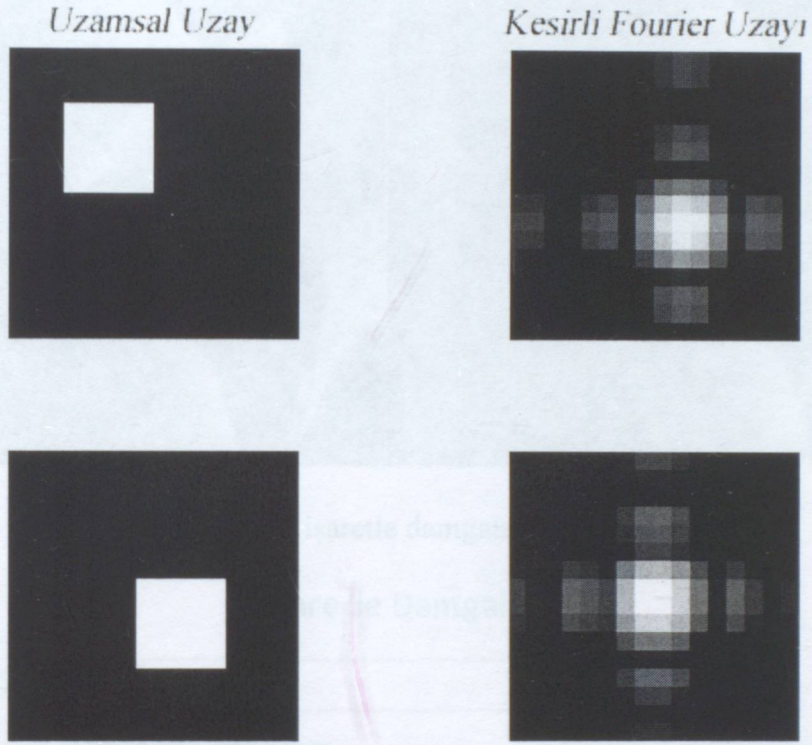
Bu uygulamada 2-boyutlu kare işaret damga olarak kullanılmıştır. Oluşturulan kare işaret, KFD ile belli açılarda döndürülmüş ve resmin segmentasyon sonrası elde edilen bloklarına eklenmiştir. Ön işleme kısmında damgalama sonrası algoritmanın performansına Watson'un KFD modeli kullanılarak bakılmış ve en yüksek performansı gösteren KFD açısı olarak seçilmiştir. Alıcı kısmında damgalanmış işaret ön işlemede seçilen aynı KFD açısıyla geri döndürülmüş ve mesaj ilinti tabanlı çıkarılmıştır.

Doğa resimleri genellikle alçak frekans bileşenleri içerdiği için, 2-boyutlu KFD sonucunda kesirli Fourier derecesi 90° 'ye yaklaştıkça enerji düşük frekanslarda toplanacaktır. Kesirli Fourier uzayında merkezde toplanacak bu enerjinin üzerine kare işaret eklenirse alıcıda yüksek ilintiyle çözümlenebileceği, uzamsal uzayda da resme çok etki etmeyeceği düşünülmüştür. Damgalama işleminin blok diyagramı Şekil 4.1'de gösterilmektedir.



Şekil 4.1 Damgalama algoritması.

Bu algoritma için kullanılan kare damga işaretleri Şekil 4.2’de gösterilmiştir. Bitleri temsil edecek damgalar arasındaki fark, iki işaretin yerlerinin farklı seçilmesiyle sağlanmıştır. Frekans uzayında kare, sinc işaretine dönüşecektir. Damgaya 90° ’ye yakın derecede KFD uygulanacağı için resmin her bir bloğu sanki bir sinc işaretiyle toplanmış gibi olacaktır. Şekil 4.3’de damgalanmış resim görülebilir.



Şekil 4.2 Kare damga işaretleri.

Uygulamada oldukça kötü performanslar elde edilmiştir. Bunun sebebi kare işaretin oldukça kısa bir işaret olması ve 0 ve 1 bitini temsil eden damga işaretlerinin birbirinden yeterince farklı olmayışındır. Damga işaretinin algılanabilirliğinin artırılabilmesi için damgalama gücündeki artış resmin çok fazla bozulmasına yol açmıştır. Damga gücü 10 olarak ayarlandığında bile 64 bitteki hatalı bit sayısı 7 bite inebilmiştir. Kare işaretle damgalanmış Lena resmi Şekil 4.3’de gösterilmiştir. Performans sonuçları Şekil 4.4’de gösterilmiştir.

4.2. İkinci Tabanlık, Damga İşareti Olarak Çarpılarak Kullanarak Damgalama

Kare işaretin düşük frekanslı performansı oldukça düşük olduğu için daha fazla frekanslı damga işaretleri kullanılmıştır. Uzamsal uzayda yüksek dereceli sinc ve frekans uzayında kare işareti kullanılmıştır. İkinci tabanlık, damga işareti olarak kare işareti kullanılmıştır. İkinci tabanlık, damga işareti olarak kare işareti kullanılmıştır.

Doğrusal çörp, anlık frekansın zamanla doğrusal olarak değiştiği bir işarettir. Basit bir çörp işareti (4.1) gibi tanımlanabilir.

$$\begin{aligned}
 s(t)|_{nT_s} &= e^{j2\pi(f_2 nT_s + \mu(nT_s)^2)} \\
 s(n) &\cong e^{j2\pi(f_2 nT_s + \mu(nT_s)^2)} \\
 \mu &= \frac{f_2 - f_1}{2}
 \end{aligned} \tag{4.1}$$

Burada f_2 doğrusal çörp işaretinin maksimum frekansını, f_1 doğrusal çörp işaretinin minimum frekansını gösterir. Anlık frekans işaretin fazının türevidir.

$$\begin{aligned}
 \varphi' &= (2\pi(f_1 t + \mu t^2))' = 2\pi(f_1 + 2\mu t) \\
 \varphi' &= (2\pi(f_1 nT_s + \mu(nT_s)^2))' = 2\pi T_s(f_1 + 2\mu T_s n)
 \end{aligned} \tag{4.2}$$

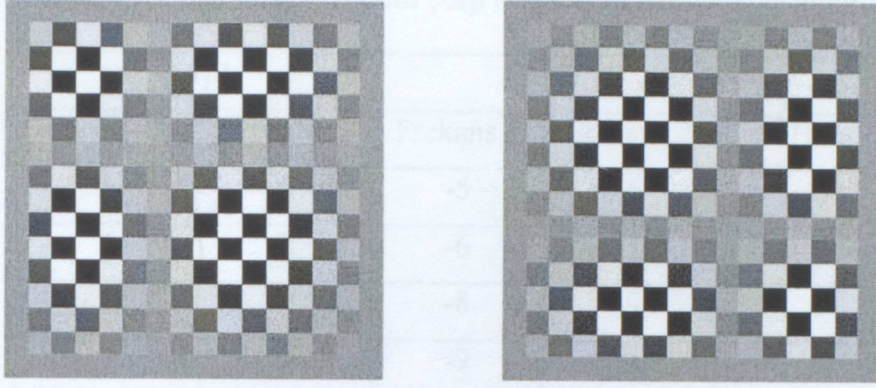
Burada işaretin anlık frekans değişiminin doğrusal olduğu görülür.

Kullanılan 2-boyutlu damga ve uygulanan KFD derecesi damgalama işleminin şifreleridir. Belli başlangıç ve bitiş frekanslarına sahip çörp işaretleri yaratılmış ve bunların başlangıç ve bitiş frekansları ters çevrilerek öteki biti tanımlayan ikinci çörp işareti yaratılmıştır. Düşük frekanstan yüksek frekansa çıkan bir çörp işaretiyle yüksek frekanstan düşük frekansa düşen bir çörp işaretinin birbirinden büyük ölçüde farklı olacağı düşünülmüştür. Elde edilen çörp çiftlerinde iyi ve kötü performans veren damga çiftleri elde edilmiştir. Çizelge 4.1'de iyi performans gösteren çörp çiftleri ve hata sayıları gösterilmiştir. Bu değerler rastgele bir bit dizisi için 64 bitteki hata sayılarını göstermektedir. Damgalama gücü 5 seçilmiştir. Dönüşüm açısı da 30° seçilmiştir.

Çizelge 4.1 Çörp performansları

Çörp1		Çörp2		
Frekans 1	Frekans 2	Frekans 1	Frekans 2	Hata
5	6	6	5	3
6	7	7	6	0
8	9	9	8	0
9	10	10	9	1
5	7	7	5	0
6	8	8	6	2
7	9	9	7	4
8	10	10	8	0
5	8	8	5	5
7	10	10	7	3
5	9	9	5	0
6	10	10	6	0
4	9	9	4	0
6	11	11	6	0
2	10	10	2	1
3	11	11	3	5
4	12	12	4	2
5	13	13	5	2

Bu damgalama şartlarında sıfır hatayla çalışan başlangıç frekansı 6 bitiş frekansı 7 ve başlangıç frekansı 7, bitiş frekansları 6 olan çörplerin görüntüleri Şekil 4.5’de gösterilmiştir.



Şekil 4.5 Başlangıç frekansı 6, bitiş frekansı 7 olan çörp işareti ve frekansları ters çevrilmiş eşi.

Bu damga işaretleriyle damgalanan resim, Şekil 4.6'de gösterilmiştir. Çok dikkatli incelenecek olunursa resimde damgalanmadan kaynaklanan bozulmalar fark edilebilir.

Damgalanmamış Resim



Damgalanmış Resim

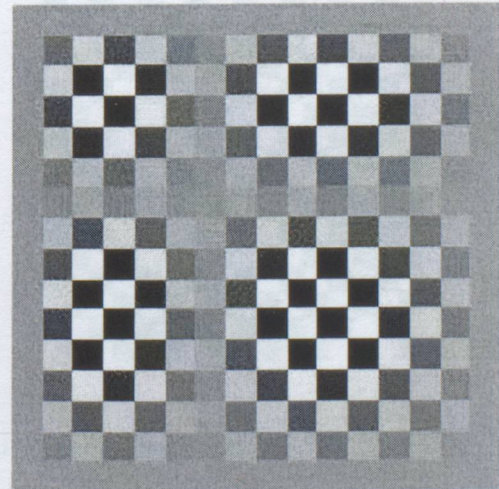
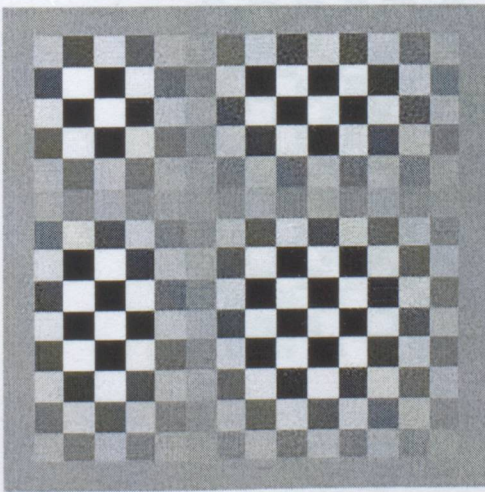


Şekil 4.6 Çörp ve frekansları ters çevrilmiş çörp ile damgalanmış resim.

Resimdeki bu bozulmaları düzeltebilmek için üretilen 2-boyutlu çörp işaretlerinin arasındaki fark maksimize edilmek istenmiştir. Üretilen bir çörp işareti 0 bitini gösteriyorsa, bu çörp işaretinin eksiyle çarpılmış hali 1 bitini gösterirse, iki damga farkının maksimum olacağı düşünülmüştür. Bu şekilde üretilen çörp işaretleri ile daha yüksek performans elde edilebilmiştir. Çizelge 4.2'de kullanılan çörp işaretleri ve hata sayıları, Şekil 4.7'de kullanılan çörp işaretleri, Şekil 4.8'de ise damgalanan resim gösterilmiştir. Damgalama gücü 5, dönüşüm açısı 30 derecedir. Kullanılan çörpün başlangıç frekansı 7 bitiş frekansı 8 dir. Hata sayısı 64 bit için 1'dir.

Çizelge 4.2 Çörp ve eksi işaretli çörp kullanarak damgalama algoritması

Çörp1		Çörp2		
Frekans 1	Frekans 2	Frekans 1	Frekans 2	Hata
5	6	-5	-6	0
6	7	-6	-7	0
8	9	-8	-9	0
9	10	-9	-10	0
5	7	-5	-7	0
6	8	-6	-8	0
7	9	-7	-9	0
8	10	-8	-10	0
5	8	-5	-8	0
6	9	-6	-9	0
5	9	-5	-9	0
6	10	-6	-10	0
6	11	-6	-11	0
5	12	-5	-12	0
5	13	-5	-13	0
5	15	-5	-15	0



Şekil 4.7 Çörp ve eksi işaretli çörp.

Damgalanmamış Resim



Damgalanmış Resim

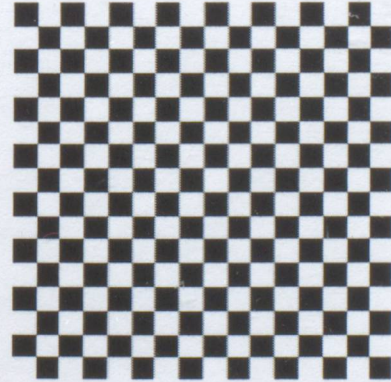
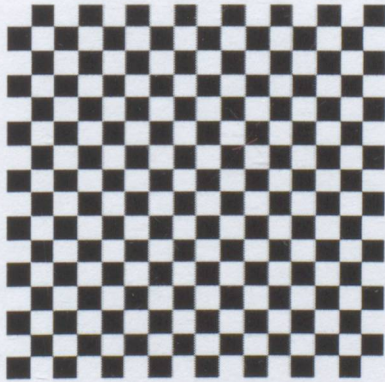


Şekil 4.8 Eksi işaretli çörp ile damgalanmış resim.

KFD'nin farklı açI deęerleri için İlinti Tabanlı, Damga İşareti Olarak Çörp Kullanılarak Damgalama (İT-ÇD) algoritmasının hata miktarları ve damgalanmış ve damgalanmamış resmin algısal uzaklıkları Bölüm 5'te incelenmiştir.

4.3 İlinti Tabanlı, 2-Boyutlu Dalga Katarıyla Damgalama Algoritması

İT-ÇD algoritmasında başlangıç ve bitiş frekansları yakın olan 2-boyutlu çörplerin daha yüksek performans gösterdiği gözlenmiştir. Alıcıda daha yüksek performanslara ulaşabilmek için yüksek farklılığa sahip işaretlerin çörp yerine kullanılabileceği düşünülmüştür (Şekil 4.9).



Şekil 4.9 Yüksek ilintili damga işaretleri.

Önceki algoritmalarındaki gibi İlinti Tabanlı, 2-Boyutlu Dalga Katarıyla Damgalama (İT-DKD) algoritmasında da KFD dönüşümü ile damgalar daha az algılanabilecek şekilde dönüştürölüp, resim bloklarına eklenmesi düşünülmüştür. Damgalama stratejisi Şekil 4.1'de gösterildiği

gibidir. Bu damgalar kullanılarak elde edilen damgalanmış görüntü Şekil 4.10'da gösterilmiştir. Burada KFD açısı 10° , damgalama gücü 4, ve hata sayısı 0'dır. İT-ÇD ve İT-DKD algoritmalarının ayrıntılı karşılaştırması Bölüm 5'te verilmiştir.

Damgalanmamış Resim

Damgalanmış Resim



Şekil 4.10 Maksimum ilintili damgalarla damgalanmış resim.

5. ALGORİTMALARIN PERFORMANSLARI

Bu bölümde, kanalda gürültü olduğu ve olmadığı durumlarda resmin ne kadar bozulduğu incelenmiştir.

5.1 Algılanabilirlik ve Hata Sayıları

Dijital damgalama işleminde resmin ne kadar bozulduğu üçüncü bölümde bahsedilen Watson'un algısal modeliyle ölçülebilir. KFD'nin farklı açı değerleri için İT-ÇD algoritmasının 64 bitteki hata sayısı ve resmin algısal uzaklığı sırasıyla Çizelge 5.1 ve Çizelge 5.2'de gösterilmiştir. Çörp frekansları, çörpün başlangıç ve bitiş frekanslarını göstermektedir.

Çizelge 5.1 İT-ÇD için hata sayıları

Hata	Çörp Frekansları								
Derece	5-6	6-7	7-8	8-9	9-10	5-7	6-8	7-9	8-10
0	1	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0
20	0	0	0	0	1	0	1	0	0
30	4	0	1	0	1	0	0	1	1
40	9	5	25	1	1	3	3	3	24
50	13	3	34	12	8	15	10	4	28
60	24	21	26	20	6	12	20	21	34
70	26	24	20	22	10	16	12	21	22
80	36	36	36	31	17	19	19	17	26
90	28	28	28	26	17	20	19	23	18

Artan KFD değerleri için hatalı bit sayısının arttığı gözlemlenmektedir. Watson modelinde algılanamadan resmin maksimum değiştirilebileceği en yüksek Watson sayısı 4 olarak belirlenmiştir. Çizelge 5.2'de düşük KFD açıları için resmin görsel olarak az değiştiği ve başarılı damgalama işlemlerinin gerçekleştirildiği görülebilir.

Çizelge 5.2 İT-ÇD için algısal uzaklıklar

Algı	Çörp Frekansları								
Derece	5-6	6-7	7-8	8-9	9-10	5-7	6-8	7-9	8-10
0	13,10	10,04	6,90	10,07	13,17	5,90	2,78	2,86	6,08
10	11,83	7,65	4,11	8,51	11,10	6,48	1,85	2,33	6,36
20	12,33	7,18	4,12	6,84	10,91	8,15	2,81	2,86	9,21
30	14,16	8,61	5,59	8,53	14,43	13,55	3,53	4,47	12,23
40	19,62	12,12	8,34	12,56	22,96	16,02	6,75	5,79	12,99
50	24,62	16,38	13,09	18,73	26,48	25,00	8,36	7,38	18,64
60	28,06	24,21	18,08	23,61	20,99	27,98	17,06	8,67	28,36
70	29,49	30,88	17,55	27,83	23,67	29,06	7,89	10,51	41,25
80	28,11	37,44	17,00	35,99	31,33	34,01	21,90	15,04	44,59
90	30,87	31,19	28,55	42,59	36,44	41,84	17,99	22,44	31,86

İT-DKD algoritması için hata ve algı çizelgesi aşağıda gösterilmiştir.

Çizelge 5.3 İT-DKD için hata ve algı bozulma miktarı sayıları

Derece	Hata	Algı
0	0	5,9
10	0	3,0
20	0	4,0
30	1	4,5
40	13	5,5
50	28	10,3
60	30	14,0
70	22	16,2
80	30	10,5
90	31	16,4

Çizelge 5.3’de hatasız ve algısal olarak fark edilemeyecek damgalama gerçekleştirilebildiği gözükmemektedir. Bu performans İT-ÇD algoritmasında yakalanamamıştır.

5.2 Saldırıları

Tezde iki saldırı türü modellenmiştir. Bunlardan biri normal dağılımlı gürültü diğeri ise alçak geçiren filtre saldırısıdır.

5.2.1 Gürültü Saldırısı

Şekil 4.1’de gürültü olarak gösterilen iletim hattında meydana gelebilecek kanal gürültüsü bu bölümde resim boyutlarında bir gürültü olarak modellenmiştir. Oluşturulan gürültü normal dağılımlı ve standart sapması 1 olan rastgele yaratılmış bit dizisidir. Gürültü gücü gürültünün çarpıldığı ve gürültünün genliğini arttıran katsayı olarak tanımlanmıştır. İT-ÇD algoritması için, gürültü gücü 5 ve 10 için resmin hata ve algısal uzaklık sayıları şu şekilde elde edilmiştir.

Çizelge 5.4 Gürültü gücü 5 için İT-ÇD algoritması hata sayısı

Hata	Çörp Frekansları								
Derece	5-6	6-7	7-8	8-9	9-10	5-7	6-8	7-9	8-10
0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	1	1	0
20	1	1	1	0	1	0	1	3	0
30	10	1	2	1	4	2	2	4	0
40	13	7	24	2	1	6	4	8	24
50	18	8	35	13	9	18	10	7	28
60	27	24	27	23	13	16	21	20	34
70	25	24	22	24	13	19	17	25	23
80	36	36	36	32	18	16	22	20	27
90	28	28	28	25	15	19	17	19	20

Çizelge 5.5 Gürültü gücü 5 için İT-ÇD algoritması algısal uzaklık miktarı

Algı	Çörp Frekansları								
Derece	5-6	6-7	7-8	8-9	9-10	5-7	6-8	7-9	8-10
0	135,6	135,0	134,4	134	135,85	134,2	134,0	134,08	134,2
10	135,8	134,7	134,2	134	135,58	134,1	134,0	134,03	134,2
20	135,5	134,5	134,1	134	135,52	134,3	134,0	134,03	134,3
30	136,4	135,2	134,4	135	135,83	134,6	134,0	134,11	134,7
40	137,9	135,6	134,7	135	138,91	134,7	134,2	134,20	134,8
50	139,1	137,2	135,5	137	139,35	136,0	134,2	134,22	134,9
60	141,1	139,2	136,2	139	138,00	136,1	134,6	134,19	136,5
70	141,9	143,4	137,3	140	139,58	137,2	134,2	134,57	138,7
80	141,9	145,1	137,1	146	141,00	137,0	135,3	134,64	140,2
90	142,7	144,3	143,5	148	145,25	139,7	134,9	135,43	138,4

Çizelge 5.6 Gürültü gücü 10 için İT-ÇD algoritması hata sayısı

Hata	Çörp Frekansları								
Derece	5-6	6-7	7-8	8-9	9-10	5-7	6-8	7-9	8-10
0	1	0	1	0	0	0	2	1	0
10	1	0	7	0	1	0	12	9	0
20	1	0	6	0	0	0	4	10	0
30	12	2	5	1	8	4	8	6	1
40	16	7	22	2	4	7	10	11	22
50	17	10	35	15	10	24	14	17	28
60	27	25	27	22	10	17	23	19	34
70	25	24	22	24	14	18	18	23	22
80	36	36	36	32	18	14	24	23	26
90	28	28	28	24	14	18	16	19	20

Çizelge 5.7 Gürültü gücü 10 için İT-ÇD algoritması algısal uzaklık miktarı

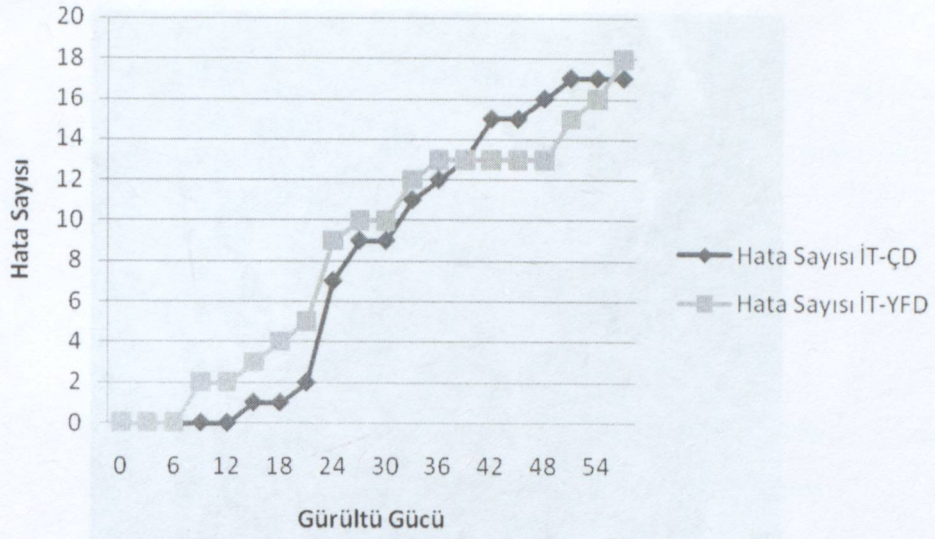
Algı	Çörp Frekansları								
Derece	5-6	6-7	7-8	8-9	9-10	5-7	6-8	7-9	8-10
0	268,7	268,5	268,2	268	268,9	268,4	268,1	268,1	268,4
10	269,1	268,4	268,2	268	268,8	268,2	268,1	268,0	268,4
20	268,7	268,2	268,1	268	268,9	268,5	268,1	268,0	268,5
30	269,1	268,7	268,3	268	268,7	269,0	268,1	268,2	269,1
40	270,2	268,7	268,4	268	270,7	269,0	268,3	268,3	269,2
50	270,4	269,7	268,9	269	270,4	270,8	268,3	268,3	269,2
60	271,6	270,4	268,8	270	269,9	270,8	268,9	268,3	271,5
70	272,2	272,5	269,7	271	271,2	272,7	268,4	268,9	274,5
80	271,6	273,9	269,9	274	270,7	272,1	269,9	268,9	276,5
90	272,9	273,3	272,5	275	274,5	276,2	269,4	270,1	274,5

Gürültüye karşı gürbüzlük performans analizleri İT-DKD için Çizelge 5.8’de gösterilmiştir.

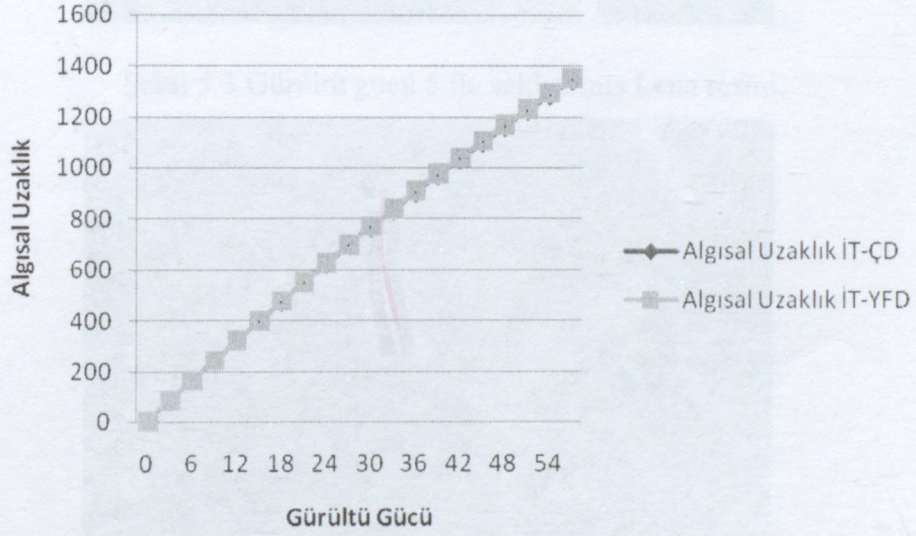
Çizelge 5.8 Gürültü gücü 5 ve10 için İT-DKD algoritması hatalı bit sayısı ve algısal uzaklık miktarı

Derece	Gürültü Gücü=5		Gürültü Gücü=10	
	Hata	Algısal Uzaklık	Hata	Algısal Uzaklık
0	4	134,78	6	268,04
10	0	134,48	2	267,34
20	1	133,81	3	265,95
30	1	133,33	3	265,04
40	12	133,32	13	265,04
50	31	133,21	31	264,78
60	25	132,02	25	262,34
70	24	130,95	24	260,31
80	36	135,40	35	269,17
90	27	139,28	27	276,82

Başlangıç ve bitiş frekansları 6 ve 7 için İT-ÇD algoritmasıyla İT-DKD algoritmasının gürültü gücü-hata ve gürültü gücü-algısal bozulma grafikleri Şekil 5.1 ve Şekil 5.2’de gösterilmiştir. Grafikler elde edilirken KFD derecesi 10 seçilmiştir. Şekil 5.3 ve Şekil 5.4’de ise gürültü gücünün Lena resmi üzerindeki etkisi gösterilmiştir.



Şekil 5.1 İT-ÇD ve İT-DKD algoritmaları gürültü gücü- hata sayısı karşılaştırması.



Şekil 5.2 İT-ÇD ve İT-DKD gürültü gücü- algısal uzaklık karşılaştırması.

5.2.2 Alçak Güçlü Gürültü Saldırısı

Bu bölümde Şekil 5.3'te görülen Lena resmi gürültü ile saldırılmıştır.

Kullanılan alçak güçlü gürültü sinyali aşağıdaki gibidir.

$$AGP = \frac{1}{1 + AGP} \cdot \sin(2\pi f t)$$

AGP gücü 5 ve 10 ile saldırılmış resmi aşağıdaki gibidir.



Şekil 5.3 Gürültü gücü 5 ile saldırılmış Lena resmi.



Şekil 5.4 Gürültü gücü 10 ile saldırılmış Lena resmi.

5.2.2 Alçak Geçiren Filtre Saldırısı

Bu bölümde Şekil 4.1'deki gürültü modeli alçak geçiren filtre olarak modellenmiştir. Alçak geçiren filtre saldırısı, resmin alçak geçiren bir filtreyle konvolüsyonundan elde edilmiştir. Kullanılan alçak geçiren filtre matrisi aşağıda gösterilmiştir.

$$AGF = \frac{1}{8+AGF \text{ Gücü}} \begin{bmatrix} 1 & 1 & 1 \\ 1 & AGF \text{ Gücü} & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (5.1)$$

AGF gücü 5 ve 10 seçilerek farklı frekans değerleri için İT-ÇD algoritmasının hata sayısı ve algısal uzaklık miktarı sırasıyla Çizelge 5.9 ve Çizelge 5.10'da gösterilmiştir.

Çizelge 5.9 AGF 5 için İT-ÇD algoritması hata sayısı

Hata	Çörp Frekansları								
Derece	5-6	6-7	7-8	8-9	9-10	5-7	6-8	7-9	8-10
0	0	0	0	0	0	0	0	0	0
10	0	0	2	0	0	0	0	0	0
20	19	1	0	1	1	0	1	6	2
30	26	16	17	2	14	6	1	21	14
40	32	25	27	14	17	13	14	16	36
50	25	21	36	22	22	26	21	25	28
60	34	34	28	33	24	19	29	27	36
70	27	28	26	25	18	22	23	32	27
80	36	36	36	33	26	25	28	27	33
90	28	28	28	26	21	22	23	25	27

Çizelge 5.10 AGF gücü 5 için İT-ÇD algoritması algısal uzaklık miktarı

Algı	Çörp Frekansları								
Derece	5-6	6-7	7-8	8-9	9-10	5-7	6-8	7-9	8-10
0	212,4	212,2	212,0	212	212,4	212,0	211,7	211,8	212,1
10	212,6	212,4	212,0	212	212,5	212,1	211,9	211,8	212,1
20	212,6	212,4	211,9	212	212,5	212,5	211,9	211,9	212,5
30	213,2	212,5	212,0	212	213,3	212,5	212,0	211,9	212,6
40	213,3	213,0	212,0	212	213,5	213,1	212,0	212,0	213,0
50	215,2	213,2	212,4	213	215,5	216,1	212,1	212,0	213,9
60	216,1	215,8	213,2	214	213,9	216,6	213,4	212,3	215,6
70	216,7	217,4	213,3	216	215,6	217,5	212,3	212,1	222,8
80	219,8	218,0	212,8	219	218,5	218,8	213,3	212,7	224,0
90	218,0	217,5	215,9	220	219,8	223,4	213,3	214,3	220,2

Çizelge 5.11 AGF gücü 10 için İT-ÇD algoritması hata sayısı

Hata	Çörp Frekansları								
Derece	5-6	6-7	7-8	8-9	9-10	5-7	6-8	7-9	8-10
0	0	0	0	0	0	0	0	0	0
10	0	0	3	0	0	0	1	0	0
20	3	1	0	0	1	0	1	2	0
30	22	4	6	1	8	3	1	10	2
40	27	19	27	5	7	9	9	13	34
50	21	15	36	20	15	22	15	17	28
60	30	31	28	32	22	19	26	27	35
70	27	27	25	24	17	21	20	30	24
80	36	36	36	32	21	21	26	22	31
90	28	28	28	26	17	20	22	25	27

Çizelge 5.12 AGF gücü 10 için İT-ÇD algoritması algısal uzaklık miktarı

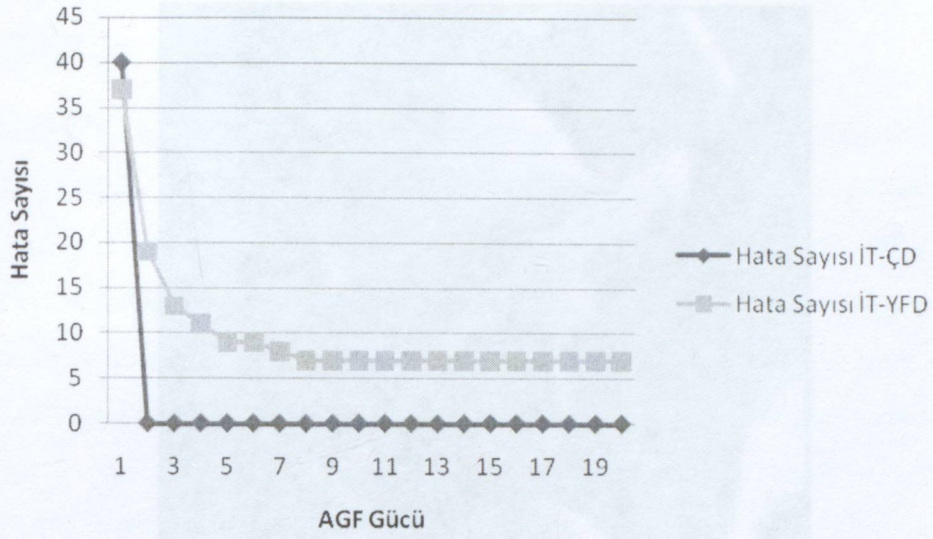
Algı	Çörp Frekansları								
Derece	5-6	6-7	7-8	8-9	9-10	5-7	6-8	7-9	8-10
0	154,7	154,0	153,5	154	154,6	153,5	153,0	153,1	153,5
10	154,7	154,1	153,3	154	154,6	153,6	153,2	153,1	153,7
20	154,8	154,0	153,3	154	154,6	154,2	153,2	153,2	154,3
30	155,7	154,3	153,5	154	155,8	154,7	153,4	153,3	154,9
40	156,4	155,3	153,7	155	156,8	155,9	153,5	153,6	155,5
50	159,0	155,9	154,5	155	159,8	160,2	153,8	153,7	156,9
60	160,6	159,9	156,0	158	157,4	160,7	156,0	154,1	160,2
70	161,4	162,3	156,1	160	159,6	162,2	154,1	153,9	169,8
80	164,8	163,7	155,1	164	163,8	164,5	156,1	155,0	170,7
90	163,1	163,1	160,3	168	165,9	170,8	156,1	157,7	165,7

AGF'ye karşı gürbüzlük performans analizleri İT-DKD için Çizelge 5.13'de gösterilmiştir.

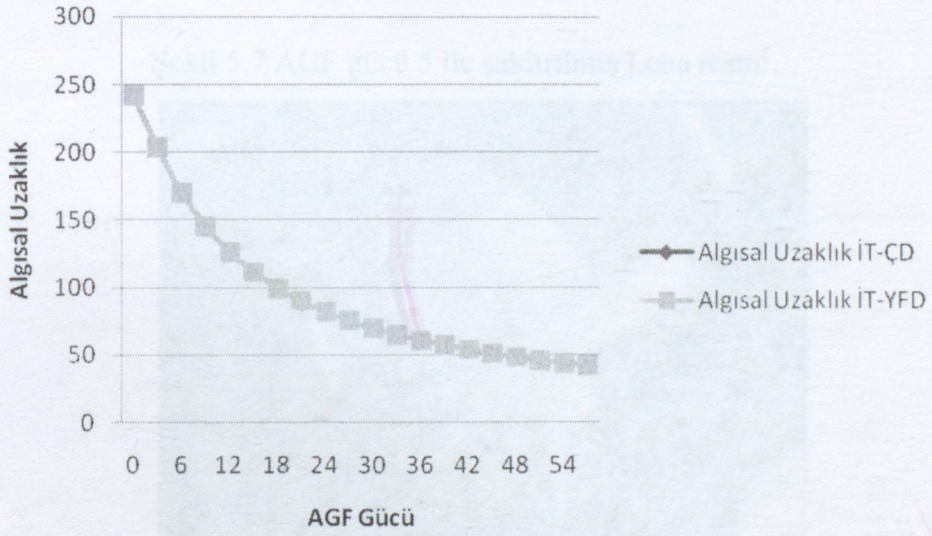
Çizelge 5.13 AGF gücü 5 ve 10 için İT-DKD algoritması hatalı bit sayısı ve algısal uzaklık miktarı

Derece	AGF=5		AGF=10	
	Hata	Algısal Uzaklık	Hata	Algısal Uzaklık
0	1	180,4533	0	138,3103
10	2	180,1304	1	138,0116
20	3	179,7863	1	137,6425
30	16	179,5389	10	137,3835
40	27	179,5384	25	137,4002
50	36	179,6481	35	137,4374
60	28	179,1619	27	136,8112
70	29	178,5887	30	136,3401
80	36	180,7781	36	138,7102
90	1	180,4533	0	138,3103

Başlangıç ve bitiş frekansları 6 ve 7 için İT-ÇD algoritmasıyla İT-DKD algoritmasının AGF gücü-hata ve AGF gücü-algısal bozulma grafikleri Şekil 5.5 ve Şekil 5.6'da gösterilmiştir. Grafikler elde edilirken KFD derecesi 10 seçilmiştir. Şekil 5.7 ve Şekil 5.8'de ise AGF gücünün Lena resmi üzerindeki etkisi gösterilmiştir.



Şekil 5.5 IT-ÇD ve IT-DKD algoritmaları AGF gücü- hata sayısı karşılaştırması.



Şekil 5.6 IT-ÇD ve IT-DKD algoritmaları AGF gücü- algısal uzaklık miktarı karşılaştırması.



Şekil 5.7 AGF gücü 5 ile saldırılmış Lena resmi.



Şekil 5.8 AGF gücü 10 ile saldırılmış Lena resmi.

6. SONUÇLAR

Tezde KFD uzayı kullanılarak ilinti tabanlı damgalama algoritması geliştirilmiştir. Özel olarak seçilmiş damga işaretlerinin performansları KFD'nin sağladığı ekstra serbestlik bölgesi kullanılarak arttırılmıştır. Algoritmalarda seçilen damga işaretleri mümkün olduğunca kolay tespit edilebilecek özellikte olması istenmiştir. Damga işaretlerinin, KFD uzayında değişik açı değerleri için algısal özelliklerinin azalması ve resimde daha zor fark edilmesi istenmiştir. Sonuçlar, KFD'nin küçük açı değerleri için algoritmanın, damgalama performansında artışa sebep olduğunu göstermiştir. Damga işaretlerinin yüksek frekans taşınması, yüksek KFD değerleri için performansını düşürmüştür.

İT-ÇD algoritmasında görsel bozulma daha yüksek olurken özellikle AGF'ye karşı gürbüzlük yüksek, İT-DKD algoritmasında ise görsel bozulma daha düşük ancak özellikle AGF'ye karşı gürbüzlük de daha düşük olmuştur (Şekil 5.5). Bunun sebebi İT-DKD algoritmasında kullanılan damga işaretlerinin daha yüksek frekansa sahip olması ve AGF saldırısından daha kolay etkilenmesidir.

İT-DKD algoritmasında kullanılan damga işaretlerinin ilintiyle algılanabilmesi daha kolaydır. Bu özellik, resim damgalandıktan sonra alıcı kısmında daha rahat algılanabilmesini sağlamıştır. Çizelge 5.3, KFD açısı 10° için İT-DKD algoritmasının görsel olarak algılanamayacak damgalama gerçekleştirdiğini göstermektedir. İT-ÇD algoritmasında da başlangıç ve bitiş frekansları birbirine yakın çörp işaretleri için, damga performanslarında artış sağlandığı gösterilmiştir (Çizelge 5.2). Başlangıç ve bitiş frekansları birbirine uzak olan damgalar için algoritmanın performansı kötüdür.

İT-ÇD algoritmasında uygun çörp frekansları 16×16 'lık bloklar için olabilecek uygun frekansların hepsi taranarak tespit edilmiştir. Başlangıç frekansı 8, bitiş frekansı 9 olan çörp işareti öteki olası damgalardan daha yüksek gürbüzlük sağlamıştır (Çizelge 5.4, Çizelge 5.6, Çizelge 5.9, Çizelge 5.11). Başlangıç frekansı 7, bitiş frekansı 8 olan çörp işareti de görsel olarak öteki çörplerden yüksek performans göstermiştir (Çizelge 5.2).

Algılama benzetimleri Watson'un AKD tabanlı modeliyle gerçekleştirilmiştir. Gönderilen ve alınan resimler arasındaki bit hata miktarına bakmak yanıltıcıdır. Az bit hata sayısına sahip resimlerde görsel olarak yüksek miktarlarda bozulmalar meydana gelebilir. Bunun için performans analizlerinde gözün algısal özellikleri kullanılmıştır.

Damgalama işlemi 2-boyutlu resimler üzerinde denenmiş olsa da, 1-boyutlu ses işaretleri ve 3-boyutlu video işaretleri için de uygulanabilir. Ancak 1-boyutlu damgalamada performans analizleri Watson'un duyuşal modeliyle gerçekleştirilmelidir.

İleride yapılacak çalışmalarda farklı damga işaretlerinin performansları incelenebilir. Bunun yanında algoritma trellis tabanlı damgalama ve kirli kağıt damgalama algoritmalarıyla birleştirilebilir ve bit hata performansı artırılabilir.

- Boashash, B. (1997), "Embedding in the Fractional Fourier Transform", *Applied and Computational Harmonic Analysis*, 16(3):132-162, Bingham.
- Buhacec, A., (2007), "Digital Watermarking of Images in the Fractional Fourier Domain", *Katholieke Universiteit Leuven Technical Report 197*.
- Claasen, T.A.C.M. ve Mecklenbrauker, W.F.G. (1980), "The Wigner Distribution - A Tool for Time-Frequency Signal Analysis, part I: Continuous-Time Signals", *Philips Journal of Research*, 35(4):217-250.
- Cohen, L., (1966), "Generalized Phase-Space Distribution Functions", *Journal of Mathematical Physics*, vol:7:1-736.
- Cohen, L. ve Morgeresu, H., (1987), *Probabilities in Quantum Mechanics in Quantum Theory and Reality*, Springer-Verlag, New York.
- Cohen, L., (1994), "Time-Frequency Analysis: Theory and Applications", *Prentice Hall Signal Processing Series*.
- Costa, M., (1983), "Writing on dirty paper", *IEEE Transaction Information Theory*, 29:439-441.
- Cox, I. J., Miller, M. L. ve McKelips, A. L., (1999), "Watermarking As Communications with Side Information", *Proceedings IEEE ISIT*, 2: 1111-1114, July 1999.
- Cox, I. J., Miller, M. L. ve Bloom, J. A., (2002), *Digital Watermarking*, Morgan Kaufmann Publishers.
- Djuricic, I., Stankovic, S. ve Pitas, I., (2000), "Digital Watermarking in the Fractional Fourier Transformation Domain", *Journal of Network and Computer Applications*, 24:167-173.
- Erdem, M. F., Kalay, M. A. ve Oksuz, H. M., (1997), "Repeated Filtering in Consecutive Fractional Fourier Domains and its Applications to Signal Restoration", *IEEE Transactions Signal Processing*, 47(5):1454-1462.
- Kernam, N. ve Tomasek, H., (1985), "Authentication System Using Successive Images in Teleimages", *Memories of The School of Science and Engineering, Waseda University*, 32:43-60.
- Miller, M. L., Duett, G. ve Cox, I. J., (1997), "Dirty paper hello codes for watermarking", *IEEE International Conference on Image Processing*.
- Miller, M. L., Duett, G. ve Cox, I. J., (2004), "Applying Informal Coding and Informal Embedding in Design a Robust High Capacity Watermark", *IEEE Transactions on Image Processing*, 13:792-807.
- Miller, M. L., (2001), "Watermarking With Dirty Paper Codes", *Proceedings IEEE International Conference on Image Processing*, 2:334-338, Thessaloniki, Greece.

KAYNAKLAR

- Altun, O., Sharma, G. ve Bocko, M. "Informed Watermark Embedding in the Fractional Fourier Domain", University of Rochester, NY.
- Barni, M., Bartolini, F., (2004), "Data hiding for fighting piracy", in IEEE Signal Proceedings, 21:28-39.
- Bultheel, A. ve Martinez, H. E., (2004), "Computation of the Fractional Fourier Transform", Applied and Computational Harmonic Analysis, 16(3):182-202, Belgium.
- Bultheel, A., (2007), "Digital Watermarking of Images in the Fractional Fourier Domain", Katholieke Universiteit Leuven Technical Report 497.
- Claasen, T.A.C.M. ve Mecklenbrauker, W.F.G, (1980), "The Wigner Distribution - A Tool for Time-Frequency Signal Analysis, part 1: Continuous-Time Signals", Phillips Journal of Research, 35(3):217-250.
- Cohen, L., (1966), "Generalized Phase-Space Distribution Functions", Journal of Mathematical Physics., vol:781-786.
- Cohen, L. ve Margeneau, H., (1967), Probabilities in Quantum Mechanics, in Quantum Theory and Reality, Springer-Verlag, New York.
- Cohen, L., (1994), "Time Frequency Analysis: Theory and Applications", Prentice Hall Signal Processing Series.
- Costa, M., (1983), "Writing on dirty paper", IEEE Transaction Information Theory, 29:439-441.
- Cox, I. J., Miller, M. L. ve McKellips, A. L., (1999), "Watermarking As Communications with Side Information", Proceedings IEEE, 87:1127-1141, July 1999.
- Cox, I. J., Miller, M., L., ve Bloom, J., A., (2002), Digital Watermarking, Morgan Kaufmann Publishers.
- Djurovic, I., Stankovic, S. ve Pitas, I., (2000), "Digital Watermarking in the Fractional Fourier Transformation Domain", Journal of Network and Computer Applications, 24:167-173.
- Erden, M. F., Kutay, M. A. ve Özaktaş, H. M., (1997), "Repeated Filtering in Consecutive Fractional Fourier Domains and its Applications to Signal Restoration" IEEE Transactions Signal Processing, 47(5):1458-1462.
- Komatsu, N. ve Tominaga, H. (1988), "Authentication System Using Concealed Images in Telematics", Memoirs of The School of Science and Engineering, Waseda University, 52:45-60.
- Miller, M. L., Doerr, G. ve Cox, I. J., (2002), "Dirty-paper trellis codes for watermarking", IEEE International Conference on Image Processing.
- Miller, M. L., Doerr, G. ve Cox, I. J. (2004), "Applying Informed Coding and Informed Embedding to Design a Robust, High Capacity Watermark", IEEE Transactions on Image Processing, 13:792-807.
- Miller, M. L., (2001), "Watermarking With Dirty Paper Codes", Proceedings IEEE International Conference in Image Processing, 2:538-541, Thessaloniki, Greece.

- Miller, M. L., Cox, I. J. ve Bloom, J. A. (2000), "Informed Embedding: Exploiting Image and Detector Information During Watermark Insertion", IEEE International Conference in Image Processing.
- Moulin, P. ve O'Sullivan, J.A., (2000), "Information-theoretic Analysis of Information Hiding", in International Conference on Acoustics, Speech, and Signal Processing.
- Özaktaş, H. M. ve Barshan, B., (1994). "Convolution, Filtering, and Multiplexing in Fractional Fourier Domains and Their Relation to Chirp and Wavelet Transforms", Journal of the Optical Society of America A, 11:547-559.
- Özaktaş, H. M., Arıkan, O., Kutay, M. A. ve Bozdağı G., (1996), "Digital Computation of the Fractional Fourier Transform", IEEE Transactions Signal Processing, 44(9):2141-2150.
- Özaktaş, H. M. ve Zalevski, Z., (2001), "The Fractional Fourier Transform with Applications in Optics and Signal Processing", Wiley & Sons, New York.
- Özaktaş, H. M., Zalevsky, Z. ve Kutay, M.A., (2001), The Fractional Fourier Transform, John Wiley & Sons, NY.
- Pei, S. C. ve Yeh, M. H., (1998), "Two Dimensional Fractional Fourier Transform" IEEE Signal Processing, 67:105-109.
- Ping, D., Brankov, J.G., Galatsanos, N.P., Yongyi, Y. ve Davoine, F, (2005), "Digital Watermarking Robust to Geometric Distortions" IEEE Transactions on Image Processing, 14(12):2140-2150.
- Solachidis, V ve Pitas, I, (2007), "Watermarking Digital 3-D Volumes in the Discrete Fourier Transform Domain" IEEE Transactions on Multimedia, 9(7):1373-1383.
- Stankovic S., Djurovic, I., Pitas I., (2001), "Watermarking in the space/spatial-frequency domain using two-dimensional Radon-Wigner distribution", IEEE Transactions on Image Processing, 10:650 – 658.
- Strang, G. ve Nguyen, T., (1996), Wavelets and Filter Banks, Wellesley-Cambridge Press.
- Vedat Tavşanoğlu, An Introduction to Digital Signal Processing, Ders Notları.
- Viterbi, A. J., (1967) "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm", IEEE Transactions on Information Theory 13(2):260-269.
- Wang, D-H., Li, D-M. ve Jun, Y., (2007), "An Improved Chirp Typed Blind Watermarking Algorithm Based on Wavelet and Fractional Fourier Transform", Fourth International Conference on Image and Graphics.
- Wang, P, Qin F., Wang X., Feng G. ve Cai C., (2006), "A Novel Blind Watermarking Scheme Based on Watermark Hopping Technique" The 8th International Conference on Signal Processing, 2:16-20.
- Wang, R. D., Xu, D. W. ve Li, Q, (2005) "Multiple Audio Watermarks Based On Lifting Wavelet Transform", International Conference on Machine Learning and Cybernetics, 4: 1959-1964, Guangzhou.
- Wang, C., Doerr, G. ve Cox, I.J., (2006), "Toward a Better Understanding of Dirty Paper Trellis Codes", in Proceedings of the IEEE International Conference on Acoustics, iSpeech, and Signal Processing, 2:233-236.
- Watson, A. B., (1993), "DCT Quantization Matrices Optimized for Individual Images", Human Vision, Visual Processing, and Digital Display IV, San Jose, California, USA.

Yu, F.Q., Zhang, Z.K. ve Xu, M.H., (2006), "A Digital Watermarking Algorithm for Image Based on Fractional Fourier Transform", 1ST IEEE Conference on Industrial Electronics and Applications.

İNTERNET KAYNAKLARI

- (1) <http://fourier.eng.hmc.edu/e161/lectures/dct/node1.html>
- (2) <http://www-ccrma.stanford.edu/~ujung/mylec/WTpart1.html>
- (3) www.compapp.dcu.ie/~john/trellis_algorithms.ppt
- (4) <http://www.cs.kuleuven.ac.be/~nalag/papers/ade/frftcomp/ccomp.pdf>
- (5) <http://www.dei.unipd.it/~nil/vis-e.html>

EKLER

Ek1: Çalıştırılan Algoritmalar

B. Blind_D_C1

- Ek1 Çalıştırılan Programlar
 Ek2 Programlarda Kullanılan Fonksiyonlar

```

mesaj=0; %mesaj 1 ya da 0 ile çalıştırılabilir
alpha=1; %Embedding Strength
alpha=0.1; %kayma hatasız gürültüsü
brus=0.5;

%-----EMBEDDER-----
% Cofin Oluşturulması
X=imread('air_little.jpg');
Co=double(X);
figure, imshow(Co,[],notsize);
% W'in Oluşturulması
W=imread('w.jpg');
figure, imshow(W,[],notsize);
% W'in Oluşturulması
if (mesaj==1);
    Wm=W;
else if (mesaj == 0);
    Wm= W;
end;
end;
% Wa ve Co'nun Oluşturulması
Wa=alpha*Wm;
Cw=Wa*Co;
figure, imshow(Cw,[],notsize);
%-----LIFTING-----
n=length(size(Cw))*2;
C=Cw+n;
%-----DETECTOR-----
% Linear Correlation Oluşturulması
rho=corrcoef(size(C),size(C))*(C.*Wm);
% thresholding
if (rho>0.9);
    mesaj=1;
else if (rho<0.9);
    mesaj=0;
end;
end;

%-----B. Blind_D_C1-----
% Cofin Oluşturulması
mesaj=0; %mesaj 1 ya da 0 ile çalıştırılabilir
alpha=1; %Embedding Strength
alpha=0.1; %kayma hatasız gürültüsü
brus=0.5;

```


Ek1: Çalıştırılan Algoritmalar

E_Blind__D_CL

```
clear all;close all;clc;
mesaj=0; %mesaj 1 ya da sıfır olabilir!!!
alfa=1; %Embedding Strenght
alfaN=0.1; %iletim hattinin gurultusu
thres=0.5;
%-----EMBEDDER-----
% Co'in Olusturulmasi-----
X=imread('uur little.JPG');
Co=double(X);
figure, imshow(Co,[],'notruesize');
% Wr'in Olusturulmasi-----
Wr=randn(size(Co));
figure, imshow(Wr,[],'notruesize');
% Wm'in Olusturulmasi-----
if (mesaj==1);
    Wm=Wr;
else if (mesaj == 0);
    Wm=-Wr;
end;
end;
% Wa'in ve Cw'nin Olusturulmasi-----
Wa=alfa*Wm;
Cw=Wa+Co;
figure, imshow(Cw,[],'notruesize');
%-----ILETIM-----
n=randn(size(Co))*alfaN;
C=Cw+n;
%-----DETECTOR-----
% Linear Correlation Olusturulmasi-----
zlc= sum(sum(1/(size(C,1)*size(C,2))*(C .* Wr)));
% thresholding-----
if (zlc>thres);
    sonuc=1;
else if (zlc<-thres);
    sonuc=0;
else
    sonuc=10;
end;
end;
```

E_TRELLIS_8__D_TRELLIS_8

```
clear all;close all;clc;
mesaj=[0 1 0 0 1 1 1 0]; %mesaj
alfa=1; %Embedding Strenght
alfaN=0; %iletim hattinin gurultusu
% thres=0.1;
```



```

%-----EMBEDDER-----
% Co'in Olusturulmasi-----
X=imread('uur toolittle.JPG');
Co=double(X);
% Cnorm=Co/(max(max(Co)));
Co=-10*(mean(mean(Co))-Co)/max(max(Co));
figure, imshow(Co,[],'notruesize');
% Wr Olusturulmasi-----
state=zeros(1,9);
for i=1:8;
    for j=1:8;
        Wr(:,:,i,j)=randn(size(Co));
    end;
end;
% TRELLIS CODE-----
% STATE TABLOSU-----
% Wm'in Olusturulmasi-----
state(1)=1;
for i=1:8;
    if (mesaj(i)==0 && state(i)==1);
        state(i+1)=1;
        Wm(:,:,i)=Wr(:,:,i,1);
    elseif(mesaj(i)==0 && state(i)==2);
        state(i+1)=3;
        Wm(:,:,i)=Wr(:,:,i,2);
    elseif(mesaj(i)==0 && state(i)==3);
        state(i+1)=5;
        Wm(:,:,i)=Wr(:,:,i,3);
    elseif(mesaj(i)==0 && state(i)==4);
        state(i+1)=7;
        Wm(:,:,i)=Wr(:,:,i,4);
    elseif(mesaj(i)==0 && state(i)==5);
        state(i+1)=1;
        Wm(:,:,i)=Wr(:,:,i,5);
    elseif(mesaj(i)==0 && state(i)==6);
        state(i+1)=3;
        Wm(:,:,i)=Wr(:,:,i,6);
    elseif(mesaj(i)==0 && state(i)==7);
        state(i+1)=5;
        Wm(:,:,i)=Wr(:,:,i,7);
    elseif(mesaj(i)==0 && state(i)==8);
        state(i+1)=7;
        Wm(:,:,i)=Wr(:,:,i,8);
    elseif(mesaj(i)==1 && state(i)==1);
        state(i+1)=2;
        Wm(:,:,i)=-Wr(:,:,i,1);
    elseif(mesaj(i)==1 && state(i)==2);
        state(i+1)=4;
        Wm(:,:,i)=-Wr(:,:,i,2);
    elseif(mesaj(i)==1 && state(i)==3);
        state(i+1)=6;

```



```

    Wm(:,i)=-Wr(:,i,3);
elseif(mesaj(i)==1 && state(i)==4);
    state(i+1)=8;
    Wm(:,i)=-Wr(:,i,4);
elseif(mesaj(i)==1 && state(i)==5);
    state(i+1)=2;
    Wm(:,i)=-Wr(:,i,5);
elseif(mesaj(i)==1 && state(i)==6);
    state(i+1)=4;
    Wm(:,i)=-Wr(:,i,6);
elseif(mesaj(i)==1 && state(i)==7);
    state(i+1)=6;
    Wm(:,i)=-Wr(:,i,7);
else(mesaj(i)==1 && state(i)==8);
    state(i+1)=8;
    Wm(:,i)=-Wr(:,i,8);
end;
end;
% Wa'in ve Cw'nin Olusturulmasi-----
Wmt=zeros(8);
for i=1:8;
    Wmt=Wm(:,i)+Wmt;
end;
Wa=alfa*Wmt;
Cw=Wa+Co;
figure, imshow(Cw,[],'notruesize');
%-----ILETIM-----
n=randn(size(Co))*alfaN;
C=Cw+n;
%-----DETECTOR-----
% Viterbi Decoder-----
for i=1:8;
    for j=1:8;
        cor(:,i,j)=sum(sum(Wr(:,i,j).*C));
        cor(:,i,j+8)=sum(sum(-Wr(:,i,j).*C));
    end;
end;
cor1=0;
cor2=0;
cor3=0;
cor4=0;
cor5=0;
cor6=0;
cor7=0;
cor8=0;
stateprev=zeros(1,9);
stateprev(1)=1;
for i=1:8;
    if (cor(:,i,1)>cor(:,i,5)&&stateprev(i)==1);
        stateprev(i+1)=1;
        cor1=cor(:,i,1)+cor1;

```



```

elseif (cor(:,i,1)<cor(:,i,5)&&stateprev(i)==5);
    stateprev(i+1)=1;
    cor1=cor(:,i,5)+cor1;
elseif (cor(:,i,9)>cor(:,i,13)&&stateprev(i)==1);
    stateprev(i+1)=2;
    cor2=cor(:,i,9)+cor2;
elseif (cor(:,i,9)<cor(:,i,13)&&stateprev(i)==5);
    stateprev(i+1)=2;
    cor2=cor(:,i,13)+cor2;
elseif (cor(:,i,2)>cor(:,i,6)&&stateprev(i)==2);
    stateprev(i+1)=3;
    cor3=cor(:,i,2)+cor3;
elseif (cor(:,i,2)<cor(:,i,6)&&stateprev(i)==6);
    stateprev(i+1)=3;
    cor3=cor(:,i,6)+cor3;
elseif (cor(:,i,10)>cor(:,i,14)&&stateprev(i)==2);
    stateprev(i+1)=4;
    cor4=cor(:,i,10)+cor4;
elseif (cor(:,i,10)<cor(:,i,14)&&stateprev(i)==6);
    stateprev(i+1)=4;
    cor4=cor(:,i,14)+cor4;
elseif (cor(:,i,3)>cor(:,i,7)&&stateprev(i)==3);
    stateprev(i+1)=5;
    cor5=cor(:,i,3)+cor5;
elseif (cor(:,i,3)<cor(:,i,7)&&stateprev(i)==7);
    stateprev(i+1)=5;
    cor5=cor(:,i,7)+cor5;
elseif (cor(:,i,11)>cor(:,i,15)&&stateprev(i)==3);
    stateprev(i+1)=6;
    cor6=cor(:,i,11)+cor6;
elseif (cor(:,i,11)<cor(:,i,15)&&stateprev(i)==7);
    stateprev(i+1)=6;
    cor6=cor(:,i,15)+cor6;
elseif (cor(:,i,4)>cor(:,i,8)&&stateprev(i)==4);
    stateprev(i+1)=7;
    cor7=cor(:,i,4)+cor7;
elseif (cor(:,i,4)<cor(:,i,8)&&stateprev(i)==8);
    stateprev(i+1)=7;
    cor7=cor(:,i,8)+cor7;
elseif (cor(:,i,12)>cor(:,i,16)&&stateprev(i)==4);
    stateprev(i+1)=8;
    cor8=cor(:,i,12)+cor8;
elseif (cor(:,i,12)<cor(:,i,16)&&stateprev(i)==8);
    stateprev(i+1)=8;
    cor8=cor(:,i,16)+cor8;
end;
end;
E_BLK_FIXED_CC__D_BLK_CC

```

```
clear all;close all;clc;
```

```
mesaj=0;          %mesaj 1 ya da sıfır olabilir!!!
```



```

alfaN=0;          %iletim hattinin gurultusu
tcc=0.55;
beta=0.25;
thres=0.2;
pic_en=128;
pic_boy=128;
%-----EMBEDDER-----
% Co'in Olusturulmasi-----

X=imread('uur little128.JPG');
Co=double(X);
figure, imshow(Co,[],'notruesize');
% Segmentasyon, Marking Space-----
% -----
en=pic_en/8;
boy=pic_boy/8;
for j=1:8;
    for i=1:8;
        Vo(:,i,j)=Co(boy*(i-1)+1:boy*i,en*(j-1)+1:en*j);
    end;
end;
% Wr'in Olusturulmasi-----
Wr=randn(size(Vo(:,i,1)));
figure, imshow(Wr,[],'InitialMagnification',500);
% Wm'in Olusturulmasi-----
if (mesaj==1);
    Wm=Wr;
else if (mesaj == 0);
    Wm=-Wr;
end;
end;
% Wa'in ve Cw'nin Olusturulmasi-----
X1=Wm-mean(mean(Wm));
Y1=Vo(:,i,4)-mean(mean(Vo(:,i,4)));
XY=0;
for (i=1:16);
    for(j=1:16);
        XY=XY+X1(i,j).*Y1(i,j);
    end;
end;
X1=X1./sum(sum(abs(X1)));
for (i=1:16);
    for (j=1:16);
        Y1(i,j)=Y1(i,j)-(XY.*X1(i,j));
    end;
end;
Y1=Y1./sum(sum(abs(Y1)));
xvo=0;
yvo=0;
for (i=1:16);
    for(j=1:16);

```



```

    xvo=xvo+Vo(i,j,4,4) .* X1(i,j);
    yvo=yvo+Vo(i,j,4,4) .* Y1(i,j);
end;
end;
xt=tcc+beta;
yt=sqrt(1-xt.*xt);
xvw=xt.*(xt.*xvo+yt.*yvo);
yvwy=yt.*(xt.*xvo+yt.*yvo);
Vw=xvw.*X1+yvwy.*Y1 + mean(mean(Vo(:, :, 4, 4)));
Cw=[Vo(:, :, 1, 1) Vo(:, :, 1, 2) Vo(:, :, 1, 3) Vo(:, :, 1, 4) Vo(:, :, 1, 5) Vo(:, :, 1, 6) Vo(:, :, 1, 7)
Vo(:, :, 1, 8);
    Vo(:, :, 2, 1) Vo(:, :, 2, 2) Vo(:, :, 2, 3) Vo(:, :, 2, 4) Vo(:, :, 2, 5) Vo(:, :, 2, 6) Vo(:, :, 2, 7) Vo(:, :, 2, 8);
    Vo(:, :, 3, 1) Vo(:, :, 3, 2) Vo(:, :, 3, 3) Vo(:, :, 3, 4) Vo(:, :, 3, 5) Vo(:, :, 3, 6) Vo(:, :, 3, 7) Vo(:, :, 3, 8);
    Vo(:, :, 4, 1) Vo(:, :, 4, 2) Vo(:, :, 4, 3) Vw      Vo(:, :, 4, 5) Vo(:, :, 4, 6) Vo(:, :, 4, 7) Vo(:, :, 4, 8);
    Vo(:, :, 5, 1) Vo(:, :, 5, 2) Vo(:, :, 5, 3) Vo(:, :, 5, 4) Vo(:, :, 5, 5) Vo(:, :, 5, 6) Vo(:, :, 5, 7) Vo(:, :, 5, 8);
    Vo(:, :, 6, 1) Vo(:, :, 6, 2) Vo(:, :, 6, 3) Vo(:, :, 6, 4) Vo(:, :, 6, 5) Vo(:, :, 6, 6) Vo(:, :, 6, 7) Vo(:, :, 6, 8);
    Vo(:, :, 7, 1) Vo(:, :, 7, 2) Vo(:, :, 7, 3) Vo(:, :, 7, 4) Vo(:, :, 7, 5) Vo(:, :, 7, 6) Vo(:, :, 7, 7) Vo(:, :, 7, 8);
    Vo(:, :, 8, 1) Vo(:, :, 8, 2) Vo(:, :, 8, 3) Vo(:, :, 8, 4) Vo(:, :, 8, 5) Vo(:, :, 8, 6) Vo(:, :, 8, 7) Vo(:, :, 8, 8)];
figure, imshow(Cw, [], 'notruesize');
% -----ILETIM-----
n=randn(size(Co))*alfaN;
C=Cw+n;
%-----DETECTOR-----
% Segmentasyon, Marking Space-----
for j=1:8;
    for i=1:8;
        V(:, :, i, j)=C(boy*(i-1)+1:boy*i,en*(j-1)+1:en*j);
    end;
end;
% Correlation Coefficients-----
%      zlc= sum(sum(1/(size(C,1)*size(C,2))*(C .* Wr)));
enboy=en*boy;
for j=1:8;
    for i=1:8;
        Vmean(i, j)=sum(sum(V(:, :, i, j)))/enboy;
    end;
end;
Wrmean=sum(sum(Wr))/enboy;
Wrtilda =Wr-Wrmean;
for i=1:8;
    for j=1:8;
        Vtilda(:, :, i, j)=V(:, :, i, j)-Vmean(i, j);
    end;
end;
for i=1:8;
    for j=1:8;
        zlc(i, j)= sum(sum(Vtilda(:, :, i, j).*Wrtilda)) / sqrt(sum(sum(Vtilda(:, :, i, j).* Vtilda(:, :, i, j))))*
sum(sum(Wrtilda.*Wrtilda)));
    end;
end;
% thresholding-----

```



```

for i=1:8;
    for j=1:8;
        if (zlc(i,j)>thres);
            sonuc(i,j)=1;
        else if (zlc(i,j)<=-thres);
            sonuc(i,j)=0;
        else
            sonuc(i,j)=10;
        end;
    end;
end;

```

E_PERC__GSCALE

```

clear all;close all;clc;
mesaj=0; %mesaj 1 ya da sıfır olabilir!!!
alfa=1; %Embedding Strenght
alfaN=0.0; %iletim hattinin gurultusu
thres=0.5;
targetJNDs=4;
%-----EMBEDDER-----
% co'in Olusturulmasi-----
X=imread('uur little128.JPG');
co=double(X);
figure, imshow(co,[],'notruesize');
pic_en=128;
pic_boy=128;
% wr'in Olusturulmasi-----
wr=randn(size(co));
figure, imshow(wr,[],'notruesize');
% Wm'in Olusturulmasi-----
if (mesaj==1);
    wm=wr;
else if (mesaj == 0);
    wm=-wr;
end;
end;
% Segmentasyon-----
en=pic_en/16;
boy=pic_boy/16;
for j=1:16;
    for i=1:16;
        vo(:, :, i,j)=co(boy*(i-1)+1:boy*i,en*(j-1)+1:en*j);
        wmo(:, :, i,j)=wm(boy*(i-1)+1:boy*i,en*(j-1)+1:en*j);
        Co(:, :, i,j)=dct2(vo(:, :, i,j));
        Wmo(:, :, i,j)=dct2(wmo(:, :, i,j));
    end;
end;
% Watson Model-----
t=[1.404 1.011 1.169 1.664 2.408 3.433 4.796 6.563;
    1.011 1.452 1.323 1.529 2.006 2.716 3.679 4.939;

```



```

1.169 1.323 2.241 2.594 2.988 3.649 4.604 5.883;
1.664 1.529 2.594 3.773 4.559 5.305 6.281 7.600;
2.408 2.006 2.988 4.559 6.152 7.463 8.713 10.175;
3.433 2.716 3.649 5.305 7.463 9.625 11.588 13.519;
4.796 3.679 4.604 6.281 8.713 11.588 14.500 17.294;
6.563 4.939 5.883 7.600 10.175 13.519 17.294 21.156];
C00=zeros(16);
s=ones(8);
% for i=1:16;
%   for j=1:16;
%       C00=Co(1,1,i,j)/((en/pic_en)*(boy/pic_boy))+C00;
%   end;
% end;
for i=1:16;
    for j=1:16;
        C00(i,j)=sum(sum(Co(:, :, i,j)))/((pic_en/en)*(pic_boy/boy));
        if (C00(i,j)==0)
            C00(i,j)=1;
        end;
    end;
end;
for i=1:16;
    for j=1:16;
        tl(:, :, i,j)=t.*((Co(1,1,i,j)/C00(i,j)).^0.649);
        for k=1:8;
            for l=1:8;
                if (k==1 & l==1)
                    m(k,l)=tl(k,l,i,j);
                else
                    m(k,l)=max(tl(k,l),((abs(Co(k,l,i,j)))^0.7).*(tl(k,l,i,j)^0.3));
                end;
            end;
        end;
        s(:, :, i,j)=m;
    end;
end;
jnds=0;
p=(Wmo(:, :, 1,1)/s(:, :, 1,1)).^4;
for i=1:16;
    for j=1:16;
        jnds=(sum(sum((Wmo(:, :, i,j)/s(:, :, i,j))^4)))+jnds;
        jnds=jnds^(1/4);
    end;
end;
teorikalfa=targetJNDs/jnds;
alfa0=teorikalfa*0.2;
alfa1=teorikalfa*1.1;
for n=1:40;
    alfa=alfa0+n*(alfa1-alfa0)/40;
    cl=co;
    wa=alfa*wm;

```



```

cw=wa+c1;
w1=cw-co;
for j=1:16;
    for i=1:16;
        w1temp(:,:,i,j)=w1(boy*(i-1)+1:boy*i,en*(j-1)+1:en*j);
        W1(:,:,i,j)=dct2(w1temp(:,:,i,j));
    end;
end;
jnds=0;
for i=1:16;
    for j=1:16;
        jnds=(sum(sum((W1(:,:,i,j)./s(:,:,i,j)).^4)))+jnds;
        jnds=jnds.^(1/4);
    end;
end;
if (n==1||abs(jnds-targetJNDs)<abs(bestJNDs-targetJNDs));
    bestAlfa=alfa;
    bestJNDs=jnds;
end;
end;
% Wa'in ve Cw'nin Olusturulmasi-----
wa=bestAlfa*wm;
cw=wa+co;
figure, imshow(abs(cw),[],'notruesize');
%-----ILETIM-----
n=randn(size(co))*alfaN;
c=cw+n;
%-----DETECTOR-----
% Linear Correlation Olusturulmasi-----
zlc= sum(sum(1/(size(c,1)*size(c,2))* (c .* wr)));
% thresholding-----
if (zlc>thres);
    sonuc=1;
else if (zlc<=-thres);
    sonuc=0;
else
    sonuc=10;
end;
end;
WatermarkMyChirp

clear all; close all; clc;
message = randsrc(64,1,[0 1]);
N=15;
alfa=3;    %gurultu
X=imread('lenna128.JPG');
Co=double(X);
temp=load('gurultu.mat');
noise=temp.n;
for aagf=1:20;
    alfaAGF=(aagf-1)*3;

```



```

n=noise*alfaN;
figure, imshow(abs(Co),[],'notruesize');
title('Damgalanmamış Resim');
for deg=1:4;
degree = (deg)*10; %transform degree (0-720)
% Segmentasyon, Marking Space-----
en=128/8;
boy=128/8;
for j=1:8;
for i=1:8;
Vo(:,i,j)=Co(boy*(i-1)+1:boy*i,en*(j-1)+1:en*j);
end;
end;
% Some Marking Space Elements-----
for j=1:8;
for i=1:8;
Vo(:,i,j)=dfrft_2d(Vo(:,i,j), -degree/90, -degree/90);
end;
end;
% Chirp-----
duration=1;
[y1, f1,t1] = mychirp(6,7,duration,N); %myChirp( f1, f2, duration, fSamp )
L(:,1)=y1'*y1;
[y2, f2,t2] = mychirp(6,7,duration,N); %myChirp( f1, f2, duration, fSamp )
L(:,2)=y2'*-y2;
Wr1 = dfrft_2d(L(:,1), -degree/90, -degree/90);
Wr2= dfrft_2d(L(:,2), -degree/90, -degree/90);
figure, imshow(abs(L(:,1)),[],'notruesize');
figure, imshow(abs(Wr1),[],'notruesize');
figure, imshow(abs(Vo(:,4,4)),[],'notruesize');
% Watermark-----
for j=1:8;
for i=1:8;
if (message(i+((j-1)*8))==0);
Vm(:,i,j)=Vo(:,i,j)+(alfa.*L(:,1));
else
Vm(:,i,j)=Vo(:,i,j)+(alfa.*L(:,2));
end;
end;
end;
figure, imshow(abs(Vm(:,4,4)),[],'notruesize');
% ----- FrFT-----
for j=1:8;
for i=1:8;
S(:,i,j) = dfrft_2d(Vm(:,i,j), degree/90, degree/90);
end;
end;
% Combining-----
Cw=[S(:,1,1) S(:,1,2) S(:,1,3) S(:,1,4) S(:,1,5) S(:,1,6) S(:,1,7) S(:,1,8);
S(:,2,1) S(:,2,2) S(:,2,3) S(:,2,4) S(:,2,5) S(:,2,6) S(:,2,7) S(:,2,8);
S(:,3,1) S(:,3,2) S(:,3,3) S(:,3,4) S(:,3,5) S(:,3,6) S(:,3,7) S(:,3,8);

```



```

S(:,:,4,1) S(:,:,4,2) S(:,:,4,3) S(:,:,4,4) S(:,:,4,5) S(:,:,4,6) S(:,:,4,7) S(:,:,4,8);
S(:,:,5,1) S(:,:,5,2) S(:,:,5,3) S(:,:,5,4) S(:,:,5,5) S(:,:,5,6) S(:,:,5,7) S(:,:,5,8);
S(:,:,6,1) S(:,:,6,2) S(:,:,6,3) S(:,:,6,4) S(:,:,6,5) S(:,:,6,6) S(:,:,6,7) S(:,:,6,8);
S(:,:,7,1) S(:,:,7,2) S(:,:,7,3) S(:,:,7,4) S(:,:,7,5) S(:,:,7,6) S(:,:,7,7) S(:,:,7,8);
S(:,:,8,1) S(:,:,8,2) S(:,:,8,3) S(:,:,8,4) S(:,:,8,5) S(:,:,8,6) S(:,:,8,7) S(:,:,8,8)];
% figure, imshow(abs(Cw),[],'notruesize');
Cwat=abs(Cw);
% title('Damgalanmış Resim, Damgalama Gücü=5');
Algi= Watson(abs(Co),Cwat);
% -----ILETIM-----
C=Cw+n;
% AGF-----
h1=(1/(alfaAGF+8))*[1 1 1;1 alfaAGF 1;1 1 1];
Cn = conv2(h1,abs(C));
C = Cn(2:129,2:129);
figure, imshow(abs(C),[],'notruesize');
%-----DETECTOR-----
% Segmentasyon, Marking Space-----
Algi2(aagf)= Watson(abs(C),Cwat);
for j=1:8;
    for i=1:8;
        V(:,:,i,j)=C(boy*(i-1)+1:boy*i,en*(j-1)+1:en*j);
    end;
end;
% Inverse FrFT-----
for j=1:8;
    for i=1:8;
        R(:,:,i,j) = dfrft_2d(V(:,:,i,j), -degree/90, -degree/90);
    end;
end;
% Bit Extraction-----
for j=1:8;
    for i=1:8;
        sonuc(i+((j-1)*8)) = compcoeff(R(:,:,i,j),L(:,:,1),L(:,:,2));
    end;
end;
sonuc=sonuc';
hata(aagf)=comp(sonuc,message);
end;

```


Ek2: Algoritmelerde Kullanılan Fonksiyonlar

DFRFT

```

function y = dfrft(f,a,p)
%
% Computes discrete fractional Fourier transform
% of order a of vector x
% p (optional) is order of approximation, default N/2
%
%
N = length(f); even = ~rem(N,2);
shift = rem((0:N-1) + fix(N/2),N)+1;
f = f(:);
if (nargin == 2), p = N/2; end;
p = min(max(2,p),N-1);
E = dFRFT(N,p);
y(shift,1) = E*(exp(-j*pi/2*a*([0:N-2 N-1+even]))).'*(E'*f(shift));
function E = dFRFT(N,p)
%
% function E = dFRFT(N,a,p) returns the NxN eigenvectors of the
% Fourier transform matrix
% The optional argument p is the order of approximation
global E_saved p_saved
if (length(E_saved) ~= N | p_saved ~= p),
    E = make_E(N,p);
    E_saved = E; p_saved = p;
else
    E = E_saved;
end;
function E = make_E(N,p)
% Returns sorted eigenvectors and eigenvalues of corresponding vectors
% Construct matrix H, use approx order ord
d2 = [1 -2 1]; d_p = 1; s = 0; st = zeros(1,N);
for k = 1:p/2,
    d_p = conv(d2,d_p);
    st([N-k+1:N,1:k+1]) = d_p; st(1) = 0;
    temp = [1:k;1:k]; temp = temp(:)'/[1:2*k];
    s = s + (-1)^(k-1)*prod(temp)*2*st;
end;
% H = circulant + diagonal
col = (0:N-1)'; row = (N:-1:1);
idx = col(:,ones(N,1)) + row(ones(N,1),:);
st = [s(N:-1:2)';s(:)];
H = st(idx) + diag(real(fft(s)));
% Construct transformation matrix V
r = floor(N/2);
even = ~rem(N,2);
V1 = (eye(N-1) + flipud(eye(N-1))) / sqrt(2);
V1(N-r:end,N-r:end) = -V1(N-r:end,N-r:end);
if (even), V1(r,r) = 1; end
V = eye(N); V(2:N,2:N) = V1;

```



```

% Compute eigenvectors
VHV = V*H*V';
E = zeros(N);
Ev = VHV(1:r+1,1:r+1);      Od = VHV(r+2:N,r+2:N);
[ve,ee] = eig(Ev);           [vo,eo] = eig(Od);
% malab eig returns sorted eigenvalues
% if different routine gives unsorted eigvals, then sort first
% [d,inde] = sort(diag(ee));   [d,indo] = sort(diag(eo));
% ve = ve(:,inde');           vo = vo(:,indo');
E(1:r+1,1:r+1) = fliplr(ve);  E(r+2:N,r+2:N) = fliplr(vo);
E = V*E;
% shuffle eigenvectors
ind = [1:r+1;r+2:2*r+2]; ind = ind(:);
if (even), ind([N,N+2]) = []; else ind(N+1) = []; end
E = E(:,ind');
DFRFT2D

```

```

function out = frft_2d(imaj, a, b);
[boy,en]=size(imaj);
for i=1:boy;
    X_frft = dfrft(imaj(i,:), a); % giriş işaretinin FrFT'si
    imaj1(i,:)=X_frft;
end;
for j=1:boy;
    X_frft = dfrft(imaj1(:,j), b); % giriş işaretinin FrFT'si
    imajfrft(:,j)=X_frft;
end;
out=imajfrft;
WATSON

```

```

function out = Watson(imaj1, imaj2);
pic_en=128;
pic_boy=128;
en=pic_en/16;
boy=pic_boy/16;
for j=1:16;
    for i=1:16;
        votemp(:,i,j)=imaj1(boy*(i-1)+1:boy*i,en*(j-1)+1:en*j);
        vmtemp(:,i,j)=imaj2(boy*(i-1)+1:boy*i,en*(j-1)+1:en*j);
        Cotemp(:,i,j)=dct2(votemp(:,i,j));
        Cwtemp(:,i,j)=dct2(vmtemp(:,i,j));
    end;
end;
% Watson Model-----
t=[1.404 1.011 1.169 1.664 2.408 3.433 4.796 6.563;
    1.011 1.452 1.323 1.529 2.006 2.716 3.679 4.939;
    1.169 1.323 2.241 2.594 2.988 3.649 4.604 5.883;
    1.664 1.529 2.594 3.773 4.559 5.305 6.281 7.600;
    2.408 2.006 2.988 4.559 6.152 7.463 8.713 10.175;

```



```

3.433 2.716 3.649 5.305 7.463 9.625 11.588 13.519;
4.796 3.679 4.604 6.281 8.713 11.588 14.500 17.294;
6.563 4.939 5.883 7.600 10.175 13.519 17.294 21.156];

```

```

C00=zeros(16);
s=ones(8);
for i=1:16;
    for j=1:16;
        C00(i,j)=sum(sum(Cotemp(:,:,i,j)))/((pic_en/en)*(pic_boy/boy));
        if (C00(i,j)==0)
            C00(i,j)=1;
        end;
    end;
end;
for i=1:16;
    for j=1:16;
        tl(:,:,i,j)=t.*((Cotemp(1,1,i,j)./C00(i,j)).^0.649);
        for k=1:8;
            for l=1:8;
                if (k==1&&l==1)
                    m(k,l)=tl(k,l,i,j);
                else
                    m(k,l)=max(tl(k,l),((abs(Cotemp(k,l,i,j)))^0.7).*(tl(k,l,i,j)^0.3));
                end;
            end;
        end;
        s(:,:,i,j)=m;
    end;
end;
Dwat=0;
for j=1:16;
    for i=1:16;
        e(:,:,i,j)=Cwtemp(:,:,i,j)-Cotemp(:,:,i,j);
        d(:,:,i,j)=e(:,:,i,j)./s(:,:,i,j);
        Dwat=sum(sum(abs(d(:,:,i,j))^4))^(1/4)+Dwat;
    end;
end;
Dwat=sum(sum(abs(d(:,:,i,j))^4))^(1/4)+Dwat;
out=Dwat;

```

MyChirp

```

function [Chirp1d,fi,t] = myChirp( f1, f2, duration, fSamp )
mu = ((f2 - f1)/(2 * duration));
Ts = 1/(fSamp); %sampling period
t = [0 : Ts : duration];
psi = 2*pi*(f1.*t + mu.*t.^2 + 1/4);
Chirp1d = real(exp(j*psi)); %vector samples of the chirp signal
fi = ((f2 - f1)/duration)*t + f1; %Calculate Instantaneous Frequency

```


ÖZGEÇMİŞ

Doğum tarihi	11.03.1983	
Doğum yeri	İstanbul-Kadıköy	
Lise	1994–2002	Kadıköy Anadolu Lisesi
Lisans	2002–2006	Gebze Yüksek Teknoloji Enstitüsü Mühendislik Fakültesi Elektronik Mühendisliği Bölümü
Yüksek Lisans	2006–2008	Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü Elektronik ve Haberleşme Müh. Anabilim Dalı, Elektronik Programı
Çalıştığı Kurumlar		
	2007-2008	Türk Telekom
	2008-Devam Ediyor	Türkiye İş Bankası

