

35420

KARADENİZ TEKNİK ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ

ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

ELEKTRONİK MÜHENDİSLİĞİ PROGRAMI

VIDEO SİNYALLERİNİN DÜŞÜK VERİ İLETİM HIZLARINDA KODLANMASI İÇİN
BLOK KARŞILAŞTIRMAYA DAYALI YENİ BİR HAREKET KESTİRİM YÖNTEMİ

Elektronik Yük. Müh. Ali GANGAL

Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsünde
"Doktor"
Ünvanının Verilmesi İçin Kabul Edilen Tezdir

Tezin Enstitüye Verildiği Tarih : 25 Ekim 1994

Tezin Sözlü Savunma Tarihi : 17 Mart 1995

Tezin Danışmanı : Yrd. Doç. Dr. Temel KAYIKÇIOĞLU *T. Kayıkçioğlu*

Jüri Üyesi : Prof. Dr. Avni MORGÜL *A. Morgül*

Jüri Üyesi : Yrd. Doç. Dr. Rıfat YAZICI *R. Yazıcı*

Enstitü Müdürü : Prof. Dr. Temel SAVAŞKAN *T. Savaşkan*

Ekim 1994

TRABZON

ÖNSÖZ

Bu tez, Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsü Elektrik-Elektronik Mühendisliği Anabilim Dalı, Elektronik Mühendisliği Doktora Programında yapılan bir çalışmadır. "Video Sinyallerinin Düşük Veri İletim Hızlarında Kodlanması İçin Blok Karşılaştırmaya Dayalı Yeni Bir Hareket Kestirim Yöntemi" konulu bu çalışmada, güncel ve çok önemli olan ekonomik, kayıpsız ve kaliteli görsel iletişim sorunlarına çözüm aranmıştır.

Bu konudaki çalışmamda bana yardımlarını esirgemeyen yöneticim sayın Yrd. Doç. Dr. Temel KAYIKÇIOĞLU' na, kuramsal bilgilerinden yararlandığım sayın Prof. Dr. Mümtaz YILMAZ' a ve Elektrik-Elektronik Bölüm' ü elemanlarına teşekkürlerimi sunarım.

Trabzon, Ekim 1994

Ali GANGAL

İÇİNDEKİLER

	<u>Sayfa No</u>
ÖZET	V
SUMMARY	VI
ŞEKİL LİSTESİ	VII
TABLO LİSTESİ	X
SEMBOL LİSTESİ	XII
KISALTMA LİSTESİ	XIII
1. GENEL BİLGİLER	1
1.1. Giriş	1
1.2. Video Sinyallerini Kodlama Yöntemlerinin İncelenmesi	5
1.2.1. Örnekleme	5
1.2.2. Seviyelendirme	7
1.2.3. Kodlama	13
1.2.3.1. Kestirim Kodlaması	13
1.2.3.2. Dönüşüm kodlaması	18
1.2.3.3. Melez kodlama	24
1.2.3.4. Hareket Kestirime Dayalı Kodlama	27
1.2.3.5. Modele Dayalı Kodlama	30
1.3. Video Sinyallerinde Hareket Kestirme Teknikleri	31
1.3.1. Türevsel Hareket Kestirme Teknikleri	31
1.3.2. Konum Özyineli Hareket Kestirme Teknikleri	41
1.3.3. Fourier Dönüşümüne Dayalı Hareket Kestirme Teknikleri	45
1.3.4. Blok Karşılaştırmaya Dayalı Hareket Kestirme Teknikleri	47
1.3.4.1. Sabit Boyutlu Blok Karşılaştırma	47
1.3.4.2. Hızlı Blok Karşılaştırma Yöntemleri	50
2. YAPILAN ÇALIŞMALAR	57
2.1. Giriş	57
2.2. Bloksal Hareket Kestirime Dayalı Kodlama İçin Yeni Bir Yöntem.....	57
2.3. Önerilen Yöntemin Video Telefon Sinyallerinin Kodlanmasında Kullanılması	62
2.4. Önerilen Yöntemin İşlem Karmaşıklığı	67

	<u>Sayfa No</u>
2.5. Yapılan Deneysel Çalışmalar ve Kullanılan Görüntü Dizileri	67
3. BULGULAR	74
3.1. Blok Karşılaştırmada Uygun Blok Boyutlarının Araştırılması	74
3.2. Değişik Hareket Kestirme Tekniklerinin Performanslarının Karşılaştırmalı Olarak İncelenmesi	80
3.2.1. Gürültünün Kestirim Hatası Entropisine Olan Etkileri	80
3.2.1.1. Sentetik Görüntü Dizisi İçin Yapılan İncelemeler	80
3.2.1.2. Standart Görüntü Dizisi İçin Yapılan İncelemeler	82
3.2.1.3. Yazarın Kaydettiği Görüntü Dizisi İçin Yapılan İncelemeler	84
3.2.2. Değişik Yöntemlere Göre Görüntü Elemanlarının Kestirilebilme Oranları .	85
3.2.3. Değişik Yöntemlerin YÇF Olasılık Yoğunluk İşlevlerinin Karşılaştırılması.	87
3.2.4. Değişik Yöntemlerde Kestirim Hatasının Bir önceki Orijinal Görüntüye Göre Sinyal/Gürültü Oranı	88
3.2.5. Görsel (Subjektif) Sonuçlar	90
3.2.6. Değişik Yöntemlerin İşlem Sayısı Bakımından İncelenmesi	94
3.2.7. Sonuçlar	95
3.3. Önerilen Yeni Hareket Kestirme Yönteminin Kodlamadaki Performansının Karşılaştırmalı Olarak İncelenmesi	95
3.3.1. Yöntemin Veri Azaltma Performansının İncelenmesi	95
3.3.1.1. Toplam Entropinin Hareket Vektörleri ve YÇF Sinyali Tarafından Paylaşımı	96
3.3.1.2. AKD Katsayılarının ve Hareket Vektörlerinin İstatistiksel Kodlanması Sonucunda Ortaya Çıkan Veri Hızları	99
3.3.2. Kodu Çözülmüş Görüntülerde Kalitenin İncelenmesi	103
3.3.2.1. Hata Değişintisi	104
3.3.2.2. Sinyal/Gürültü Oranı	105
3.3.2.3. Görsel İncelemeler	107
4. İRDELEME VE TARTIŞMA	112
5. SONUÇLAR	114
6. ÖNERİLER	116
7. KAYNAKLAR	117
8. ÖZGEÇMİŞ	124

ÖZET

VIDEO SINYALLERİNİN DÜŞÜK VERİ İLETİM HIZLARINDA KODLANMASI İÇİN BLOK ARŞILAŞTIRMAYA DAYALI YENİ BİR HAREKET KESTİRİM YÖNTEMİ

İletim kanallarının ve depolama ortamlarının ekonomik kullanımı veya video telefon, video konferans amaçlı haberleşmeler için video sinyallerinde etkili veri hızı azaltma teknikleri kullanılması gereklidir.

Video görüntülerinin düşük veri iletim hızlarında kodlanması uzamsal ve zamansal eksende artıklığın yok edilmesini gerektirir. Veri hızı azaltımı, cisimlerin hareketlerinin bir işlevi olarak tanımlanabilen görüntü çerçeveleri arasındaki artıklığın azaltılmasıyla yapılabilir.

Bu çalışmada, sayısal video sinyallerinde çeşitli veri hızı azaltma teknikleri incelenmekte ve video telefon sinyallerinin kodlanmasında yeni bir hareket kestirim yöntemi sunulmaktadır. Bu yöntem, önceden tanımlanmış değişik şekillere girebilen blokların ardışıl iki görüntü çerçevesi arasında karşılaştırılmasına dayanmaktadır. Hareket kestirme biriminde, en az veri hızını sağlayan blok şekli uyarlamalı olarak seçilmektedir. Elde edilen veriler, ayrık kosinüs dönüşümü ve istatistiksel kodlama teknikleriyle biraz daha azaltılmaktadır. Önerilen yöntemde, karşılaştırma ölçütü olarak, performansı ortalama kare hata ölçütünününe çok yakın olan ve daha az işlem gerektiren ortalama mutlak hata ölçütü kullanılmıştır. Hareket yönlerinin hızlı olarak saptanmasında basitliği nedeniyle üç adımlı yönlü arama tekniği kullanılmıştır.

Önerilen yöntemin performansı, sentetik olarak üretilmiş ve gerçek video görüntüleri için ayrı ayrı test edilerek diğer yöntemlerinkine karşılaştırılmıştır. Sonuçlardan, teknik yazında yakın zamana kadar belirtilenlerden daha iyi video kalitesi elde edildiği görülmüştür.

Anahtar kelimeler: Video kodlama, Video telefon, Veri hızı azaltma,
Hareket kestirme.

SUMMARY

A NEW MOTION ESTIMATION METHOD BASED ON BLOCK MATCHING FOR LOW BIT RATE VIDEO CODING

For economical using of transmission channels and storage media or for videophone communications, it is required that using of effective bit rate reduction techniques for video signals.

Coding video signals in order to reduce transmission bit rates requires removal of the redundancy in spatial and temporal domain. Bit rate reduction can be achieved by utilizing the interframe redundancy which can be characterized by motion field of objects.

In this study, various bit rate reduction techniques for video signals are reviewed and a new motion estimation method is presented for videophone coding. This method is based on matching of predefined a number of different block shapes between successive frames. In motion estimation stage, the block shape which requires the least data rate is adaptively chosen. This data is further reduced by applying discrete cosine transform and statistical coding techniques. The mean of the absolute difference criterion is selected as the matching criterion due to its performance close to that of mean square error criterion and less computational effort. Three-step directed search method is used due to its simplicity.

The performance of the method has been tested on computer generated and real video sequences and compared to those of the other methods. The results demonstrate much better video quality than the reported in recent literature.

Key Words: Video Coding, Videophone, Bit Rate Reduction,
Motion Estimation.

ŞEKİL LİSTESİ

	<u>Sayfa No</u>
Şekil 1. Bir seviyelendiricinin giriş ve çıkış işlevi.....	7
Şekil 2. Çerçeveler arası ve çerçeve içi sinyal için seviyelendirme karakteristikleri	
a) Çerçeveler arası blok için	8
b) Çerçeve içi blok için	8
Şekil 3. Görüntülerin örneklenmesi ve seviyelendirilmesi.....	9
Şekil 4. İki örnekli bir blokun iki boyutlu gösterimi	11
Şekil 5. Blok seviyelendirmeye iki boyutlu bir örnek.....	12
Şekil 6. Bir TVKM verici ve alıcı dizgesi.....	13
Şekil 7. Uyarlamalı kazanç kontrolü.....	16
Şekil 8. Uyarlamalı sınıflandırma	17
Şekil 9. Uyarlanı TVKM dizgesi	18
Şekil 10. Bir boyutlu dönüşüm kodlayıcı	19
Şekil 11. Melez kodlama.....	25
Şekil 12. Hareket dengelemeli bir kodlayıcı ve kod çözücü.....	29
Şekil 13. Modele dayalı kodlama verici ve alıcı.....	30
Şekil 14. Optik akı kısıtının geçersiz olduğu durum	33
Şekil 15. Türevsel hareket kestirme tekniği	39
Şekil 16. Aradeğerleme işlemi	44
Şekil 17. Bir konum özyineli hareket kestiricinin ilkesel şeması.....	45
Şekil 18. Faz ilişkisi tekniği ile hareket kestirimi	47
Şekil 19. Blok karşılaştırma.....	48
Şekil 20. BSBA yönteminin iki değişkenli bir dışbükey işleve uygulanışı	51
Şekil 21. Bir seferde bir arama yöntemiyle blok karşılaştırma	51
Şekil 22. EYA yönteminin iki değişkenli bir dışbükey işleve uygulanışı	52
Şekil 23. EYA yöntemiyle blok karşılaştırma.....	53
Şekil 24. Logaritmik arama yöntemiyle blok karşılaştırma	54
Şekil 25. Düzeltilmiş logaritmik arama yöntemiyle blok karşılaştırma.....	55
Şekil 26. Üç adımlı yönlü arama yöntemiyle blok karşılaştırma	56

Şekil 27. Görüntünün bir bölgesi için alt blokların bir önceki görüntüde arandıkları alanlar	58
Şekil 28. Aynı hareket vektörüne sahip alt blokların birleştirilmesi	59
Şekil 29. Dört adet alt blok kullanılması durumunda ortaya çıkması mümkün olan bölge biçimleri	60
Şekil 30. Video telefon sinyalleri kodlayıcı ve kod çözücü blok şeması	63
Şekil 31. "I" tipi (çerçeve içi) ve "P" tipi (çerçeveler arası) kodlanan görüntü çerçeveleri	64
Şekil 32. Çerçeve içi blokların DC bileşenlerinin fark kodlaması yapılması	64
Şekil 33. AKD katsayılarının zigzag taranması	65
Şekil 34. Sentetik görüntü dizisinin ilk 8 çerçevesi	69
Şekil 35. "Claire" görüntü dizisinin ilk 24 çerçevesi	70
Şekil 36. "Miss America" görüntü dizisinin ilk 24 çerçevesi	71
Şekil 37. "Trevor" görüntü dizisinin ilk 24 çerçevesi	72
Şekil 38. Yazarın görüntü dizisinin ilk 8 çerçevesi	73
Şekil 39. Arama uzaklığına ve hareket kestirmede kullanılan blok boyutlarına göre elde edilen fark (YÇF) görüntüleri (sentetik görüntü dizisi)	78
Şekil 40. Arama uzaklığına ve hareket kestirmede kullanılan blok boyutlarına göre elde edilen fark (YÇF) görüntüleri ("Claire" görüntü dizisi)	79
Şekil 41. Sentetik görüntü dizisinde kestirim hatasının entropisi	82
Şekil 42. "Claire" görüntü dizisinde kestirim hatasının entropisi	84
Şekil 43. Kestirilebilen görüntü elemanı değerlerinin yüzdesi (eşik ≤ 3)	
a) Sentetik görüntü dizisi	86
b) "Claire" görüntü dizisi	86
Şekil 44. Değişik hareket kestirme teknikleri kullanılarak elde edilen YÇF sinyali için olasılık yoğunluk işlevleri. a) Sentetik görüntü dizisi	87
b) "Claire" görüntü dizisi	88
Şekil 45. Kestirim hatasının bir önceki orijinal görüntüye göre sinyal/gürültü oranı (SGO) a) Sentetik görüntü dizisi	89
b) "Claire" görüntü dizisi	89
Şekil 46. Bazı fark görüntüleri (sentetik görüntü dizisi, 7. çerçeve-4. çerçeve)	91
Şekil 47. Bazı fark görüntüleri ("Claire" görüntü dizisi, 7. çerçeve-4. çerçeve)	92

Şekil 48. Bazı fark görüntüleri (yazarın görüntü dizisi, 7. çerçeve-4. çerçeve)	93
Şekil 49. Sentetik görüntü dizisinin kodlanmasında ortaya çıkan toplam veri hızı	102
Şekil 50. "Claire" görüntü dizisinin kodlanmasında ortaya çıkan toplam veri hızı	103
Şekil 51. Yazarın görüntü dizisinin kodlanmasında ortaya çıkan toplam veri hızı	103
Şekil 52. Alıcıda elde edilen görüntüler ile orijinal görüntüler arasındaki hataların değişimleri ("Trevor" görüntü dizisi)	104
Şekil 53. Alıcıda elde edilen görüntülerin sinyal/gürültü oranları ("Trevor" görüntü dizisi)	106
Şekil 54. Alıcıda kodu çözülen görüntüler ve hata görüntüleri. a): Orijinal görüntü. b)-d): Alıcıda kodu çözülen görüntülerin kullanılan blok boyutlarına göre sinyal/gürültü oranları. e)-g): 10 kat kuvvetlendirilmiş hata görüntüleri (Cl07).....	108
Şekil 55. Alıcıda kodu çözülen görüntüler ve hata görüntüleri. a): Orijinal görüntü. b)-d): Alıcıda kodu çözülen görüntülerin kullanılan blok boyutlarına göre sinyal/gürültü oranları. e)-g): 10 kat kuvvetlendirilmiş hata görüntüleri (Miss07).....	109
Şekil 56. Alıcıda kodu çözülen görüntüler ve hata görüntüleri. a): Orijinal görüntü. b)-d): Alıcıda kodu çözülen görüntülerin kullanılan blok boyutlarına göre sinyal/gürültü oranları. e)-g): 10 kat kuvvetlendirilmiş hata görüntüleri (Trev07).....	110
Şekil 57. Alıcıda kodu çözülen görüntüler ve hata görüntüleri. a): Orijinal görüntü. b)-d): Alıcıda kodu çözülen görüntülerin kullanılan blok boyutlarına göre sinyal/gürültü oranları. e)-g): 10 kat kuvvetlendirilmiş hata görüntüleri (A07).....	111

TABLO LİSTESİ

	<u>Sayfa No</u>
Tablo 1. Video sinyallerinde parlaklık bileşenine ait yaklaşık örnekleme ve seviyelendirme sayıları, veri iletim hızları ve hedeflenen hız değerleri	10
Tablo 2. Arama uzaklığına göre bozulma işlevinin (veya ilişki işlevinin) hesaplanma sayısı	49
Tablo 3. Blok boyutuna göre hareket vektörleri sayısı (CIF ve QCIF boyutlu bir görüntü için)	49
Tablo 4. Alt blok sayısına göre ortaya çıkan farklı bölge sayısı ve bölge başına düşen maksimum entropi	61
Tablo 5. Arama uzaklığına göre ortalama mutlak hatanın azalma yüzdesinin (%), YÇF değişintisinin (σ^2 : variance), YÇF entropisinin (E_f), Hareket vektörleri entropisinin (E_h) ve toplam entropinin (E_t) değişimi (sentetik dizi)	75
Tablo 6. Arama uzaklığına göre ortalama mutlak hatanın azalma yüzdesinin (%), YÇF değişintisinin (σ^2 : variance), YÇF entropisinin (E_f), Hareket vektörleri entropisinin (E_h) ve toplam entropinin (E_t) değişimi ("Claire" dizisi)	76
Tablo 7. Arama uzaklığına göre ortalama mutlak hatanın azalma yüzdesinin (%), YÇF değişintisinin (σ^2 : variance), YÇF entropisinin (E_f), Hareket vektörleri entropisinin (E_h) ve toplam entropinin (E_t) değişimi (yazarın dizisi)	77
Tablo 8. Değişik hareket kestirme tekniklerinin Gauss gürültüsü eklenmiş sentetik görüntüler üzerinde göreceli hareket izleme performansı, YÇF'nin entropisi (7. çerçeve ve 4. çerçeve arasındaki hareket)	81
Tablo 9. Değişik hareket kestirme tekniklerinin Gauss gürültüsü eklenmiş "Claire" görüntü dizisi üzerinde göreceli hareket izleme performansı, YÇF'nin entropisi (7. çerçeve ve 4. çerçeve arasındaki hareket)	83

Tablo 10. Değişik hareket kestirme tekniklerinin Gauss gürültüsü eklenmiş yazarın görüntü dizisi üzerinde göreceli hareket izleme performansı, YÇF'nin entropisi (7. çerçeve ve 4. çerçeve arasındaki hareket)	85
Tablo 11. Değişik hareket kestirme yöntemlerinde 16x16 boyutlu bir bölgenin kestirimi için maksimum işlem sayısı	94
Tablo 12. YÇF ve hareket vektörlerinin toplam entropisi (Sentetik görüntü dizisi) ..	96
Tablo 13. YÇF ve hareket vektörlerinin toplam entropisi ("Claire" görüntü dizisi) ..	97
Tablo 14. YÇF ve hareket vektörlerinin toplam entropisi ("Miss America" görüntü dizisi)	97
Tablo 15. YÇF ve hareket vektörlerinin toplam entropisi ("Trevor" görüntü dizisi) ..	98
Tablo 16. YÇF ve hareket vektörlerinin toplam entropisi (yazarın görüntü dizisi) ...	98
Tablo 17. Seviyelendirilmiş AKD katsayılarının ve hareket vektörlerinin kodlanması sonucu ortaya çıkan veri hızları (sentetik görüntü dizisi)	100
Tablo 18. Seviyelendirilmiş AKD katsayılarının ve hareket vektörlerinin kodlanması sonucu ortaya çıkan veri hızları ("Claire" görüntü dizisi)	100
Tablo 19. Seviyelendirilmiş AKD katsayılarının ve hareket vektörlerinin kodlanması sonucu ortaya çıkan veri hızları ("Miss America" görüntü dizisi)	101
Tablo 20. Seviyelendirilmiş AKD katsayılarının ve hareket vektörlerinin kodlanması sonucu ortaya çıkan veri hızları ("Trevor" görüntü dizisi)	101
Tablo 21. Seviyelendirilmiş AKD katsayılarının ve hareket vektörlerinin kodlanması sonucu ortaya çıkan veri hızları (yazarın görüntü dizisi)	102
Tablo 22. Kodu çözülmüş görüntüler ile orijinal görüntüler arasındaki hata değişimi (σ^2) değerleri	105
Tablo 23. Kodu çözülmüş görüntülerin sinyal/gürültü oranları (dB)	106

SEMBOL LİSTESİ

- f** : frekans
x : yatay uzamsal koordinat
y : düşey uzamsal koordinat
m : örnekleme noktası üzerinde yatay yönde örnek numarası
n : örnekleme noktası üzerinde düşey yönde örnek numarası
e : kestirim hatası
 σ^2 : değişim (variance)
 μ : ortalama değer
V : dönüşüm katsayısı
v : hız
d : hareket vektörü
B : veri hızı
F(.) : Fourier dönüşümü
 $\delta(.)$: birim vuruş işlevi
I(.) : görüntü elemanı parlaklık değeri
 $\hat{I}(.)$: görüntü elemanı parlaklık kestirim değeri
D(.) : bozulma işlevi (distortion function)
C(.) : Çapraz ilişki işlevi (cross correlation function)

KISALTMA LİSTESİ

AFD	: Ayrık Fourier Dönüşümü
AKD	: Ayrık Kosinüs Dönüşümü
BSBA	: Bir Seferde Bir Arama
CCITT	: Uluslararası Telefon ve Telgraf Danışma Yarıkurulu (The International Telegraph and Telephone Consultative Committee)
CIF	: Ortak Ara Formatı (Common Intermediate Format)
ÇA	: Çapraz Arama
ÇF	: Çerçeveler Farkı
DA	: Dikgen Arama
DLA	: Düzeltilmiş Logaritmik Arama
EYA	: Eşlenik Yönlere Arama
Faz	: Faz İlişkisi Tekniği
HDTV	: Yüksek Tanımlı Televizyon (High Definition Television)
HFD	: Hızlı Fourier Dönüşümü
HKY	: Hareket Kestirimi Yok
ISDN	: Tümlşik Hizmetler Sayısal Şebekesi (Integrated Services Digital Network)
JPEG	: Birleşik Fotoğraf Uzmanları Grubu (Joint Photographic Experts Group)
KLD	: Karhunen Loeve Dönüşümü
KÖ	: Konum Özyineli Hareket Kestirme
LA	: Logaritmik Arama
MPEG	: Hareketli Resim Uzmanları Grubu (Moving Pictures Experts Group)
NTSC	: Ulusal Televizyon Sistemi Kurulu (National Television System Committee)
OKH	: Ortalama Kare Hata
OMH	: Ortalama Mutlak Hata
PAL	: Faz -Değişim-Satır (Phase-Alternation-Line)
QCIF	: Çeyrek Ortak Ara Formatı (Quarter-Common Intermediate Format)
SAF	: Satır Fark Sinyali
SBK	: Sıralı Aramalı Blok Karşılaştırma

SÜF	: Sütun Fark Sinyali
TAKD	: Ters Ayrık Kosinüs Dönüşümü
THK	: Türevsel Hareket Kestirimi
TVKM	: Türevsel Vuruş Kodlamalı Modülasyon
ÜAYA	: Üç Adımlı Yönlü Arama
VKM	: Vuruş Kodlamalı Modülasyon
VLSI	: Çok-Büyük-Çapta-Tümleşim (Very-Large-Scale-Integrated)
YÇF	: Yerdeğiřtirmiş Çerçevesel Farkı



1. GENEL BİLGİLER

1.1. Giriş

Sayısal haberleşme sistemleri veri işleme esnekliği, daha az yanılırlı iletim gibi özelliklerinden dolayı analog haberleşme sistemlerine bir üstünlük sağlamaktadırlar. Bu nedenle, video sinyallerinin de çeşitli amaçlarla sayısal olarak işlenmesi, depolanması ve iletilmesi benimsenmektedir. Ancak sayısal video sinyallerinin iletilmesi çok büyük veri hızları, depolanması da çok büyük bellek boyutları gerektirmektedir. Bellek kapasitelerinin ve iletim kanallarının band genişliklerinin sınırlı olması nedeniyle video sinyallerinin düşük veri hızlarında iletilmesi ve depolanması istenmektedir. Bu nedenle, video sinyallerinin önemli bir kalite kaybı olmaksızın veri azaltma teknikleri kullanılarak sıkıştırılması konusu önem kazanmaktadır.

Video sinyalleri çok yüksek oranda artıklık (redundance) içermektedirler. Artıklık, iletilmesine gerek duyulmayan bilgidir ve sıkıştırma işlemleriyle yok edilebilir. Görüntü kalitesinde önemli bir kayıp oluşturmadan ya da kabul edilebilir kayıplar oluşturarak video sinyallerini daha az sayıda veriyle temsil etmek için yapılan işlemlere görüntü sıkıştırma veya video sinyallerinde veri azaltma denilmektedir [1].

Bir saniyedeki görüntü çerçevesi (frame) sayısı, her çerçevedeki görüntü elemanlarının (pixel) sıklığı ve her görüntü elemanının alabileceği seviye sayısı, ayrıntının (detail) ve görüntü kalitesinin bir ölçüsüdür. Görüntü kalitesini doğrudan etkileyen bu büyüklükler band genişliğiyle de doğru orantılıdır. Kullanım amacı göz önüne alınarak band genişliğini sınırlı tutmak amacıyla, farklı video sistemleri farklı kalitede görüntüler kullanmaktadırlar. Farklı sistemlerin kullandıkları görüntü nitelikleri hareket içerikleri bakımından da farklıdır. Örneğin video telefonda genellikle yavaş hareket içeren baş ve omuz görüntüleri yaygındır. Bu nedenle video telefon sinyallerinde artıklık çok fazladır. Televizyon görüntülerinde ise her türlü hareket içeriği bulunduğu için, veri istatistiği de daha değişkendir ve artıklık daha azdır. Bu nedenle bir video sistemi için kullanılacak veri azaltma yöntemini, o video sisteminin özelliklerine göre saptamak uygundur.

Görüntü sıkıştırma işlemleri video sinyaline, örnekleme (sampling), seviyelendirme (quantization) ve kodlama (coding) aşamalarında değişik şekillerde uygulanabilir. Frekans bandı sınırlandırılmış video sinyali, içerdiği bilgiyi kaybetmeyecek ve "örtüşme" (aliasing) meydana gelmeyecek şekilde seyrek örneklenebilir. Örnekleme kuramında,

örnekleme hızının bu alt sınırına "Nyquist hızı" denilmektedir. Buna göre bir sinyal, en az band genişliğinin iki katı frekansla örneklenmelidir. Ancak video sinyallerinin fazla miktarda artıklık içermeleri ve frekans boyutundaki bazı özellikleri nedeniyle bu sınırların da altına bozulma ortaya çıkarmadan inmek mümkün olabilmektedir. Ayrıca örneklenmiş değerler bir alt örneklemeyle daha da seyreltilerek, bunlar aradeğerleme (interpolation) teknikleriyle yeniden yerlerine konabilmektedir [1].

Video sinyali örneklendikten sonra her örnek değeri seviyelendirilmektedir. Seviye sayısı band genişliğiyle logaritmik olarak orantılıdır ve sınırlandırılmak zorundadır. İnsan gözünün parlaklık farklarını algılama özelliğinden yararlanılarak seviye sayısı sınırlandırılmaktadır. 256 seviye yeterli kabul edilmektedir. Seviyelendirme işlemi video sinyaline doğrudan veya frekans dönüşümü alındıktan sonra da uygulanabilmektedir. Daha az sayıda seviye kullanmak için doğrusal seviyelendirme yerine eğrisel (nonlinear) seviyelendirme de yapılabilmektedir [1].

Seviyelendirme işleminden sonra veriler kodlama aşamasına getirilir. Video sinyallerinin kodlanmasında kullanılan teknikler, iki ana gruba ayrılabilir. Bunlar, kestirime dayalı (predictive) teknikler ve dönüşüm (transform) teknikleridir. Kestirime dayalı tekniklerdeki amaç, zamansal veya uzamsal boyutta ardışıl görüntü elemanları arasındaki ortak artıklığı (mutual redundance) yok edip sadece yeni bilgileri (veya kestirim hatalarını) seviyelendirmek ve kodlamaktır. Dönüşüm almaya dayalı tekniklerde ise, görüntü elemanlarının üniter (unitary) dönüşümü alınarak çok daha az sayıda olan (idealde birbirisiyle ilişkisiz) katsayılar elde edilir ve bunlar seviyelendirilerek kodlanır. Dönüşüme dayalı tekniklerinin verimliliği, kestirime dayalı tekniklerde olduğu gibi, görüntü elemanları arasındaki çapraz ilişki (cross correlation) katsayılarının yüksek olmasına bağlıdır [1].

Kestirime ve dönüşüme dayalı kodlama teknikleri veri hızı azaltmada tek başlarına yeterince iyi sonuçlar verememektedirler [1]. Bu nedenle iki tekniği de içeren melez (hybrid) yöntemler daha çok kullanılmaktadır. Melez yöntemlerde genel olarak kestirim hatalarının üniter dönüşümleri alınmakta ve ortaya çıkan katsayılar seviyelendirilerek kodlanmaktadır. Kestirim hataları, görüntünün yüksek ayrıntı ve hareket içeren alanlarında daha fazla olmaktadır. Bu nedenle yüksek ayrıntı ve hareket içeren görüntüler için daha fazla veri iletilmesi gerekmektedir. Eğer kullanılan sistem, Uluslararası Telefon ve Telgraf Danışma Yürütürü' nun (CCITT) H.261 tavsiyesinde belirtildiği gibi, Tümlleşik Hizmetler Sayısal Şebekesinin (ISDN) bir kanalını (64kbit/s) veya iki kanalını (112 kbit/s) kullanan ve kesin bir hız kısıtı altında çalışan bir video telefon sistemi ise [2], kestirim hatalarının tümünü iletmek mümkün olmadığı için bu sefer görüntünün kalitesinden ödün vermek gerekmektedir.

Veri iletim hızını daha çok azaltabilme ve görüntü kalitesini koruyabilmeye ilgili yukarıda sadece bir kısmı anlatılan sorunları gidermek için, görüntü elemanlarının

zamansal ilişkisini etkili olarak kullanabilen teknikler gereklidir. Bu amaçla, görüntü elemanlarının hareketlerini kestirerek zamansal ilişkiyi azaltan teknikler kullanılmaktadır ve bu tekniklere "hareket dengeleme" (motion compensation) teknikleri denilmektedir [1]. Hareket dengeleme tekniği içeren kodlama işlemine hareket dengelemeli kodlama "Motion compensated coding" denilmektedir.

Uygulamada kolay olması bakımından, ekrandaki bir cismin hareketi görüntü çerçeveleri arasında küçük parçaların yerdeğiştirmesi (piecewise displacements) şeklinde yaklaştırılabilir. Hareket kestirimiyle bulunan yerdeğiştirme vektörü, hareket dengelemeli bir kestiriciyi yönlendirmek için kullanılır. Kodlamanın başarılı olabilmesi için hareket kestiricinin az yanılı, hızlı ve gürültüden bağımsık olması gerekmektedir. Bu nedenle hareket kestiricinin tasarımı iyi yapılmalıdır.

Hareket dengelemeli kodlamanın bir başka yararı da, seyrek örnekleme işleminde ortaya çıkmaktadır. Örnekleme işleminde, örnekler seyrek alınarak veri azaltılmaktadır. Kodlayıcıda atlanan ve alıcıya gönderilmeyen örnek değerleri ise, alıcıya gönderilen hareket bilgisinden yararlanarak yeniden elde edilmektedir. Böylece seyrek örnekleme nedeniyle ortaya çıkan bilgi kaybı azaltılmaktadır.

Kestirim hatalarının veya seçilen görüntü elemanları değerlerinin frekans dönüşümü alınarak görüntü verileri daha da azaltılır. Karhunen Loeve dönüşümü, dönüşüm katsayıları arasında az ilişki bulunması ve minimum ortalama kare hatayı sağlaması gibi verimli kodlamanın genel şartlarını en iyi şekilde yerine getiren özelliklere sahiptir. Ancak çok fazla işlem gerektirmesi nedeniyle pratik olarak uygulaması zordur [1]. Bu nedenle, bu dönüşüme yakın performans gösteren kosinüs dönüşümü genellikle kullanılır. Kosinüs dönüşümü alındıktan sonra ortaya çıkan dönüşüm katsayıları, göz duyarlığına da bağlı olarak seviyelendirilir [3], [4], [5], [6], [7], [8], [9], [10], [11].

Kodlamada, genellikle azaltılmış sayıda sıfırdan farklı katsayıyla temsil edilen görüntü blokları veya dönüşüm katsayıları önce seğırtim boyu (Run-Length) kodlanır. Daha sonra bu veriler hareket bilgisi ve (varsa) diğer yan bilgilerle beraber bir entropi (entropy) kodlama veya aritmetik kodlama tekniğiyle kodlanır [12], [13]. Entropi kodlama, istatistiksel bir kodlama yöntemidir ve veri gruplarının çok sık rastlananlarını kısa kod sözcüğüyle, az rastlananlarını da uzun kod sözcüğüyle gösterme prensibine dayanır. Dönüşüme dayalı kodlama gibi bazı veri azaltma yöntemlerinde tersinemez bilgi kaybı olurken entropi kodlama kayıpsız bir kodlamadır. Sadece kayıpsız kodlamalarla veri hızını istenen düzeylere indirebilmek mümkün değildir. Bu nedenle video sinyallerinin kodlanmasında genellikle istatistiksel, kestirim ve dönüşüm yöntemleri birlikte kullanılır [1].

Veri hızı azaltmada modele dayalı kodlama ve özellik belirleyerek kodlama gibi teknikler de günümüzde kullanım aşamasına gelmiştir [6], [14], [15]. Özellikle yüksek

hızlı mikroişlemcilerin ve Çok-Büyük-Çapta Tümlleşim (VLSI: very-large-scale integrated) teknolojisinin gelişmesiyle bu tekniklerin de daha ümit verici olacağı açıktır.

Çerçeve içi kodlama (intraframe coding), durağan görüntülerin sıkıştırılması amacıyla yapılmaktadır. Durağan görüntülerin sıkıştırılması konusunda da günümüzde birçok çalışma yapılmakta olup, Uluslararası Standartlar Örgütünün (ISO) oluşturduğu bir yarıkurul, Birleşik Resim Uzmanları Grubu (JPEG), sayısal durağan görüntülerin standartları üzerinde çalışmalar yapmış ve 3 ile 50 kat azaltma oranları elde edilmiştir [16]. Peş peşe gelen görüntü çerçeveleri, herbiri durağan bir görüntü gibi düşünülüp çerçeve içi kodlama yapılabilir. Ancak çerçeve içi kodlama tek başına kullanıldığında, özellikle video telefon sinyalleri için istenen veri hızlarına insan gözünü rahatsız etmeyen bozulmalar ortaya çıkarmadan inmek mümkün olamamaktadır. Bu nedenle video telefon sinyallerinin kodlanmasında sadece kestirim hataları kodlanmamakta, hataların yığılmasını önlemek amacıyla belirli aralıklarda çerçeve içi kodlama yapılmaktadır.

ISO güdümünde oluşturulan bir başka yarıkurul, Hareketli Resim Uzmanları Grubu (MPEG) videoyu bilgisayarda işlenecek bir veri haline getirmek için standartlar oluşturmaya başlamıştır. MPEG sayısal video görüntülerinin 1,5Mbit/s nin altında kalacağını öngörmekte, özellikle de sayısal bilgi saklama düzenlerinin giriş ve çıkış kısıtlarını göz önüne almaktadır [17]. 1,5 Mbit/s hızında kodlanmış videonun bugünkü VHS türü videonun niteliğini sağlaması öngörülmüştür. MPEG standartları yayın amaçlı video telefon, video toplantı, Yüksek tanımlı televizyon (HDTV) gibi uygulamaları kapsamamakta, bu tür iki tarafın etkileşimini içeren alanlarda CCITT' e öneriler getirmektedir.

Bu çalışmada önerilen yöntem, yukarıda anlatılan uygulamalar için kullanılabilir. Ancak buradaki çalışmada laboratuvar olanakları da göz önüne alınarak sadece video telefon görüntüleri için testler yapılmıştır. Örnekleme, seviyelendirme ve kodlama gibi aşamalar H.261 video telefon standardına uygun olarak seçilmiştir.

Video sinyallerini kodlama teknikleri bu tezin birinci bölümünde özet olarak anlatılmakta ve hareket dengelemeli kodlamanın diğerlerine göre üstünlüğü ortaya konmaktadır. Çalışmanın özünü oluşturan ve video sinyallerinin kodlanmasında en önemli konulardan biri olan hareket kestirme konusu da yine bu bölümde ele alınmıştır.

İkinci Bölüm'de video sinyallerinde sıkıştırmada daha etkili olabilecek değişken blok büyüklüğü kullanan bir yöntem sunulmaktadır. Bu yöntem, video telefon görüntülerinin kodlanmasında sınıanmış ve elde edilen bulgular karşılaştırmalı olarak Üçüncü Bölüm'de verilmiştir.

1.2. Video Sinyallerini Kodlama Yöntemlerinin İncelenmesi

Video sinyali, bir cisim veya manzaranın kamerayla alınan görüntüsünün taranmasıyla elde edilir. Görüntü taramanın temel parametrelerinin çoğu insan gözünün algılama ve ayırtma özelliğine göre seçilir. Televizyondaki tarama işlemi, saniyede 25 veya 30 kez yapılır ve her taramada bir görüntü çerçevesi elde edilir. Her görüntü çerçevesi, cisimlerin bir anlık konumlarını belirten bir resim olarak düşünülebilir. Bir görüntüdeki düşey uzamsal boyuttaki ayrıntı satır sayısıyla, yatay uzamsal boyuttaki ayrıntı ise sistemin band genişliği ile ilgilidir. Örnekleme hızı, gerekli olan yatay ayrıntıyı sağlayacak kadar seçilir ve örnekleme kuramından hareketle yapılır. Böylece, en küçük görüntü biriminin boyutları örnekleme aralığına ve satır sayısına bağlı olur. Bu en küçük görüntü birimine görüntü elemanı (pixel) adı verilir [1]. Her örnek değeri veya örnek grubu sınırlı sayıda seviyeyle temsil edilir ve kodlanır. Kodlama ise, kestirim ve dönüşüm alma teknikleri de kullanılarak, değişik şekillerde yapılabilir.

Video sinyallerinde veri hızı örnekleme, seviyelendirme ve kodlama aşamalarının herbirinde azaltılabilir. Aşağıda örnekleme, seviyelendirme ve kodlama aşamaları anlatılmaktadır.

1.2.1. Örnekleme

Video sinyalinin sayısala çevrilmesindeki ilk aşama örnekleme aşamasıdır. Band genişliği ve hız bakımından ekonomik örnekleme yapabilmek için, görüntü önce belirlenebilir (deterministic) işlevlerle ifade edilir ve band sınırlandırılır. İki boyutlu bir işlevin Fourier dönüşümü frekans düzleminde sınırlı bir bölgenin dışında sıfır yapılırsa, band sınırlandırılmış olur [1]. Örneklenmiş bir görüntünün Fourier dönüşümü, ölçeklenmiş olarak örnekleme frekansları kadar aralıklı bir ızgara üzerinde giriş görüntüsünün Fourier dönüşümünün dönemi (periodic) olarak tekrarlanmış şeklindedir. Orijinal görüntünün örneklenmiş görüntüden tekrar elde edilebilmesi için örnekleme frekansı orijinal görüntünün band genişliğinin iki katından daha fazla seçilmelidir. Örnekleme hızının bu alt sınırı "Nyquist hızı" veya "Nyquist frekansı" olarak adlandırılır [1]. Bu frekansın örnekleme eksenindeki karşılığı (reciprocal) ise, Nyquist aralığı diye adlandırılır. Örnekleme kuramına göre, Nyquist hızının üzerinde örneklenmiş band sınırlı bir görüntü, bir alçak geçiren süzgeçten geçirilerek hatasız geri alınabilir.

Örnekleme frekansının yarısından daha yukardaki frekanslara "katlanma frekansları" (foldover frequency) denir [1], [18]. Tayfin ardışıl tekrarlarının üstüste binmesi orijinal görüntüde katlanma frekansları oluşturur. Buna "örtüşme" (aliasing) denir. Örtüşme süzgeçlemeyle giderilemez. Örtüşmeyi engellemek için bir yol, görüntüyü örneklemeden

önce bir alçak geçiren süzgeçten geçirmektir. Buradaki alçak geçiren süzgecin band genişliği, örnekleme frekansının yarısından daha az olmalıdır.

İnsanın görme sistemi fazla hareket içeren görüntü bölgelerindeki (yüksek zamansal frekanslar) yüksek uzamsal frekansları iyi algılayamaz. İnsan gözünün bu şekilde bir alçak geçiren süzgeçleme karakteristiğinin, yapılan bir takım ölçmeler sonunda baklava dilimi (diamond) şekline yakın olduğu ortaya çıkmıştır [1]. Bu nedenle iki boyutlu bir frekans düzlemindeki tayf baklava dilimi şeklinde sınırlanabilir. İnsan görme sistemi görüntünün yüksek hareket içeriği olan yerlerinde (yüksek zamansal frekansların olduğu bölgelerde) yüksek uzamsal frekanslara karşı duyarsızdır. Bu özelliklerden yararlanılarak "satır atlamalı" tarama ve "örgülü örnekleme" motifleri kullanılarak band genişliğinin düşük olması sağlanır.

Örgülü örnekleme görsel (subjective) görüntü kalitesi bakımından daha iyi sonuç vermektedir. Dairesel simetrik ve/veya dairesel bir bölgede band sınırlı işlevler için, örgülü örnekleme motifinin dikdörtgen örneklemeyle göre daha az örnek gerektirdiği veya aynı örnekleme hızında, dikdörtgen kafese göre daha az örtüşme meydana getirdiği kaydedilmiştir [1].

Dikdörtgen örneklemenin avantajı, işlemenin daha kolay olmasıdır. Örneğin, kestirme ve süzgeçleme işlemi dikdörtgen kafes üzerinde daha iyi uygulanır. Bu nedenle video sinyallerinin kaynak standardı genellikle dikdörtgen örneklemeyle belirtilir.

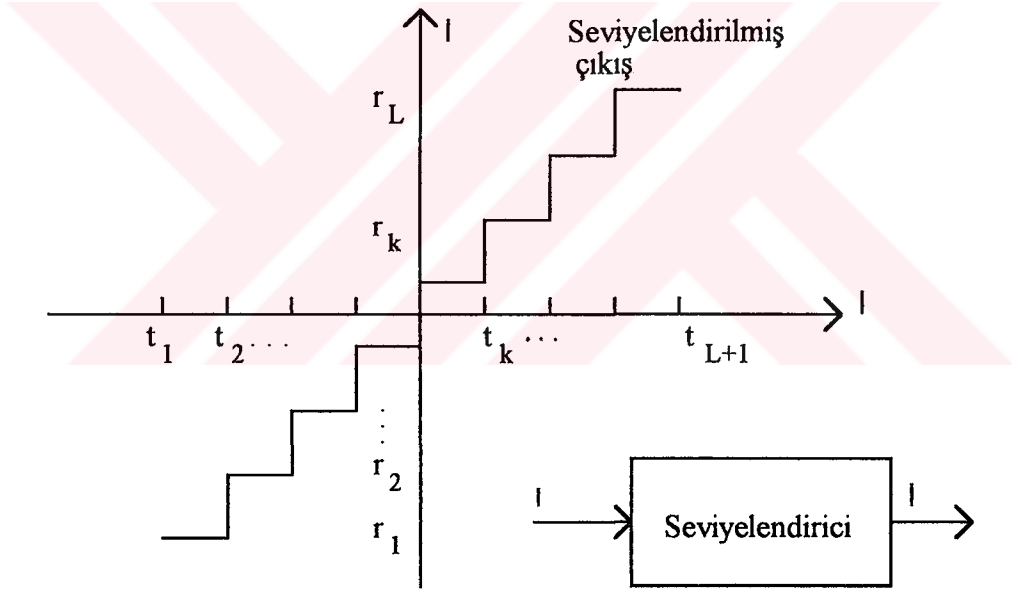
Renkli video sinyallerinin örneklenmesi de yine insan gözünün algılama özelliklerine göre yapılır [18]. İnsan gözünün renk ayrıtı duyarlılığı, parlaklık ayrıtı duyarlılığından daha azdır. Üç renk bileşeni buna göre düşük ayrıtılı olarak elde edilirse, parlaklık ayrıtısı yetersiz kalır. Üç renk bileşeninin band genişlikleri parlaklık ayrıtısını sağlayacak kadar arttırılırsa, bu sefer de band genişliği gereksiz yere artacaktır. Bunu önlemek ve siyah-beyaz televizyonla da uyum sağlamak için, renkli sinyallerin üç ana renk bileşeni olan kırmızı, yeşil ve mavi renk bileşenleri (K,Y,M), önce bir parlaklık ve iki renk fark bileşenine (P, P-K, P-M) çevrilir. P bileşeni daha yüksek band genişliğiyle, iki renk fark bileşeni ise daha düşük band genişliğiyle gösterilebilir. Renk fark bileşenleri örneğin NTSC sisteminde I ve Q, PAL sisteminde de U ve V koordinatlarında gösterilirler. Bu koordinatlardaki renk farkı algılanması, farklı renk bileşenleri için eşit değildir (non Euclidean). Renkli görüntülerin bu özellikleri dikkate alınarak uygulamada kullanılan görüntü kodlama sistemlerinde band genişliği daha az olan renk sinyalleri düşük hızlarda örneklenir. Tipik olarak I ve Q sinyalleri parlaklık sinyalinin üçte bir veya altıda bir hızında örneklenirler.

Renkli görüntülerin kodlanmasında bir başka yöntem, bileşik (composite) renk sinyalinin kodlanmasıdır [19], [20]. Kontrol edilecek olan sinyalin tek bir sinyal olması istendiğinde bu biçim yararlı olmaktadır. Fakat, parlaklık ve renk fark sinyallerinin aynı frekans bandında olmamaları nedeniyle, süregelen siyah-beyaz görüntü kodlama

teknikleri verimli bir şekilde doğrudan bileşik sinyallere uygulanamamaktadır. Tipik örnekleme frekansı olarak, taşıyıcı frekansının Nyquist hızının üzerindeki en küçük tamsayı çarpanlarından olan 3 fsc (fsc: renk taşıyıcı frekansı) veya 4 fsc kullanılır. Örnekleme sırasında görüntü elemanlarının göresel fazları da hesaba katılır. Örneğin, 3 fsc örneklemede taşıyıcının uygun örnekleri arasında 120° , 4 fsc örneklemede 90° faz farkı olmaktadır. Bileşik renkli (composite color) sinyallerin örneklenmesi, tayfin düzensiz şekli nedeniyle daha karmaşık ve zordur.

1.2.2 Seviyelendirme

Video sinyalini sayısala çevirmenin bir diğer aşaması seviyelendirmedir. Bir seviyelendirici ile sürekli bir I değişkeni, ayrık bir I değişkenine karşılık düşürülür. I , sonlu bir $\{r_1, r_2, \dots, r_L\}$ sayılar kümesinin değerlerini alabilir. Bu karşılık düşürme, genellikle bir merdiven basamağı işleviyle olur (Şekil 1).

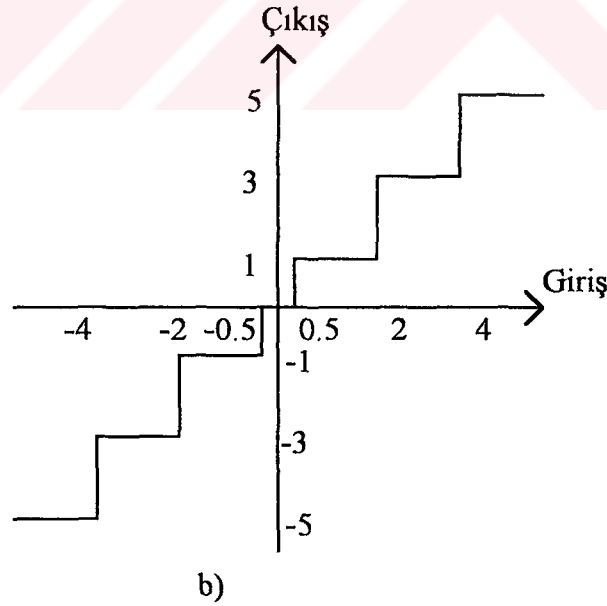
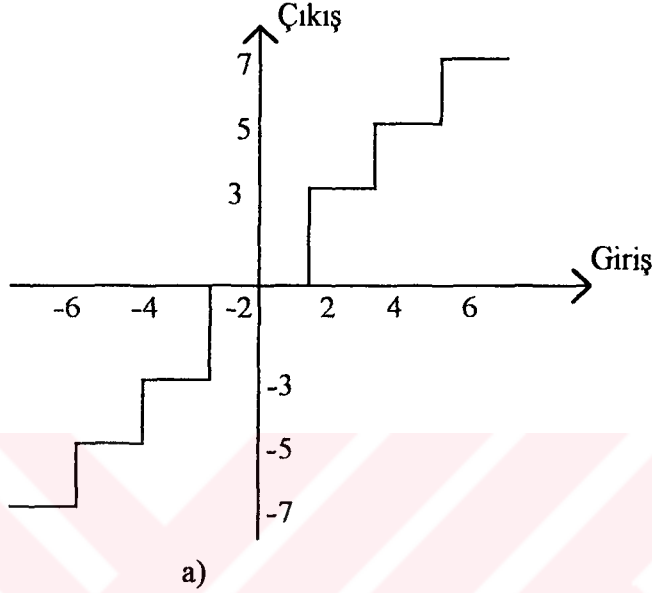


Şekil 1. Bir seviyelendiricinin giriş ve çıkış işlevi.

Seviyelendirme kuralı: $t_k, k=1, \dots, L+1$, I 'nin t_1 ve t_{L+1} minimum ve maksimum değerleri arasındaki sürekli artan geçişleri veya karar düzeyleri kümesi olsun. Eğer I $[t_k, t_{k+1})$ aralığındaysa, k 'inci düzey olan r_k 'ya karşılık düşürülür. Bir çok seviyelendirici tasarım şekli, performans ve basitlik arasında seçim yapılmasını gerektirir [1].

Seviyelendirme işlemi, doğrudan video sinyaline uygulanacağı gibi, bu sinyalin bir uniter dönüşümünü alıp, elde edilen dönüşüm katsayılarına da uygulanabilir. Örneğin, bir görüntü çerçevesinin ayrık kosinüs dönüşümü (AKD) katsayıları seviyelendirilebilir.

Dönüşüm katsayılarının alabilecekleri seviyelerin istatistiksel dağılımı ve insan gözünün özellikleri de düşünülerek bu seviyeler değişken adım büyüklüklü olabilir. Örneğin, eğer dönüşüm katsayıları bir görüntü çerçevesine aitse, düşük genlik değerleri daha ince, eğer iki görüntünün farkına aitse düşük genlikler daha kaba seviyelendirilebilir [17]. Şekil 2' de iki ayrı duruma örnek olarak seviyelendirme işlevi verilmiştir.



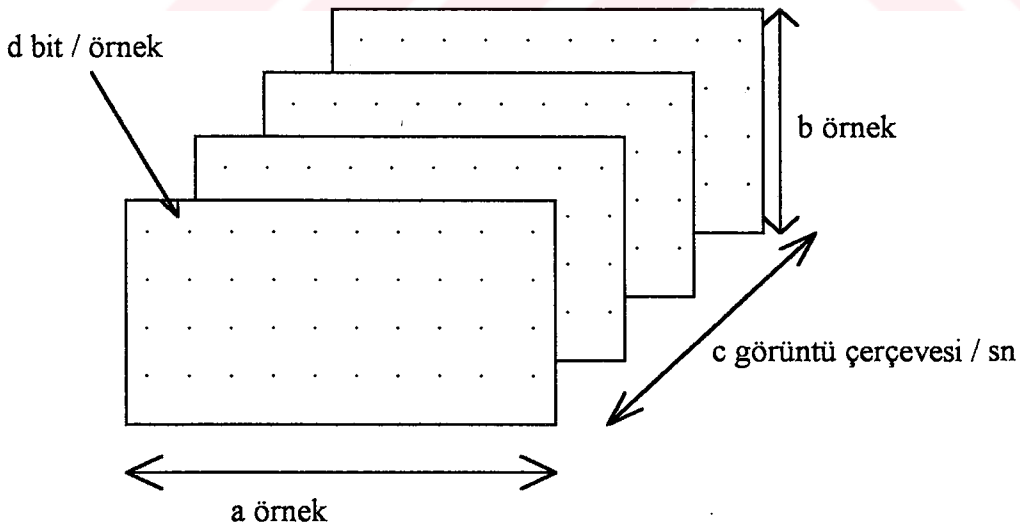
Şekil 2. Çerçevesel ve çerçeve içi sinyal için seviyelendirme karakteristikleri (adım büyüklüğü=2). a)Çerçevesel blok için. b)Çerçeve içi blok için.

Seviyelendirme hatasının gözle görülebilme özelliği, frekans içeriğine bağlıdır. Bu nedenle yüksek frekans bileşenlerini daha kaba seviyelendirmek uygundur.

Seviyelendirme matrisi, gösterme (display) karakteristiği, görüş uzaklığı ve kaynak gürültüsünün miktarına göre seçilir. Bu nedenle, uygulama alanına göre özel bir seviyelendirme matrisi seçmek mümkündür.

Bir çerçeve içi görüntüye ait dönüşüm katsayılarının enerjisi, bütün frekanslarda olabilir. Uygun bir adım büyüklüğüyle seviyelendirme yapılır. Fakat dönüşümü alınan katsayılar kestirim hatasına (örneğin iki görüntü çerçevesinin farkı) ait katsayılar, yüksek frekans bileşenleri daha fazla olur. Ayrıca kestirim işleminin düşük frekans bileşenleri için daha iyi olduğu düşünülürse, düşük frekans bileşenlerinin enerjisinin minimum olduğu varsayılarak çerçeve içi kodlama şekli uygulanabilir. Aksi takdirde iki tür için iki ayrı seviyelendirme şekli uygulanır. İki seviyelendirici de uniform ve sabit adım büyüklüğüne sahiptir, sıfır etrafındaki davranışları ise farklıdır. Şekil 2' de adım büyüklüğü 2 olan iki adet seviyelendirici görülmektedir. Çerçeve içi sinyalin dönüşümü sonunda ortaya çıkan katsayılar sıfır düzeyi için seviyelendirme bölgesi, adım büyüklüğünden daha küçüktür, Bu bölgeye "pasif bölge" (deadzone) denilmektedir. Dönüşüm katsayıları çerçeveler arası sinyale ait ise pasif bölge daha büyük seçilir.

Örneklenmiş ve seviyelendirilmiş video sinyalleri için gereken veri hızı, Şekil 3' den yararlanarak bulunabilir. Şekilden de görüldüğü gibi, bir görüntü çerçevesi yatay yönde a , dikey yönde ise b kadar örnekle gösterilebilir. Eğer saniyedeki görüntü sayısı c ile gösterilirse, ve her örneğin seviye sayısı d bit ile temsil edilirse saniyedeki veri hızı, $B=abcd$ bit/s olur.



Şekil 3. Görüntülerin örneklenmesi ve seviyelendirilmesi.

Uygulamada, veri hızı yukardaki B den biraz daha fazladır. Çünkü örnekleme işlemi genellikle, bazı yardımcı bilgileri yerleştirme amacıyla tarama geri dönme boşluklarında da devam edebilir. Yukardaki $B=abcd$ ifadesi, kaba bir bilgi vermesi için kullanılmıştır.

Renkli görüntüler, üç ayrı renk bileşeni içerdikleri için veri hızı 2B civarındadır. Üç renk bileşeni, band genişliğinden tasarruf amacıyla genellikle bir parlaklık ve iki renk fark sinyaline çevrilmektedirler. Veri hızının 3B olmamasının nedeni, renk fark sinyalinin parlaklık sinyalinden daha az örnekle ve her örneğin daha az seviyeyle gösterilebilmesinden kaynaklanmaktadır.

Yukardaki 4 parametre değeri, uygulama alanına göre farklılık göstermektedir. Bazı video sistemleri için bu parametreler aşağıdaki Tablo 1'de listelenmiştir. Aşağıdaki tablo, sadece parlaklık sinyali için kaba veri hızları hakkında bilgi vermesi için kullanılmıştır.

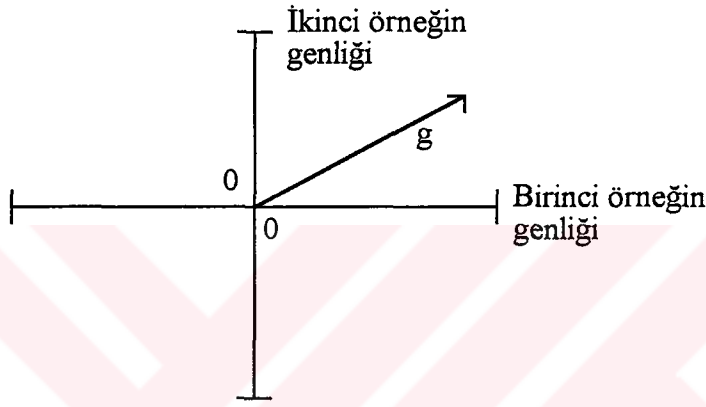
Tablo 1. Video sinyallerinde parlaklık bileşenine ait yaklaşık örnekleme ve seviyelendirme sayıları, veri iletim hızları ve hedeflenen hız değerleri.

Sistem	a	b	c	d	abcd (Mbit/sn)	Hedeflenen hız (Mbit/sn)
Video telefon (CIF)	352	288	30	8	24	0.008-0.064
(QCIF)	176	144	30	8	6	0.064
Video konferans (CIF)	352	288	30	8	24	0.384
Sayısal TV (CCIR 601)	720	576	25	8	83	5-10
HDTV	1920	1150	50	8	883	35-135
MPEG	352	288	30	8	24	0.9-1.5

CIF (Common Intermediate Format) için parametreler tabloda gösterilmiştir. QCIF (Quarter-CIF) için görüntü boyutları 176x144 görüntü elemanı (piksel) olarak seçilmiştir [2], [21]. Bu format video telefon gibi düşük detay kullanan sistemlerde kullanılabilir. CIF için renk bileşenleri de yine quarter-CIF olarak örneklenir.

Genelde örnekleme hızı örtüşme, seviyelendirme düzey sayısı seviye görünmesi (contouring) ve saniyedeki görüntü sayısı da kırpışma (flickering) sınırına kadar azaltılabilir. Bu sınırların altındaki seyrek örnekleme ve kaba seviyelendirmenin getirdiği bozulmalar, insan gözünü genellikle fazla rahatsız eder. Ancak, yine de detay ve hareket içeriğine göre uyarlamalı örnekleme, doğrusal olmayan seviyelendirme ve satır atlamalı tarama gibi yöntemlerle bu sınırlar zorlanabilir [1].

Buraya kadar anlatılan seviyelendirme çeşitleri skalar seviyelendirme adı altında toplanabilirler. Başka bir seviyelendirme şekli de "vektör seviyelendirme" dir (Vector Quantization). Skalar seviyelendirmede, her noktadaki kestirim hatasının genliği komşu görüntü elemanlarındaki kestirim hatalarından (veya komşu katsayıların genliklerinden) bağımsız olarak seviyelendirilir. Vektör seviyelendirmede ise, bir örnekler bloku toplu olarak seviyelendirilir. Buna basit bir örnek, iki parlaklık örneğine sahip bir bloktur. Eğer iki örnekli kombinasyon Şekil 4' de gösterildiği gibi tanımlanırsa (burada iki örneğin genliği dik eksenler boyunca çizilmiştir), iki örneğin oluşturduğu herhangi bir motif şekilde gösterilen anlamda bir vektörle gösterilebilir.

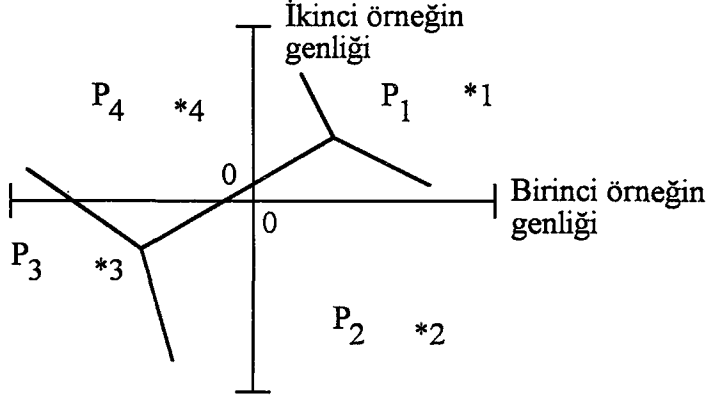


Şekil 4. İki örnekli bir bloğun iki boyutlu gösterimi.

Örnek olarak, 1bit/örnek hızında bir iletim yapılabilmesi için blok başına iki bit kullanılması gerekir. Böyle bir blok için dört farklı motif tanımlanabilir. Bu dört farklı motif, kodlanacak görüntüdeki bloklara giren vektörlerinin beklenen dağılımına uygun olarak seçilir. Bir görüntüyü 1bit/örnek ile kodlama işlemi, her bloku dört motifle karşılaştırmaktan ve minimum bozulmayı veren motifi seçmekten ibarettir.

Şekil 5' de blok seviyelendirme vektör diyagramında gösterilmiştir. P_i , $i=1,2,3,4$ bölmelerine düşen bütün giriş vektörleri sırasıyla 1,2,3 ve 4 numaralı durumlar olarak gösterilir. Kod çözücüde, kodlayıcıdan alınan kod sözcüğüne karşı düşen dört durum üretilir. Vektörler kümesine "vektör seviyelendirici", "blok seviyelendirici" veya "kod tablosu (codebook)" denir [22].

İki örnekli bir blok için iki boyutlu gösterim, k örnekli bir blok için bir k boyutlu vektöre genişletilebilir. Bir vektör, k örnekli blokta bir özel motife karşılık düşer. Şekil 4 takip edilerek, k -boyuttaki m .inci eksen, m .inci eksen hariç bütün k örnekleri sıfır olan motifler kümesini tanımlar. Bu eksenler birbirine ortogonaldır. Çünkü m .inci koordinat değeri (veya m . inci örnek genliği), diğer $k-1$ koordinatın (veya örnek genliğinin) değerini etkilemeden değiştirilebilir.



Şekil 5. Blok seviyelendirmeye iki boyutlu bir örnek (Örneğin, P1 bölgesine düşen bütün giriş blokları vektör-1 olarak seviyelendirilir).

K-boyutta vektörlerin koordinatlarını tanımlamak için başka eksenler de kolaylıkla seçilebilir. Örneğin, AKD gibi herhangi bir ortogonal dönüştürme, k adet ortogonal eksenin yeni bir yönlenmiş şeklini (örneğin dönmüş şeklini) gösterir.

Vektör seviyelendirme, uzun hesaplamalar gerektirmektedir. Bu hesaplamalardan birincisi, blok büyüklüğüne uyan bir vektörler kümesi seçilmesidir. Bunun için örnek başına ortalama bit sayısı ve görüntü kaynak vektörlerinin beklenen dağılımının hesaplanması gerekir. İkincisi, kodlayıcı referans vektörlerin hangisinin belirli bir bozulma ölçütüne göre özel giriş vektörüyle uyduğunu saptamalı. Sonuncusu, uygun kod tablosundan seçilen vektörün kod sözcüğü alıcıya gönderilir. Normal olarak vektörlerin kod tablosu önceden saptanır. Kod çözme konusu, daha basittir. Burada, alınan kodsözcüğüne karşılık gelen kod tablosundan ilgili motif elde edilir.

Kod tablosu üretimindeki hesapsal zorluklar nedeniyle blok büyüklükleri (örneğin 16 veya daha az örnek), seviyelendirmedeki hesapsal yük nedeniyle de kod tablolarındaki vektörlerin sayısı sınırlandırılır. Getirilen bu sınırlamalarla beraber, vektör seviyelendirmenin örneğin AKD' den biraz daha iyi sonuç verdiği kaydedilmiştir [22].

Vektör seviyelendirmenin en büyük avantajı, skalar seviyelendirmeye olan üstünlüğüdür. Bloktaki skalar büyüklükler arasındaki istatistiksel bağımlılıkları yok eder. Doğrusal dönüşümler yalnızca doğrusal bağımlılıkları azaltabilirken, vektör seviyelendirme doğrusal olmayan bağımlılıkları da azaltabilir. Vektör seviyelendirmenin performansını arttırmak birçok faktöre bağlıdır. Örneğin bu faktörlerden biri, vektördeki değerler arasındaki istatistiksel bağımlılıkların derecesidir. Performans arttırımı ile beraber hesap yoğunluğu ve bellek gereksinimi de artar. Bu ek maliyet, uygulamaya göre değişebilir. Vektör seviyelendirme, düşük veri hızları için uygundur. Çünkü burada performans arttırımı daha önemlidir ve düşük veri hızlarında vektör seviyelendirme için ek maliyet çok büyük değildir. Bu tez çalışmasında, örnekleme, seviyelendirme ve vektör

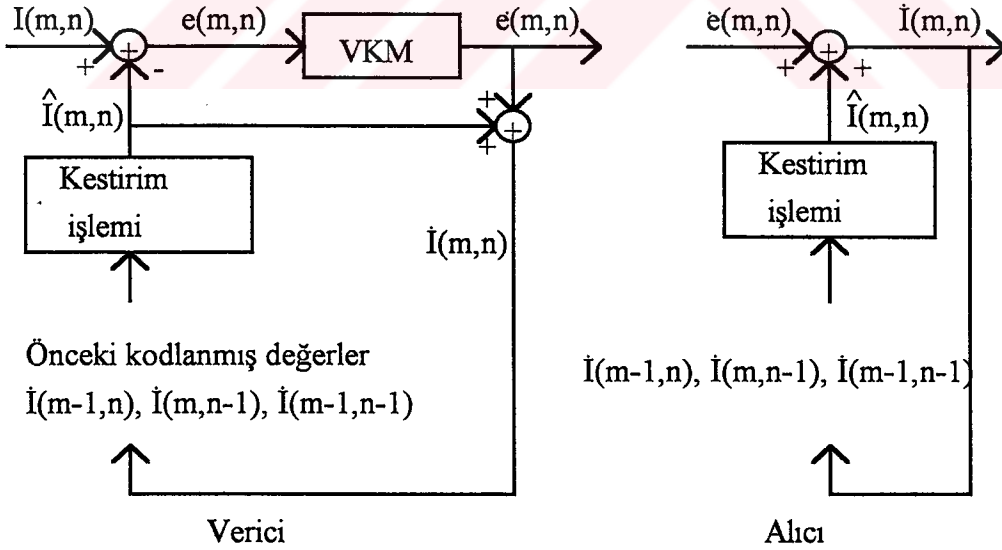
seviyelendirmeye dayalı veri azaltma teknikleri deneysel olarak incelenmemiştir. Bununla beraber, bu tekniklerin veri azaltma konusundaki önemleri vurgulanmaya çalışılmıştır.

1.2.3 Kodlama

1.2.3.1 Kestirim Kodlaması

Görüntü elemanları arasındaki ilişki derecesine ve veri istatistiğine bağlı olan artıklık, kestirim işlemleriyle yok edilebilir. Bu düşünceden hareket ederek, görüntü elemanları olduğu gibi kodlanmayıp, bu elemanlar için kestirilen değerlerle gerçek değerleri arasındaki fark kodlanır. Buna kestirim kodlaması adı verilir [1], [23]. Kestirim kodlamasına bir örnek olarak türevsel vuruş kod modülasyonu (TVKM) verilebilir. Aşağıdaki şekilde bir TVKM alıcı ve verici dizgesinin blok şeması gösterilmektedir.

Bir görüntü elemanının parlaklık değeri $I(m,n)$, daha önceki kodu çözülmüş (geri alınmış) görüntü elemanlarının parlaklık değerlerinden yararlanarak kestirilir. Kestirilmiş değer $\hat{I}(m,n)$ ile gösterilir. Şekil 6' da görüldüğü gibi $I(m,n)$ kodlanmadan önce, $\hat{I}(m-1,n)$, $\hat{I}(m,n-1)$, $\hat{I}(m-1,n-1), \dots$, değerleri geri besleme işlemiyle tekrar elde edilir. Buradaki amaç, $I(m,n)$ nin kestirilmesinde, daha önceki kodlanmış görüntü elemanları değerlerini kullanarak sistemdeki $e(m,n) = I(m,n) - \hat{I}(m,n)$ ' nin değişimini (varyansını) azaltmaktır.



Şekil 6. Bir TVKM verici ve alıcı dizgesi.

Kestirim hatası $e(m,n)$, bir doğrusal veya doğrusal olmayan seviyelendirici kullanan bir VKM sistemiyle seviyelendirilir. Seviyelendirilmiş $e(m,n)$, $e(m,n)$ ile gösterilir. Alıcıda, $e(m,n)$, $\hat{I}(m,n)$ ile birleştirilerek $I(m,n)$ ' nin kestirim değeri bulunur. Daha önce

kodu çözülmüş olan görüntü elemanı değerleri vericide ve alıcıda bilindiği için, alıcı ve vericide aynı $\hat{I}(m,n)$ elde edilir. Henüz kodlanmamış görüntü elemanı parlaklık değerlerinin kodlanmasında kullanılmak amacıyla $I(m,n)$ nin kod çözülmüş değeri $I(m,n)$ alıcıda da gerekir. Kodlamada kodu çözülmüş değerlerin de kullanılmasının seviyelendirme gürültüsünün yayılmasını engelleyici etkileri vardır. Şekil 6' da gösterilen TVKM sistemi için aşağıdaki denklemler yazılabilir [24]:

$$e(m,n) = I(m,n) - \hat{I}(m,n) \quad (1)$$

$$e(m,n) = Q[e(m,n)] \quad (2)$$

$$\hat{I}(m,n) = \hat{I}(m,n) + e(m,n) \quad (3)$$

Burada $Q[e(m,n)]$, $e(m,n)$ ' nin bir VKM sistemiyle seviyelendirildiğini gösterir. Yukardaki (1) ve (2) denklemlerinden yararlanılarak seviyelendirme gürültüsü $e_Q(m,n)$ şöyle gösterilebilir [24]:

$$e_Q(m,n) = \hat{I}(m,n) - I(m,n) = e(m,n) - e(m,n) \quad (4)$$

(1), (2) ve (3) ifadeleriyle temsil edilen bir TVKM sistemi, VKM'nin bir geliştirilmiş şekli olarak da düşünülebilir. Özel olarak, $\hat{I}(m,n)$ sıfır yapıldığında TVKM, bir VKM' ye indirgenmiş olur.

TVKM' de $I(m,n)$, daha önce kodu çözülmüş değerlerin doğrusal birleşiminden aşağıdaki gibi kestirilir [24]:

$$\hat{I}(m,n) = \sum_{(k_1,k_2) \in R_a} a(k_1,k_2) \hat{I}(m-k_1,n-k_2) \quad (5)$$

Burada R_a , $a(k_1,k_2)$ ' nin sıfırdan farklı olduğu bir (k_1,k_2) bölgesidir. Tipik olarak $\hat{I}(m,n)$ değeri, $\hat{I}(m-1,n)$, $\hat{I}(m,n-1)$ ve $\hat{I}(m-1,n-1)$ ' in doğrusal olarak birleştirilmesinden elde edilir. $I(m,n)$ ' nin kestirimi $e(m,n)$ ' nin değışintisini azaltmak amacıyla yapıldığı için, $a(k_1,k_2)$ ' yi aşağıdaki işlevi minimize ederek (minimization) kestirmek mümkündür [24]:

$$E[e^2(m,n)] = E \left[\left(I(m,n) - \sum_{(k_1,k_2) \in R_a} a(k_1,k_2) \hat{I}(m-k_1,n-k_2) \right)^2 \right] \quad (6)$$

$I(m,n)$, $a(k_1, k_2)$ ' nin bir işlevi ve kullanılan özel seviyelendiriciye bağlı olduğu için, (6) ifadesinin çözümü doğrusal olmayan (nonlinear) bir problemdir. Pratikte, $I(m,n)$ yerine $I(m,n)$ kullanılarak $a(k_1, k_2)$ katsayıları enküçükleme işlemiyle kestirilir [24]:

$$E \left[\left(I(m,n) - \sum_{(k_1, k_2) \in R_a} a(k_1, k_2) I(m - k_1, n - k_2) \right)^2 \right] \quad (7)$$

(7) ifadesindeki enküçüklenecek işlev, $a(k_1, k_2)$ ' nin bir karesel (quadratic) formu olduğu için, (7)' nin çözümü, aşağıdaki formda doğrusal denklemler kümesinin çözülmesiyle gerçekleştirilir [24]:

$$R_f(l_1, l_2) = \sum_{(k_1, k_2) \in R_a} a(k_1, k_2) R_f(l_1 - k_1, l_2 - k_2) \quad (8)$$

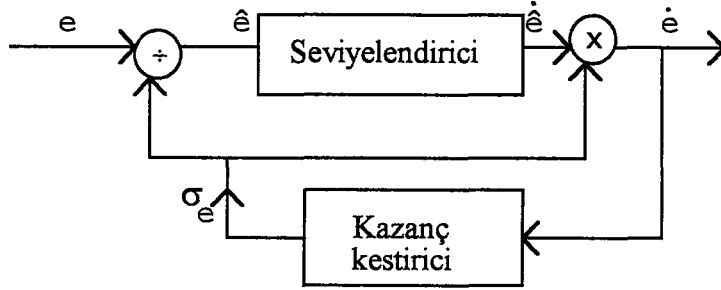
Burada $I(m,n)$ ' nin, ilişki işlevi $R_f(m,n)$ olan bir durağan rasgele işlem (stationary random process) olduğu varsayılmıştır.

Alıcı tarafta gerçek sinyal değerini elde edebilmek için, vericidekiyle zamandaş çalışan bir kestirici kullanılır. Kodu çözülmüş fark değeri, işaret bitine göre, bu kestirici çıkış değeri ile ya toplanır ya da çıkartılır. Video sinyalinin hızlı değişimler göstermesi durumunda, TVKM fark değeri doğrusal kestiricilerle sıfır seviyesine yaklaştırılmaz. Dizgenin fark değerlerine daha duyarlı olabilmesi için kullanılan yöntemlerden birisi, uyarlanır (adaptive) kestiriciler kullanmaktır. Seviyelendirici ve kestirici karakteristikleri görüntü verisinin yerel istatistiğindeki değişimlere bağlı olarak değiştirilir. Önceden kodlanmış görüntü eleman değerleri, uyarlanır kodlayıcının çalışma modunu belirlemekte kullanılır. Uyarlamalı kestiriciler genellikle görsel görüntü kalitesini (özellikle kenarlar veya köşeler) iyileştirmek amacıyla kullanılırlar. İyi bir uyarlama tekniği, birçok kestiriciyi birden kullanmaktır [1]. Görüntünün uzamsal ve zamansal ekseninde değişik ilişki düzeylerine sahip olmasından yararlanarak, maksimum ilişkinin yönü, önceki kodlanmış görüntü elemanı değerlerinden hesaplanır ve kullanılması gereken kestirici seçilir.

Sabit bir kestirici kullanıldığında, kestirim hatasının değışintisi görüntünün uzamsal detayındaki değışimlere bağlı olarak düzensizce değışir. Bunun için uyarlamalı seviyelendiriciler kullanmak yararlıdır. Genellikle iki tür uyarlamalı seviyelendirme yaklaşımı yapılabilir:

Uyarlamalı Kazanç Kontrolü: Basit bir uyarlamalı seviyelendirici, her adımda kestirim hatasının değışintisini değerlendirir ve buna bağlı olarak da seviyelendirmedeki

adım büyüklüğünü ayarlar. Şekil 7' de uyarlamalı kazanç kontrolü için ilkesel şema verilmiştir.



Şekil 7. Uyarlamalı kazanç kontrolü.

$\sigma_e^2(j)$ ve $\hat{\sigma}_e^2(j)$, bir TVKM sisteminin j . nci adımındaki seviyelendirici giriş ve çıkışlarının değişimini gösterebilir (iki boyutlu sistem için bunun anlamı, (m,n) 'yi j 'ye karşılık düşürmektir). $\hat{\sigma}_e^2(j)$ alıcıda olduğu gibi vericide de bulunduğu için $\hat{\sigma}_e^2(j)$ 'yi kestirmek kolaydır. Basit bir kestirim, üstel ortalama değışinti kestiricisidir (exponential average variance estimator) [1]:

$$\hat{\sigma}_e^2(j+1) = (1-\gamma)[\hat{\sigma}_e^2(j)]^2 + \gamma \sigma_e^2(j), \quad \hat{\sigma}_e^2(j) = (\hat{\sigma}_e(0))^2, \quad j=0,1,\dots \quad (9)$$

burada $0 \leq \gamma \leq 1$ dir. Küçük seviyelendirme hataları için $\hat{\sigma}_e(j)$, $\sigma_e(j)$ 'nin bir kestirim değeri olarak kullanılabilir. Pratik olarak, (9)' un kestirimi için aşağıdaki yaklaşım yapılabilir [1]:

$$\hat{\sigma}_e^2(j) \approx \gamma \sum_{m=1}^N |e(j-m)| \quad (10)$$

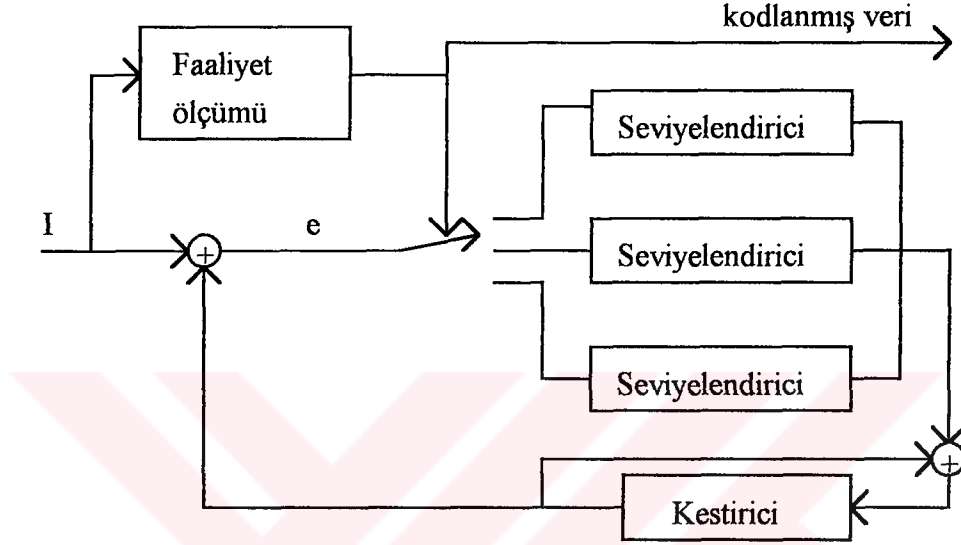
burada γ deneysel olarak ortalama kare hatanın enküçüklenmesiyle saptanan bir sabittir.

Düşük hızlarda bu kestirim yöntemi zayıf olduğu için, bunun yerine, bir $\sigma_e = g(m, n)$ kazancı tanımlanıp özyineli olarak düzeltilir [1].

$$g(m, n) = \sum_{(k,l) \in R_a} \alpha_{k,l} g(m-k, n-l) M(|q_{m-k, n-l}|), \quad g_{\min} \leq g(m, n) \leq g_{\max} \quad (11)$$

Burada $M(|q_i|)$, q_i seviyesi için çarpma faktörü ve $\alpha_{k,l}$: toplamı 1'e eşit olan ağırlık katsayılarıdır. Genellikle $\alpha_{k,l}=1/N_{R_a}$ alınır. Burada N_{R_a} , R_a nedensel bölgesi için görüntü elemanları sayısıdır.

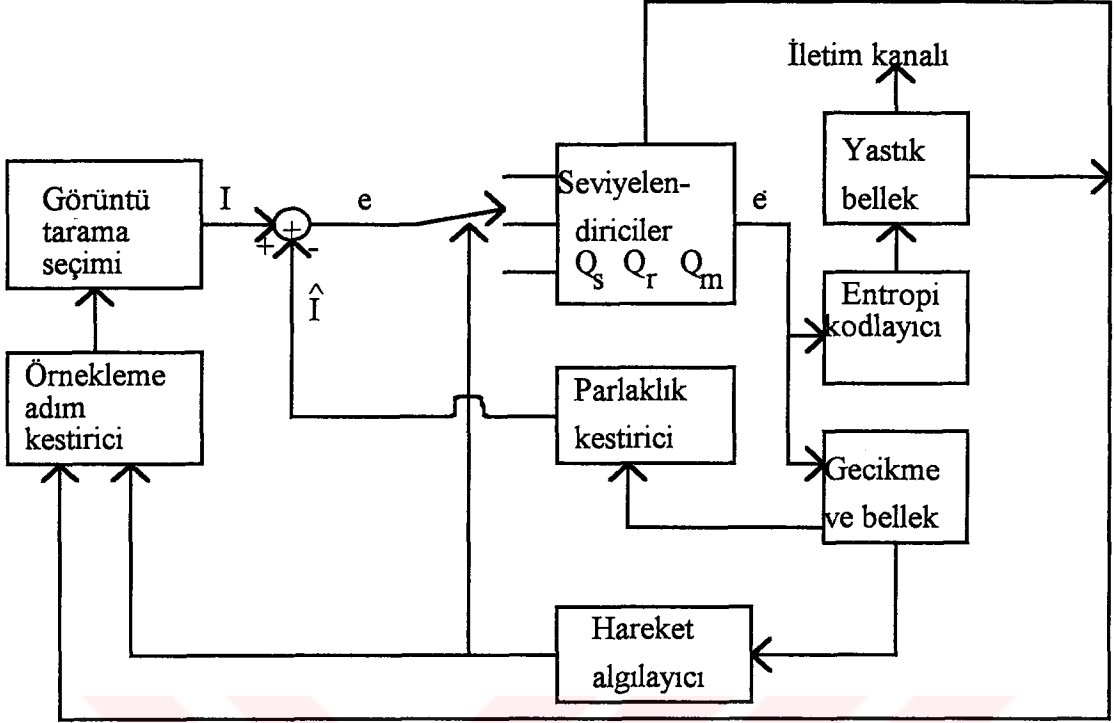
Uyarlamalı Sınıflandırma: Görüntü, uzamsal ayrıntı veya faaliyet içeriğine göre farklı bölgelere ayrılır ve her bölge için farklı seviyelendirici karakteristikleri kullanılır (Şekil 8).



Şekil 8. Uyarlamalı Sınıflandırma.

Basit bir faaliyet ölçütü, kestirilecek görüntü elemanlarına komşu olan görüntü elemanlarının değişintisidir. Düz bölgeler kenar veya detaylı bölgelerden daha ince olarak seviyelendirilir. Bu yöntem, faaliyet miktarı arttıkça gürültü görünürlüğünün azalmasının üstünlüğünü kullanır.

Kestirim işlemi, ardışıl görüntü çerçeveleri arasındaki artıklığın (temporal redundance) azaltılmasını da içeriyorsa buna "çerçeveler arası kodlama" denilmektedir. Burada kestirim işlemine daha önceki görüntü çerçevelerindeki komşu görüntü elemanları da dahil edilir. Ancak, hareket içeren görüntülerde kestirim işlemi hareketin olduğu alanlarda iyi yapılamadığı için uyarlamalı kestirciler burada da kullanılabilir. Hareket içeriğine göre uyarlama, çerçeveler arası kodlama metodunda performansta önemli kazançlar sağlar. Video sinyali, eğer hareket içermeyen bir görüntü alanına aitse (c_s), yavaş ve düzgün değişen bir alana aitse (c_m), hızlı hareket içeren bir alana aitse (c_r) gibi kısımlara ayrılır (Şekil 9).



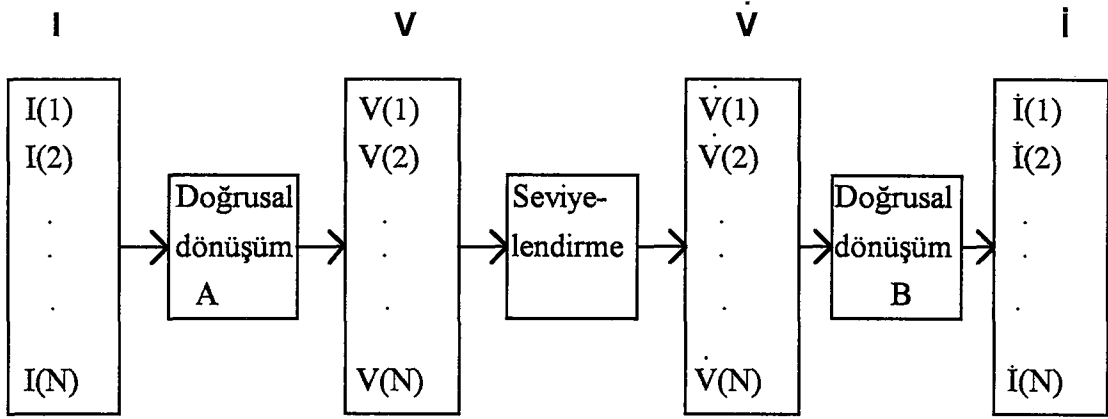
Şekil 9. Uyarlanı TVKM dizgesi.

Bu sınıflandırma, daha önce kodlanmış bazı görüntü elemanı değerlerinin çerçeveler arası farklarının mutlak toplamına dayanarak yapılır. Hızlı hareketler içeren alanlar için uzamsal ekseninde iki boyutlu kestirici kullanmak daha iyi sonuç vermektedir. Her kestirici sınıfında, fark sinyali için kullanılan seviye sayısı o sınıfın aktivitesiyle orantılı olarak seçilir. Bu metod, çerçeve yenileme (conditional replenishment) tekniğine göre aynı sinyal/gürültü oranında veri hızını iki kat azaltabilmektedir [1].

1.2.3.2. Dönüşüm Kodlaması

Görüntü elemanlarının parlaklık veya renk bilgilerini taşıyan bir görüntü matrisi, toplam enerjisinin büyük bir kısmı nispeten az sayıda dönüşüm katsayısıyla temsil edilecek şekilde başka bir koordinat ekseninde gösterilir. Yeni koordinat eksenindeki eleman değerlerinin birbiriyle ilişkisiz olması dönüşüm almanın en önemli amacıdır [25], [26]. Dönüştürme işleminin fazla miktarda hesaplama gerektirmesi nedeniyle görüntü verileri dönüşüm almadan önce genellikle alt öbeklere bölünür. Bu nedenle dönüşüm kodlamasına blok seviyelendirme adı da verilir [1]. Örneğin, $M \times N$ boyutlu görüntü datası $p \times q$ boyutlarında alt görüntü data öbeklerine ayrılır. Her bir öbek içerisindeki görüntü noktalarına ait dönüşüm katsayıları belirlenir. İkinci aşamada, her bir

öbek içerisindeki dönüşüm katsayıları ayrı ayrı basamaklandırılır ve kodlanır. Bu alt öbeklere temel görüntüler (basis images) denir ve üniter matrislerle temsil edilirler. Şekil 10' da bir boyutlu dönüşüm kodlayıcının ilke şeması verilmektedir.



Şekil 10. Bir boyutlu dönüşüm kodlayıcı.

Şekil 10'da vektör bloklarının üst kısmında koyu renk ile yazılan semboller matris anlamındadır.

Aşağıda, görüntü işleme konusunda yararlı olan KL dönüşümü, Ayrık Fourier Dönüşümü (AFD), ve Ayrık Kosinüs Dönüşümü (AKD) özet olarak açıklanmaktadır.

Karhunen-Loeve Dönüşümü (KLD): KL dönüşümü, ilk olarak Karhunen ve Loeve tarafından, sürekli bir rastlantı olayına (random process) ilişkin seri açılımı olarak tanıtılmıştır. Rastlantı dizileri için KL seri açılımının bir ayrık eşdeğeri olan "temel bileşenler" konusunda ilk çalışmaları Hotelling yapmıştır. Bu nedenle KL dönüşümü, "Hotelling dönüşümü" veya "temel bileşenler metodu" olarak da adlandırılır [1].

Gerçek bir $N \times 1$ boyutlu I rastlantı vektörü için KL dönüşümünün temel vektörleri (basis vectors), I vektörünün özilişki matrisinin (R) ortonormalize edilmiş özvektörleriyle (orthonormalized eigenvectors) verilir [1].

$$R\phi_k = \lambda_k \phi_k \quad 0 \leq k \leq N-1 \quad (12)$$

ϕ , R 'yi köşegen matrise (\cap matrisi) dönüştüren $N \times N$ boyutlu üniter bir matris olsun. (*) ifadesi karmaşık eşlenik, (T) ise transpoze anlamındadır. I vektörünün KL dönüşümü ve ters KL dönüşümü aşağıdaki gibi tanımlanır [1]:

$$V = \phi^{*T} I \quad (13)$$

$$I = \phi V = \sum_{k=0}^{N-1} V(k) \phi_k$$

Burada ϕ_k , ϕ matrisinin k'inci kolonudur. Koyu yazılmış karakterler ise matris anlamında kullanılmıştır. R matrisi, (14) ifadesinde ϕ matrisi yardımıyla köşegen matrise dönüştürülmektedir:

$$E[VV^*] = \phi^{*T} \{E[II^{*T}]\} \phi = \phi^{*T} R \phi = \Lambda = \text{Diag}\{\lambda_k\} \quad (14)$$

Özilişki matrisi yerine ortak değişinti matrisi de (covariance matrix) kullanılabilir:

$$R_\theta \hat{=} \text{cov}[I] \hat{=} E[(I - \mu)(I - \mu)^T] = E[II^T] - \mu\mu^T = R - \mu\mu^T, \quad \mu \hat{=} E[I] \quad (15)$$

Eğer μ vektörü biliniyorsa R_θ 'ın köşegen matrisi, $I - \mu$ sıfır ortalamalı rastlantı olayının KL dönüşümünü belirler.

İki boyutlu görüntü verileri için KL dönüşümü aşağıdaki gibi hesaplanabilir:

$N \times N$ boyutlu bir $I(m,n)$ görüntüsü, özilişki işlevi (16) ifadesiyle verilen bir rastlantı alanıyla (random field) gösterilsin [1]:

$$E[I(m,n)I(m',n')] = r(m,n; m',n'), \quad 0 \leq m, m'; n, n' \leq N-1 \quad (16)$$

KL dönüşümünün temel görüntüleri ortonormalize edilmiş $\xi_{k,l}(m,n)$ köşegen vektörlerdir. Bu vektörler (17) ifadesi yardımıyla bulunur [1].

$$\sum_{m'=0}^{N-1} \sum_{n'=0}^{N-1} r(m,n; m',n') \xi_{k,l}(m',n') = \lambda_{k,l} \xi_{k,l}(m,n), \quad 0 \leq k,l, m, n \leq N-1 \quad (17)$$

Matris formda (17) ifadesi aşağıdaki gibi yazılabilir:

$$R \xi_i = \lambda_i \xi_i, \quad i=0, \dots, N^2-1,$$

$$R = E[II^T] \quad (18)$$

Burada ξ_i , $\xi_{k,1}(m,n)$ ' nin $N^2 \times 1$ boyutlu vektörel bir gösterimi, R ise $N^2 \times 1$ boyutlu i vektörüne karşılık gelen $N^2 \times N^2$ boyutlu görüntü özilişki matrisidir. Eğer R ayrılabilir bir matris ise, $N^2 \times N^2$ boyutlu ξ matrisinin kolonları da (ξ_i) ayrılabilir. Örneğin, $r(m,n; m',n') = r_1(m,m')r_2(n,n')$ ve $\xi_{k,1}(m,n) = \phi_1(m,k)\phi_2(n,1)$ olsun. Bu ifadeler matrisel olarak şöyle gösterilebilir:

$$R = R_1 \otimes R_2 \quad \xi = \phi_1 \otimes \phi_2 \quad (19)$$

$$\phi_j R_j \phi_j^{*T} = I_j, \quad j = 1,2 \quad (20)$$

Burada (\otimes) ifadesi kronoker çarpmayı (kronocker product) ifade eder [1]. i ' nin KL dönüşümü aşağıdaki gibi hesaplanır:

$$v = \xi^{*T} i = (\phi_1^T \otimes \phi_2^{*T}) i \quad (21)$$

veya, satır sırasına göre tek boyuta dönüştürülmüş (row ordered) vektörler için Kl ve ters KL dönüşümleri aşağıdaki gibi hesaplanır:

$$V = \phi_1^{*T} I \phi_2^* \quad (22)$$

$$I = \phi_1 V \phi_2^T \quad (23)$$

Görüntü özilişkisini ayrılabilir işlevlerle modellemenin avantajı, $N^2 \times N^2$ kadar matris özdeğer problemi çözme yerine yalnızca $2 \times N \times N$ matris özvektör probleminin çözülmesi gerekir.

KL dönüşüm katsayılarının değışintisi katsayıların derecelerinin artmasıyla monoton azalan değerde olduğu ve diğer dönüşümlere göre en iyi enerji uyumu (energy compaction) özelliđi gösterdiği için optimum dönüşüm olarak bilinir. Fakat pratik veri azaltma metodlarında kullanılması zordur. Bunun nedenlerinden biri, dört ayrı kademede yapılabilen hesaplamalar çok fazladır. Bu kademeler, orijinal ilişki matrisi, köşegenleştirme, KL matrisinin elemanları ve dönüşüm katsayılarının hesabıdır. İkincisi, görüntü istatistiđi nadiren kararlıdır (stationary). Her yeni görüntü bloku için alıcıya yeni bir ters dönüşüm matrisi gönderilmesi gerekir. Bu ise pratik değildir. Bununla beraber KL dönüşümü, diğer dönüşüm yöntemlerinin göresel performans ölçümü için kullanılabilir.

Hızlı KL dönüşümü için bazı teknikler geliştirilmiş ve günümüzde de bu çalışmalar devam etmektedir [27].

Ayrık Fourier Dönüşümü (AFD): Bir $\{I(n), n=0,1,\dots,N-1\}$ dizisinin ayrık Fourier dönüşümü aşağıdaki gibi tanımlanır [1]:

$$v(k) = \sum_{n=0}^{N-1} I(n) \left(W_N^{kn} \right), \quad k = 0, 1, \dots, N-1 \quad (24)$$

burada $W_N \triangleq e^{-j2\pi/N}$ dir. Ters dönüşüm ise aşağıdaki gibi verilebilir:

$$I(n) = \frac{1}{N} \sum_{k=0}^{N-1} v(k) \left(W_N^{-kn} \right), \quad n = 0, 1, \dots, N-1 \quad (25)$$

Yukardaki (24) ifadesi $1/\sqrt{N}$ ve (25) ifadesi de \sqrt{N} ile çarpılmak suretiyle ölçeklenirse, bu iki dönüşüm uniter dönüşüm şekline girer. Görüntü işlemede uniter dönüşüm kullanmak daha uygun olmaktadır. $M \times N$ boyutlu bir görüntüye ilişkin iki boyutlu AFD, ayrılabilir bir dönüşümdür (separable transform). Aşağıdaki (26) ve (27) ifadeleri iki boyutlu uniter AFD çifti için kullanılmıştır:

$$v(k,l) = \frac{1}{N} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} I(m,n) W_N^{km} W_N^{ln}, \quad 0 \leq k \leq M-1, \quad 0 \leq l \leq N-1 \quad (26)$$

$$I(m,n) = \frac{1}{N} \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} v(k,l) W_N^{-km} W_N^{-ln}, \quad 0 \leq m \leq M-1, \quad 0 \leq n \leq N-1 \quad (27)$$

N boyutlu bir dizinin Ayrık Fourier dönüşümü için N^2 karmaşık çarpma ve toplama gerekmektedir. İşlem sayısını azaltmak için geliştirilmiş hızlı Fourier dönüşümü algoritmaları kullanılabilir [28]. N boyutlu bir dizinin hızlı AFD ile işlenmesi $N \log_2 N$ karmaşık toplama ve çıkarma gerektirir.

Ayrık Kosinüs Dönüşümü: Bir $\{I(n), n=0,1,\dots,N-1\}$ dizisinin ayrık Kosinüs dönüşümü aşağıdaki gibi tanımlanır [1]:

$$v(k) = \alpha(k) \sum_{n=0}^{N-1} I(n) \cos \frac{\pi(2n+1)k}{2N}, \quad 0 \leq k \leq N-1 \quad (28)$$

$$\alpha(0) = \sqrt{\frac{1}{N}}, \quad \alpha(k) = \sqrt{\frac{2}{N}}, \quad 1 \leq k \leq N-1$$

Ters AKD ise aşağıdaki ifadeyle hesaplanır:

$$I(n) = \sum_{k=0}^{N-1} \alpha(k) v(k) \cos \frac{\pi(2n+1)k}{2N}, \quad 0 \leq n \leq N-1 \quad (29)$$

$N \times N$ boyutlu bir görüntüye ilişkin AKD, aşağıdaki (30) ifadesiyle tanımlanır:

$$v(k, l) = \alpha(k) \alpha(l) \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} I(m, n) \cos \frac{\pi(2m+1)k}{2M} \cos \frac{\pi(2n+1)l}{2N},$$

$$0 \leq k \leq M-1, \quad 0 \leq l \leq N-1 \text{ ve}$$

$$\alpha(0) = \sqrt{\frac{1}{N}}, \quad \alpha(k) = \alpha(l) = \sqrt{\frac{2}{N}}, \quad 1 \leq k, l \leq N-1 \quad (30)$$

Ters AKD ise aşağıdaki ifadeyle hesaplanır:

$$I(m, n) = \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} \alpha(k) \alpha(l) v(k, l) \cos \frac{\pi(2m+1)k}{2M} \cos \frac{\pi(2n+1)l}{2N},$$

$$0 \leq k \leq M-1, \quad 0 \leq l \leq N-1 \quad (31)$$

$M \times N$ boyutlu bir görüntüye ilişkin AKD ayrılabilir bir işlem olduğundan, (30) ve (31) ifadeleri tek boyutlu AKD ile hesaplanabilirler. Ayrıca hızlı AKD algoritmaları, hızlı AFD algoritmalarından yararlanarak kolayca geliştirilebilir [29], [30].

Dönüşüm alma işlemi sonucunda ortaya çıkan katsayılar incelendiğinde, örneğin Fourier dönüşümünde, enerjinin sıfır frekanslı bileşen etrafında toplandığı görülür. Bu durum, küçük değerlikli yüksek frekans bileşenlerinin alçak frekanslı bileşenler yanında gözardı edilebileceğini gösterir. Kosinüs dönüşümünde ise enerjinin büyük bir bölümü dönüşüm katsayı matrisinin sol üst köşesinde toplandığı görülür.

Görüntülerin dönüşüm kodlanmasında, görüntüleri küçük bloklara bölerek işlem yapmak hesapları birkaç kat azaltırken donanım karmaşıklığını önemli ölçüde azaltmaktadır. $\log_2 MN / \log_2 pq$ ifadesi, küçük bloklara bölünmüş bir görüntü çerçevesi için hesaplama miktarındaki azalmayı verir. Burada M ve N görüntünün boyutlarını, p ve q ise blok boyutlarını göstermektedir. Görüntüyü küçük bloklara bölmenin hesaplama miktarını ne kadar azalttığı örnek olarak KL dönüşüm yöntemi için incelenebilir: 512×512 boyutlu bir görüntü 16×16 lık bloklara bölünürse, bir görüntü çerçevesine düşen blok sayısı 1024 olarak bulunur. Yukardaki formül kullanılarak hesaplamadaki azalma miktarı, $\log_2(512 \times 512) / \log_2(16 \times 16) = 18/8 = 2.25$ olarak bulunur.

Kodlama verimini artırmak için, uyarlanır dönüşüm kodlama algoritmaları da geliştirilmiştir. Aşağıda üç çeşit uyarlama tipi anlatılmaktadır:

1-Dönüşümün uyarlanması: İstatistiksel parametrelerde herhangi bir değişiklik olduğu zaman, dönüşümün temel vektörleri de değiştirilir. Bu yöntem karmaşıktır ve ekonomik değildir [1].

2-Veri tahsisinin uyarlanması (bit allocation adaptation): Görüntü blokları, uzamsal aktivitelerine (örneğin verilerin değışintisi) göre sınıflandırılır ve her bloka tahsis edilecek bit sayısı belirlenir. Burada bloktan bloka değışik veri hızları ortaya çıkar, fakat verilerin tüm görüntü bloku üzerinden daha verimli kullanılması sağlanır.

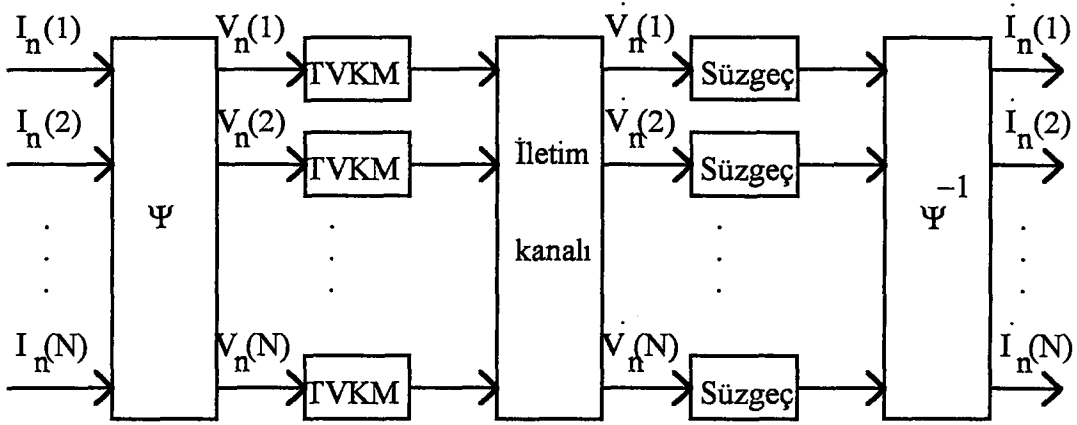
3-Seviyelendirmede seviye sayısının uyarlanması: Her blok aynı bozulmaya sahip olacak şekilde her bloka bit tahsisi yapılır. Bu işlem, görüntüde uniform bir bozulma meydana getirir ve göz tarafından daha az algılanır.

Uyarlanır seviyelendirme dizgelerinde bit tahsisi sabit tutulur fakat dönüşüm katsayılarının değışintisine bağı olarak seviye sayısı değıştirilir. Dönüşüm katsayılarının değışintisi, ya ortak değışinti (covariance) modelinin istatistiksel parametrelerinin uyarlanmasıyla ya da dönüşüm katsayılarının yerel kare genliklerinin ortalamasını alarak kestirilebilir [1].

Dönüşüm temel vektörlerinin uyarlanır seçilmesi, maliyeti arttırmaktadır. Çünkü istatistiksel parametrelerde oluşan herhangi bir değışimde yeni bir KL temel vektör kümesi gerekir. Daha pratik bir metod, bloktaki uzaysal aktiviteye (örneğin verideki değışim) uygun olarak, görüntü blokuna uyarlanır veri tahsisi yapmaktır. Görüntü blokları ise, bazı özellikleri önceden saptanarak bir çok gruba ayrılır. Bu, bloktan bloka ortalama hızı değışken bir ortalama hız ortaya çıkarır. Fakat, görüntü bloklarının bütünü üzerinden toplam verinin daha iyi bir kullanımını sağlar. Diğeri, her blok aynı bozulmaya sahip olacak şekilde görüntü bloklarına veri tahsisi yapmaktır. Bunun sonucunda görüntü kalitesinde oluşan azalma göz tarafından çok az algılanır.

1.2.3.3. Melez Kodlama

Bu kodlama yöntemi, hem dönüşüm kodlamalarını, hem de kestirim kodlamalarını içerir. Görüntü verisi uniter dönüşümlerle birbirinden bağımsız tek boyutlu veri dizilerine dönüştürülür. Bu veri dizileri daha sonra kestirim kodlama yöntemlerinden biri ile, örneğin TVKM ile işlenir (Şekil 11).



Şekil 11. Melez Kodlama.

Melez kodlama, N bağımsız (decoupled) skaler TVKM kodlayıcı yardımıyla bir $N \times 1$ vektör TVKM elde etme olarak da algılanabilir. Melez kodlama, dönüşüm ve kestirim kodlama tekniklerinin birleştirilmesiyle yapılır. Tipik olarak, görüntünün ele alınan herhangi bir yönünde (örneğin düşey) görüntü elemanlarının ilişkisini yok etmek için bir boyutunda üniter dönüşümü alınır. Her dönüşüm katsayısı diğer bir yönde (örneğin yatay) bir boyutlu TVKM ile ardışıl olarak kodlanır [1], [12].

Melez kodlama tekniği, TVKM'nin donanım basitliğini ve dönüşüm kodlamasının güçlü performansının avantajını birleştirir. Bu metodun donanım karmaşıklığı, bir boyutlu dönüşüm kodlayıcı ve en fazla N adet TVKM kanalı kadardır. Pratikte TVKM kanallarının sayısı N den çok azdır. Çünkü dönüşüm vektörünün çoğu elemanları sıfırdır ve bu yüzden kodlanmalarına gerek yoktur.

I_n , $n=0,1,\dots$, bir görüntüdeki $N \times 1$ uzunluklu kolonları gösterebilir ve aşağıdaki gibi dönüştürülerek V_n kolonları elde edilsin [1]:

$$V_n = \Psi I_n, \quad n = 0, 1, \dots \quad (32)$$

Her k için, $V_n(k)$ dizisi genellikle bir birinci dereceden otomatik gerileyen (autoregressive) işlem olarak şöyle modellenir [1]:

$$V_n(k) = a(k)V_{n-1}(k) + b(k)e_n(k), \quad 1 \leq k \leq N, \quad n \geq 0$$

$$E[e_n(k)e_{n'}(k')] = \sigma_e^2(k)\delta(k-k')\delta(n-n') \quad (33)$$

Burada (\cdot) sembolü karmaşık eşleniğin transpozitesi anlamındadır. Yukarıdaki modelin parametreleri, her k için $V_n(k)$, $n=0,1,\dots$ ifadesinin ortak değişiminden belirlenir. k 'nci kanal için TVKM eşitliği şöyle yazılabilir [1]:

$$\text{Kestirici: } \hat{V}_n(k) = a(k)\hat{V}_{n-1}(k) \quad (34)$$

$$\text{Seviyelendirici girişi: } \hat{e}_n(k) \hat{=} (V_n(k) - \hat{V}_n(k)) / b(k) \quad (35)$$

$$\text{Seviyelendirici çıkışı: } \dot{e}_n(k)$$

$$\text{Süzgeç: } \dot{V}_n(k) = \hat{V}_n(k) + b(k)\dot{e}_n(k) \quad (36)$$

Alıcı, (36)'ya göre dönüştürülmüş vektörleri oluşturur ve ters dönüşümünü alır [1]. İdeal olarak dönüşüm, l_n 'nin KL dönüşümü olmalıdır. Pratikte ise, hızlı bir sinüsoidal dönüşüm (kosinüs veya sinüs) kullanılabilir.

Her TVKM kanalı için kullanılacak seviyelendiricinin seçimi tasarımda önemli bir aşamadır. n_k , k'ncü TVKM kanalına tahsis edilen bit sayısı ve $\sigma_e^2(k)$, k'ncü kanaldaki seviyelendirici ortalama kare hatasını gösterebilir. B, ortalama veri hızı (bit/pixel) (37) ifadesi ile hesaplanır [1]:

$$B = \frac{1}{N} \sum_{k=1}^N n_k, \quad n_k \geq 0 \quad (37)$$

Bütün TVKM kanallarının kararlı durumda olduğu varsayılırsa, herhangi bir vektörün kodlanmasındaki ortalama kare hata (gürültüsüz kanal için), TVKM kanallarının ortalama bozulmasına eşittir. Yani,

$$D = \frac{1}{N} \sum_{k=1}^N g_k(n_k) \sigma_e^2(k), \quad g_k(x) \hat{=} \frac{f(x)}{1 - |a(k)|^2 f(x)} \quad (38)$$

Burada $f(x)$ ve $g(x)$ birim değışintili kestirim hatası için sırasıyla, seviyelendirici ve k'ncü TVKM kanalına ait bozulma hız işlevleridir (distortion-rate functions). Melez kodlama için veri tahsis problemi, (38) ifadesini (37) şartlarında enküçüklemek demektir.

Uniter dönüşümleri alınmış sütunlara ilişkin görüntü verileri TVKM işlemine sokulup, ya aynı iletim kanalında çoğullanarak yada farklı farklı kanallarda iletilirler. Alıcı tarafta bu veriler süzgeçlendikten sonra ters uniter işlemlerine sokularak görüntü sütun verileri yeniden elde edilir.

Uyarlanır Melez Kodlama: Bu kodlama algoritmasında, görüntü istatistiğinde değışimlere otomatik gerileyen (AR) model parametreleri uyarlanır. Basit bir metod, her TVKM kanalına uyarlanır değışinti kestirme algoritması uygulamaktır (uyarlamalı kazanç kontrolü). Uyarlanır sınıflandırma metodu ise, özellikle düşük hızlarda daha iyi sonuç verir. Her görüntü kolonu, W adet (genellikle dört) önceden saptanmış sınıfa

ayrılır. Bu sınıflar kolonların değışinti dađılımları eşit olacak şekilde sınıflandırılır. Yüksek dinamik aktiviteli kolonlar için düşük aktivite kolonlarından daha fazla olacak şekilde, farklı sınıflar arasında veriler paylaşılır.

Sınıf üyelik bilgisi, kolon başına ek olarak $\log_2 W$ adet bit veya $(1/N)\log_2 W$ bit/örnek gerektirir. Genellikle uyarlanır melez kodlama dizgesi uyarlanır olmayan tekniđe göre özellikle düşük hızlarda birkaç dB lik bir iyileştirme getirir.

Üç çeşit kodlama şekli karşılaştırılacak olursa aşağıdaki sonuçlar görülebilir: Dönüşüm kodlamasıyla, kestirime dayalı kodlamalardan daha fazla veri azaltması elde edilebilir. Seviyelendirme veya kanal hataları nedeniyle oluşan herhangi bir bozulma, ters dönüşüm sırasında tüm görüntü üzerine azalmış bir şekilde yaydırılır. Bu bozulma, kestirim hatasından (kaynakta yerel olarak gözükten) daha az algılanabilir düzeydedir. Kestirim kodlamasının başarısı veri istatistiđindeki değışimlere çok duyarlıdır. Bu nedenle pratikte, yalnızca uyarlanır kestirim kodlama dizgelerinin başarımı (uyarlanır olmayan) dönüşüm kodlamalarının başarımına yaklařır. İki boyutlu çalışırken, sonlu dereceden nedensel kestiriciler dönüşüm kodlamasının veri azaltma yeteneđine ulaşamazlar. Çünkü, iki boyutlu bir rasgele alanın, sonlu dereceden nedensel bir gösterimi var olmayabilir. Fakat kestirim kodlamada, bellek gereksinimi ve işlem sayısı daha azdır. Bununla beraber, sayısal donanım ve bellek maliyetlerinin günümüzdeki sevindirici azalmasıyla dönüşüm kodlamasının ekonomikliđi de sorun olmaktan çıkmaktadır. Ayrıca hızlı dönüşüm algoritmaları kullanarak işlem sayısını azaltmak mümkündür. Bunlara örnek olarak hızlı ayrık Fourier dönüşümleri ve hızlı ayrık kosinüs dönüşümleri verilebilir. Bu tez çalışmasında, 8x8 boyutlu bir görüntü bloku için kullanılan hızlı AKD algoritmasında 16x11 çarpma ve 16x58 toplama işlemi yapılmaktadır. Kestirim kodlamalarının daha fazla veri istatistiđine bađımlı olmaları, hareket içerikli görüntülerde hatalara neden olur. Bu durumda, doğrusal kestirim kodlaması yerine uyarlanır kestirim kodlaması hatalara karşı daha bađışıklıdır. Kestirim kodlamalarının donanım olarak basitliđi dönüşüm kodlamalarına karşı bir üstünlük teşkil eder. Bir melez kodlayıcı gürültülü görüntülere ve görüntü istatistiđindeki değışimlere kolayca uyarlanıp, özellikle hareket içeriđi bulunan görüntülerde çerçeveler arası (interframe) görüntü data azaltımı için çok kullanışlı olabilir. Kanal hatalarına karşı TVKM' den daha az duyarlıdır, fakat dönüşüm kodlama kadar güçlü deđildir.

1.2.3.4. Hareket Kestirime Dayalı Kodlama

Video sinyallerinde genellikle büyük ölçüde zamansal ilişki bulunması nedeniyle bir görüntü çerçevesinde gözükten cisimler komşu görüntü çerçevelerinde de gözükmeye devam ederler. Hareket etmekte olan cisimler ise çerçeveden çerçeveye bir

miktar yer deęiřtirmiş olarak gözüktürler. Bu yerdeęiřtirme miktarını kestirme işleminde "hareket kestirme" denilmektedir. Yerdeęiřtiren (hareketli) görüntü elemanları arasında da çerçeve içi ve çerçeveler arası ilişkiler vardır. Çerçeve içi ilişkinin derecesi, hareketli cismin uzamsal detayına baęlıdır. Çerçeveler arası ilişki ise, hareket içeriğine baęlıdır. Çok yavaş hızlarda hareket eden cisimler, daha çok zamansal ilişki oluştururlar. Hareket hızı arttıkça, zamansal ilişki azalır. Bununla beraber, hareketin yönü ve hızı kestirilebilirse, bu ilişkilerden doğan artıklıkların yok edilmesi mümkündür. Böylece kodlamada hareket içeriğinden bağımsız olan bir veri hızı elde edilebilir. Bu kodlama şekline "hareket dengelemeli kodlama" (motion compensated coding) denilmektedir [1], [31], [32], [33], [34], [35].

Hareket dengelemeli kodlayıcı-kodçözücü blok şeması Şekil 12'de verilmektedir. Sistemin girişinde bir i 'nci görüntü çerçevesindeki iki boyutlu bir $M \times N$ boyutlu görüntü elemanı bloku I_i ile ve dönüşümü V_i ile gösterilsin. Blok içinde her (k,l) noktası için $V_i(k,l)$, $i=1,2,\dots$ dizisi tek boyutlu bir rasgele işlem (random process) olarak düşünülür ve uygun bir tek boyutlu TVKM metoduyla kodlanır. $\hat{V}_i(k,l)$ 'yi üretir ve bunun iki boyutlu ters dönüşümünü alır. Dönüşüm olarak ayrık kosinüs dönüşümü (AKD) ve her TVKM kanalı için bir birinci dereceden AR modeli kullanılabilir. Hareket dengelemeli melez kodlayıcıda kestirim hatası şu şekilde ifade edilebilir [1]:

$$e_i(k,l) = V_i(k,l) - \alpha \hat{V}_{i-1}(k,l) \quad (39)$$

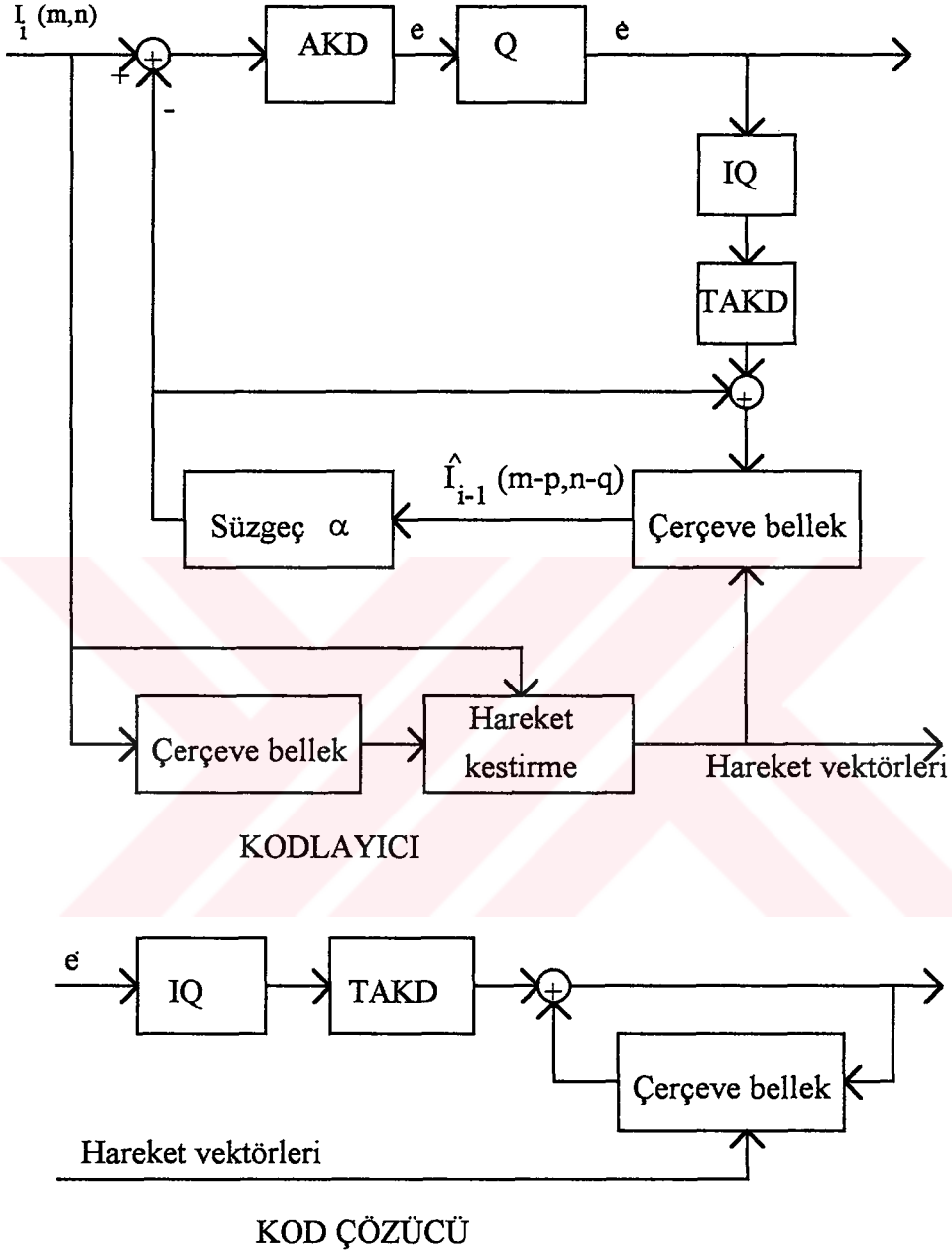
Burada, hareket kestirilerek elde edilmiş görüntü elemanı parlaklık fark değerleri dizisinin $(\hat{I}_{i-1}(m-p, n-q))$ dönüşümü $\hat{V}_{i-1}(k,l)$ ile gösterilmektedir. α ise kestirim katsayısıdır. Eğer α her (k,l) kanalı için sabitse, $e_i(k,l)$, $I_i(m,n) - \alpha \hat{I}_{i-1}(m-p, n-q)$ 'nin dönüşümüyle aynı olur. Böylece bir hareket dengelemeli melez kodlayıcı elde edilmiş olur.

Gürültünün etkisi hesaba katılmazsa, bir hareket dengelemeli kodlama tekniğinde, hareket sadece öteleme ise, hareketli görüntü elemanlarının gerçek ve kestirilmiş değerleri arasında fark olmaz. Bu da görüntü elemanları fark sinyalinin entropisini azaltır. Fakat hareket sadece öteleme olmayıp, dönme de içeriyorsa, yine de hareketi izlemek mümkündür, fakat bu durumda kestirim hatasının entropisi artar.

Hareketin tamamen dengelenemediği olaylarda, bazı zamanlarda kısa vadeli veri hızı, hareket dengeleme işlemi kullanılmayan duruma göre daha fazla olabilir (çünkü hareket bilgisini içeren yan bilgilerin de iletilmesi gerekmektedir).

Hareket kestirimi yapıldıktan sonra, bir çerçevedeki görüntü elemanları ile daha önceki çerçeveye ait olan ve hareketleri dengelenen görüntü elemanları arasındaki farklar kodlanırlar. Örneğin, burada bir dönüşüm kodlayıcı kullanılabilir. Görüntü elemanlarına

ait parlaklık fark değerleri genellikle küçük olduğu için, dönüşüm katsayıları da düşük entropiye sahip olurlar.



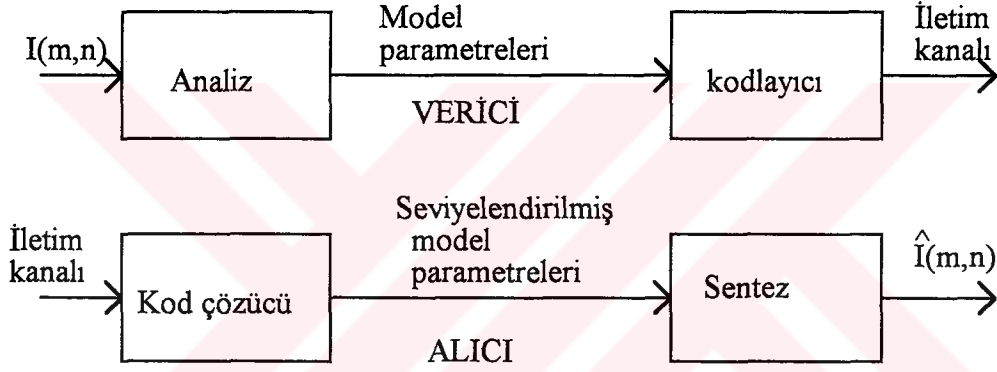
Şekil 12.Hareket dengeli bir kodlayıcı ve kodçözücü. AKD: Ayrık kosinüs Dönüşümü. TAKD: Ters ayrık kosinüs dönüşümü. Q: Seviyelendirici. IQ: Ters seviyelendirici.

Video sinyallerinde hareket kestirimi için geliştirilmiş bir çok algoritma vardır. Bu algoritmalar dört ana grupta toplanabilirler: türevsel teknikler (differential techniques), konum özyineli teknikler (pel-recursive techniques), Fourier dönüşümüne

dayalı teknikler ve blok karşılaştırma teknikleridir. Hareket içeriğinden arındırılmış (bağımsız) katsayılar (kestirim hataları veya dönüşüm katsayıları) belirlendikten sonra seçirim boyu (Run-length) kodlama veya istatistiksel bir kodlama uygulamak veri hızını daha da azaltır. Bu kodlama şekilleri kayıpsızdır [13], [36], [37].

1.2.3.5. Modele Dayalı Kodlama

Görüntü modeli oluşturarak kodlamada, bir görüntü veya görüntü parçası modellenir ve model parametrelerinden görüntü sentezlenir. Vericide, görüntü analiz edilerek model parametreleri belirlenir, alıcıda da kestirilmiş ve seviyelendirilmiş olan bu parametrelerden görüntü tekrar oluşturulur. Bir modele dayalı görüntü kodlama sistemi Şekil 13' de gösterilmektedir. Bu sistem, bir analiz/sentez sistemi gibi düşünülebilir.



Şekil 13. Modele dayalı kodlama verici ve alıcı.

Modele dayalı kodlayıcılar, dönüşüm ve kestirim kodlayıcılardan daha az veri hızlarında, anlaşılır (intelligible) görüntü sentezleme yeteneğine sahip olabilirler. Ancak, özellikle görüntülerin sentezini yapabilecek basit bir model geliştirmede bile, model parametrelerinin geliştirilmesi ve bunlardan bir görüntü sentezlenmesi çok işlem gerektirir. Modele dayalı kodlamada, görüntü parlaklıklarının detaylı gösterilmesi yerine, duvar, gökyüzü gibi arka planlar tanınır ve bunlara sadece basit modellerle sentezlenerek benzer arka planlar oluşturulur.

Tipik video telefon görüntülerinin çok düşük veri hızlarında modele dayalı kodlanması, günümüzde çok ümit verici olması nedeniyle bazı araştırmacılar tarafından ilgi çekmektedir [15], [24]. Modele dayalı kodlama teknikleri, diğer kodlama teknikleriyle birleştirilerek kullanılabilir. Örneğin, hareket kestirme tekniklerini içeren bir modele dayalı kodlama yöntemi mümkündür [24].

1.3. Video Sinyallerinde Hareket Kestirme Teknikleri

1.3.1. Türevsel Hareket Kestirme Teknikleri (Differential Motion Estimation Techniques)

Bu tekniklerde, görüntünün her noktası için uzamsal ve zamansal parlaklık değişim ölçüleri (gradient) belirlenir ve bundan yararlanarak hareketin bir kestirimi yapılır [38]. Sadece gözlenen görüntü çerçevelerini kullanarak hareket kestirimi yapabilmek için genellikle bazı varsayımlar yapılması gerekir:

1) Bir cisimin görüntüsünün parlaklık değerleri, bu cisimin hareket etmesiyle değişmez. Yani, her görüntü elemanının parlaklık değeri kendi hareket eğrisi boyunca sabittir.

2) iki ardışıl görüntü arasında üstüste çakışan iki noktanın parlaklığı değişmiş ise bu değişme bir hareketten dolayı olmuştur. Ayrıca, bir noktadaki parlaklık yerdeğiştirme miktarıyla doğrusal olarak değişir.

Bu varsayımlar "optik akı kısıtı" (optical flow constraint) olarak bilinir [39], [40]. Diğer bir varsayım, "düzgünlük kısıtı" (smoothness constraint) olarak bilinir ve görüntüde birbirine komşu olan görüntü elemanlarına ilişkin anlık hız vektörlerinin birbirinden çok farklı olmadıklarını belirtir [39], [40]. Hareket eden cisimlerin birbirlerini örtmediği (occlusion) ve hareket eden bir cisimin arka tarafından yeni cisimlerin ortaya çıkmadığı (covered/uncovered background) kabul edilmektedir. Pratikte bu varsayımların geçersiz olması hareketlerin hatalı kestirilmesine neden olur.

Hareket eden bir cismin ardışıl iki görüntü çerçevesindeki görüntülerinin parlaklık dağılımı arasındaki ilişki aşağıdaki gibi yazılabilir[38]:

$$I(x, y, t) = I(x - \delta x, y - \delta y, t - \tau) = I(Z - \delta d, t - \tau) \quad (40)$$

burada $I(x, y, t)$: bir cismin $Z = (x, y)^T$ uzamsal konumunda ve t zamanındaki görüntüsüne ilişkin parlaklık işlevi, τ : bir görüntü süresi, δd : bir cismin t ile $t - \tau$ zamanları arasındaki yer değiştirme miktarını gösterir. $\delta d = [\delta x, \delta y]^T$ olarak tanımlanmıştır. Bir cismin (x, y) noktasındaki anlık hızı çerçeveler arasındaki yerdeğiştirme ile bağımlı olduğu için, çerçeveler arasındaki zaman farkı bilindiği sürece yerdeğişim kestirimi veya anlık hız kestirimi yapılabilir. Anlık hız vektörlerinin iki boyutlu uzamsal koordinatlara göre dağılımına "optik akı" (optical flow) denir [41]. Böylece hareket kestirimi, yerdeğişim kestirimi ve optik akı kestirimi aynı anlamda kullanılabilir.

Türevsel tekniklerde, hareket vektörlerinin kestirilebilmesi için yukarıda anlatılan varsayımlardan hareket edilerek optik akı kısıtı ve düzgünlük kısıtı denklemleri

tanımlamak ve bu denklemleri çözmek yararlıdır. Optik akı kısıtını yazabilmek için (40) eşitliğinin sağ tarafı (x,y,t) noktası etrafında Taylor serisine açılır[42]:

$$I(x, y, t) = I(x, y, t) + \delta x \frac{\partial I}{\partial x} + \delta y \frac{\partial I}{\partial y} + \delta t \frac{\partial I}{\partial t} + e, \quad (x, y) \in \mathfrak{R} \quad (41)$$

burada e , açılımdaki ikinci ve daha yüksek dereceden terimleri gösterir. Eğer δx , δy ve δt çok küçükse yüksek dereceli terimler ihmal edilebilir. \mathfrak{R} ise aynı hareket eğrisine sahip olan hareketli görüntü elemanları bölgesini temsil etmektedir. e ihmal edilir, ortak terimler sadeleştirilir ve ifade δt ile bölünürse aşağıdaki ifade ortaya çıkar:

$$\frac{\delta x}{\delta t} \frac{\partial I}{\partial x} + \frac{\delta y}{\delta t} \frac{\partial I}{\partial y} + \frac{\partial I}{\partial t} + Q(\delta t) = 0 \quad (42)$$

burada $Q(\delta t)$, δt sıfıra giderken limiti alındığında ihmal edilebilen bir terimdir. Hareket eden bir cisme ait parlaklığın, yer değiştirme miktarıyla doğrusal olarak değiştiği kabul edilerek (yer değiştirme, görüntünün içerdiği en yüksek frekansın dalga boyundan daha küçük kabul edilerek) yerdeğiştirme vektörü için aşağıdaki ifade yazılabilir[1]:

$$\mathbf{d} = \int_{t_0}^{t_0+\tau} \mathbf{v} dt \approx \mathbf{v} \tau \quad (43)$$

Buna göre, (42) ifadesi aşağıdaki gibi de yazılabilir:

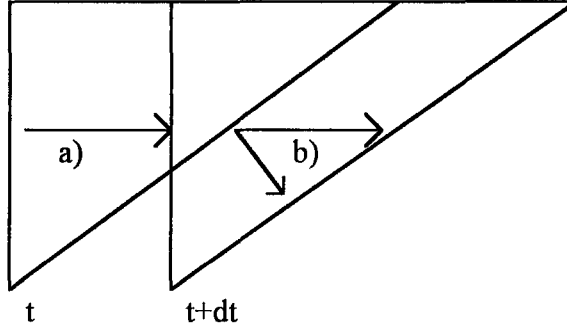
$$v_1 \frac{\partial I}{\partial x} + v_2 \frac{\partial I}{\partial y} + \frac{\partial I}{\partial t} = 0 \quad (44)$$

veya

$$\mathbf{v}(x, y, t) \nabla I(x, y, t) + \frac{\partial I(x, y, t)}{\partial t} = 0 \quad (45)$$

burada $\mathbf{v}(x,y,t)$ bir t zamanında ve (x,y) konumundaki cismin hızını, $\mathbf{v}(\cdot) \nabla I(\cdot)$ ise \mathbf{v} vektörü ile görüntü parlaklık işlevi I 'nin uzamsal değişim ölçüsünün yönsüz (scalar) çarpımını gösterir. $v_1 = \delta x / \delta t$, $v_2 = \delta y / \delta t$ iki bilinmeyen anlık hız bileşenidir. Yukardaki (44) (veya (45)) ifadesine "uzamsal-zamansal kısıtlama denklemi" (spatio-temporal constraint equation) denir [24].

Optik akı kısıtı tek başına hareket kestirim problemini tanımlamaz. Çünkü yerdeğişim ve anlık hız vektörleri her noktada iki boyutlu olduğu halde optik akı her noktada tek bir kısıt denklemini verir. Bu durumda optik akı kısıtı anlık hız vektörünün sadece $[\partial I / \partial x \ \partial I / \partial y]$ yönündeki bileşenini verir. Şekil 14' de optik akı kısıtının yetersizliği gösterilmiştir. a) durumunda kenar harekete dik olduğu için kestirilen hareket doğru, b) durumunda ise doğru değildir.



Şekil 14. Optik akı kısıtının geçersiz olduğu durum

Anlık hız veya yerdeğişim vektörlerinin her iki bileşenini de tanımlayabilmek için her noktada en az iki denklem gerekir. Birden fazla denklem elde edebilmek için $\partial I / \partial x$, $\partial I / \partial y$, $\partial I / \partial t$ 'yi (v_1 ve v_2 ' yi sabit varsayarak) bir çok K_i : (x_i, y_i, t_i) , $1 \leq i \leq N$ noktasında değerlendirerek çözümü bulunabilen bir doğrusal denklemler takımı elde edilir[24]:

$$v_1 \frac{\partial I}{\partial x} \Big|_{(K_i)} + v_2 \frac{\partial I}{\partial y} \Big|_{(K_i)} + \frac{\partial I}{\partial t} \Big|_{(K_i)} \approx 0, \quad 1 \leq i \leq N \quad (46)$$

Hız kestirimi de aşağıdaki ifade enküçüklenerek (minimize edilerek) yapılır[24]:

$$\text{Hata} = \sum_{i=1}^N \left[v_1 \frac{\partial I}{\partial x} \Big|_{(K_i)} + v_2 \frac{\partial I}{\partial y} \Big|_{(K_i)} + \frac{\partial I}{\partial t} \Big|_{(K_i)} \right]^2 \approx 0, \quad 1 \leq i \leq N \quad (47)$$

Yukardaki hata ifadesi v_1 ve v_2 ' bilinmeyen parametrelerin bir karesel (quadratic) formu olduğu için, çözüm için iki doğrusal denklem gerekir. Daha genel olarak, uzamsal-zamansal kısıtlama denkleminin \mathfrak{R} ile gösterilen bir uzamsal-zamansal bölgede geçerli olduğu varsayılarak, hız bileşenlerini kestirmek için aşağıdaki denklem enküçüklenir[24].

$$\text{Hata} = \left[\iiint_{(x,y,t) \in \mathfrak{R}} v_1 \frac{\partial I}{\partial x} \Big|_{(K_i)} + v_2 \frac{\partial I}{\partial y} \Big|_{(K_i)} + \frac{\partial I}{\partial t} \Big|_{(K_i)} \right]^2 dx dy dt \quad (48)$$

Buradaki tümlev yerine toplam yazılırsa (47) ifadesi elde edilir. Yukardaki hata ifadesini v_1 ve v_2 'ye göre türeterek ve sonucu sifira eşitleyerek aşağıdaki ifadeler elde edilir:

$$\mathbf{W} \mathbf{v} = \Gamma \quad (49)$$

burada w , v ve Γ aşağıdaki gibidir:

$$\mathbf{W} = \begin{bmatrix} \iiint_{(x,y,t) \in \mathfrak{R}} \left(\frac{\partial I}{\partial x} \right)^2 dx dy dt & \iiint_{(x,y,t) \in \mathfrak{R}} \frac{\partial I}{\partial x} \frac{\partial I}{\partial y} dx dy dt \\ \iiint_{(x,y,t) \in \mathfrak{R}} \frac{\partial I}{\partial x} \frac{\partial I}{\partial y} dx dy dt & \iiint_{(x,y,t) \in \mathfrak{R}} \left(\frac{\partial I}{\partial y} \right)^2 dx dy dt \end{bmatrix} \quad (50)$$

$$\mathbf{v} = [v_1, v_2]^T \quad (51)$$

$$\Gamma = \begin{bmatrix} \iiint_{(x,y,t) \in \mathfrak{R}} \frac{\partial I}{\partial x} \frac{\partial I}{\partial t} dx dy dt \\ \iiint_{(x,y,t) \in \mathfrak{R}} \frac{\partial I}{\partial y} \frac{\partial I}{\partial t} dx dy dt \end{bmatrix} \quad (52)$$

Yukardaki iki doğrusal denklemin bir çok çözümü olabilir. Bu nedenle çözümde bazı sorunlar mevcuttur. Örneğin, I uzamsal-zamansal \mathfrak{R} bölgesinde sabit olsun. $\partial I / \partial x$, $\partial I / \partial y$ ve $\partial I / \partial t$ ayrı ayrı sıfır, \mathbf{W} ve Γ matrislerindeki bütün elemanların da sıfır olduğu varsayılsın. Bu durumda herhangi (v_1, v_2) (49)'u sağlar. Bunun anlamı, düzgün parlaklığa sahip bir bölgedeki herhangi bir hız I 'ya etki etmez ve böylece gerçek hız I dan kestirilemez. Keskin bir kenar olması durumunda da bu kenara paralel yöndeki hız I ya etki etmez ve kestirilemez. Bu problemlere bir çözüm aşağıdaki gibi önerilmektedir [24]:

λ_1 ve λ_2 W matrisinin öz değerlerini ve α_1 ve α_2 de bunlara karşılık gelen ortonormal öz vektörleri (orthonormalized eigenvectors) gösterebilir. (49) ifadesinin çözümü için üç durum belirlenebilir:

$$1. \text{ durum: } v=0, \lambda_1, \lambda_2 < \text{eşik} \quad (53)$$

$$2. \text{ durum: } v = \left(\left[\alpha_1^T \Gamma \right] \alpha_1 \right) / \lambda_1, \quad \lambda_1 \ll \lambda_2 \quad (54)$$

$$3. \text{ durum: } v = W^{-1} \Gamma, \quad 1 \text{ ve } 2 \text{ nin dışındaki şartlar} \quad (55)$$

1. durumda, düzgün bir parlaklık bölgesi üzerinde çalışılıyorsa hız sıfıra setlenir. 2. durum tam bir keskin kenarı gösterir. (54)' deki hız kestirimi kenara dik yön boyuncadır. 3. durumda (49)' daki doğrusal denklemleri çözmek gerekir. (49)' daki doğrusal denklemleri çözmek için türevlerin bir kaç uzamsal-zamansal konum için hesaplanması gerekmektedir. Bunun için uzamsal polinomsal aradeğerleme metodu üç boyut için kullanılabilir. Bunun diğer yöntemlere göre üstünlüğü, hesapsal basitliği ve gürültüye karşı güçlü olmasıdır. Üç boyutlu aradeğerlemede, aradeğerlenmiş $I(x,y,t)$ [24],

$$\hat{I}(x, y, t) = \sum_{i=1}^N S_i \phi_i(x, y, t) \quad (56)$$

olarak bulunur. (49)' daki kısmi türevler S_i cinsinden hesaplandıktan sonra, S_i katsayıları aşağıdaki işlev enküçüklenerek bulunur[24]:

$$\text{Hata} = \sum_{n_1} \sum_{n_2} \sum_{n_3 \in \mathbb{N}} \left[I(x, y, t) - \sum_{i=1}^N S_i \phi_i(x, y, t) \right]^2 \Bigg|_{x = n_1 T_1, y = n_2 T_2, t = n_3 T_3} \quad (57)$$

(49)' daki kısmi türevler S_i cinsinden daha önce hesaplandığına göre (57)'deki hata ifadesini S_i ' ye göre enküçükleme için uygun bir \mathfrak{R} bölgesini kapsayan bir doğrusal denklemler takımının çözümü gerekir.

Yukarda anlatılan hareket kestirme yöntemi, yeknesak (uniform) öteleme hareketlerinin varsayıldığı uzamsal-zamansal alanların belirlenmesini gerektirir. Uygun bir \mathfrak{R} ' nin saptanması birkaç görüntü elemanı boyundaki doğru yerdeğiştirmeye sahip bir ilk yerdeğişim kestirimini gerektirir. Pratikte video telefon görüntülerinde iki ardışıl çerçeve arasında 10 görüntü elemanı boyundan daha fazla farklılıklar olmaz. İlk

yerdeğiştirme (veya hız) kestirimi için bir yaklaşım bitişik görüntü elemanı için daha önce hesaplanan yerdeğiştirmeyi kullanmak ve buna göre uygun \mathfrak{R}' yi belirlemektir.

(45) ifadesinde iki hız bileşeninin bulunabilmesi için teknik yazında, düzgünlük kısıtını sağlayan diğer çözümler de önerilmiştir. Bunlardan bazıları aşağıdaki gibidir:

1) v_1 ve v_2 iki hız bileşeninin değişim ölçülerinin kare genlikleri Horn and Schunck [38] tarafından bir düzgünlük kısıtı olarak kullanıldı:

$$\left(\frac{\partial v_1}{\partial x}\right)^2 + \left(\frac{\partial v_1}{\partial y}\right)^2 \text{ ve } \left(\frac{\partial v_2}{\partial x}\right)^2 + \left(\frac{\partial v_2}{\partial y}\right)^2$$

$$e_c^2 = \left\| \left[\frac{\partial(dx/dt)}{\partial x} \quad \frac{\partial(dx/dt)}{\partial y} \right]^T \right\|^2 + \left\| \left[\frac{\partial(dy/dt)}{\partial x} \quad \frac{\partial(dy/dt)}{\partial y} \right]^T \right\|^2 \quad (58)$$

Burada da anlık hız vektörünün kestirimi bir kısıtlı en iyileme (constrained optimization) problemine dönüşür. Yani hata işlevini optik akı kısıtı altında enküçükleyen dd/dt değeri aranır. Uygulamada görüntülerde gürültü bulunduğu ve türevler sonlu farklarla yaklaşıldığından optik akı kısıtı kesin olarak sağlanamaz. Bu durumda yeni bir maliyet işlevi tanımlanabilir:

$$e^2 = \int_{\mathfrak{R}} (c^2 e_c^2 + e_b^2) dd \quad (59)$$

burada c^2 bir sabit, \mathfrak{R} ise ilgilenilen bölgeyi gösterir. Anlık hız vektörü kestirimi bu yeni maliyet işlevini enküçükleyen değer olarak alınır. Bu enküçükleme hesabı değişiler hesabı (calculus of variations) kullanılarak gerçekleştirilirse, her noktada şu iki denklem elde edilir:

$$\left[\frac{\partial I}{\partial x} \frac{dx}{dt} + \frac{\partial I}{\partial y} \frac{dy}{dt} - \frac{\partial I}{\partial t} \right] \frac{\partial I}{\partial x} = c^2 \frac{\partial^2}{\partial x^2} \frac{dx}{dt} + \frac{\partial I}{\partial x} \frac{\partial I}{\partial t} \quad (60)$$

$$\left[\frac{\partial I}{\partial x} \frac{dx}{dt} + \frac{\partial I}{\partial y} \frac{dy}{dt} - \frac{\partial I}{\partial t} \right] \frac{\partial I}{\partial y} = c^2 \frac{\partial^2}{\partial x^2} \frac{dy}{dt} + \frac{\partial I}{\partial y} \frac{\partial I}{\partial t}$$

burada $\partial^2 / \partial x^2 = \nabla^2$ laplasyen (laplacian) ve "." ise altındaki miktarın kestirimini gösterir. Horn and Schunck bu doğrusal olmayan denklemler takımını her noktada Gauss-Siedel yöntemi ile çözmüşlerdir [42].

2) v_1 ve v_2' nin Laplasyenlerinin (Laplacian) kareleri toplamı Horn and Schunck [42]:

$$\nabla^2 v_1 = \frac{\partial^2 v_1}{\partial x^2} + \frac{\partial^2 v_1}{\partial y^2} \quad \text{ve} \quad \nabla^2 v_2 = \frac{\partial^2 v_2}{\partial x^2} + \frac{\partial^2 v_2}{\partial y^2} \quad (61)$$

3) Başka bir seçenek olarak Treitak ve Pastor Laplasyen' i enküçüklemeyi bir düzgünlük kısıtı olarak önerdiler [43]. Bu yöntemde, parlaklığın sürekli ikinci dereceden türevlere sahip olduğu varsayılmakta ve optik akış denklemi x ve y ' ye göre türetilmektedir. Böylece iki bilinmeyenli iki doğrusal denklem elde edilmektedir:

$$\frac{\partial^2 I}{\partial x \partial t} = -\frac{\partial^2 I}{\partial x^2} v_1 - \frac{\partial^2 I}{\partial x \partial y} v_2 \quad (62)$$

$$\frac{\partial^2 I}{\partial y \partial t} = -\frac{\partial^2 I}{\partial x \partial y} v_1 - \frac{\partial^2 I}{\partial y^2} v_2$$

Bu sisteme ilişkin matrisin determinanı, bir t zamanında Gauss eğrisi şeklinde bir parlaklık yüzeyini temsil eder. Bu, güvenilir hız kestirimini sağlar. Çünkü, Gauss eğrisinin yüksek yerlerinde kontrast fazladır ve buralardaki hız kestirimi daha güvenilir olur. İkinci türevlerin sürekli olmasını sağlamak için görüntüyü bir alçak geçiren süzgeçten geçirmek yararlıdır.

4) Yerdeğiştiren cisimlerin sınırlarındaki hız bileşenlerinde ani değişiklikler olabilir. Bu durumda düzgünlük kısıtı uyarlamalı olarak kullanılabilir. Bu amaçla Nagel ve Enkelmann [44] ve Fogel [45] ağırlıklı ve yönlü düzgünlük kısıtları önerdiler. Burada, hareket nedeniyle parlaklık düzeylerindeki değişme problemi, doğrusal parlaklık düzeyi değişmesi modeli ile sınırlanmamaktadır. Bu durumda, parlaklık işlevi $x = x_0, y = y_0$ noktasında Taylor serisine açılır ve ikinci dereceden terimler ihmal edilmez:

$$I(x, y) = I(x_0, y_0) + \frac{\partial I}{\partial x}(x - x_0) + \frac{\partial I}{\partial y}(y - y_0) + \frac{1}{2} \frac{\partial^2 I}{\partial x^2}(x - x_0)^2 + \quad (63)$$

$$\frac{\partial^2 I}{\partial x \partial y}(x - x_0)(y - y_0) + \frac{1}{2} \frac{\partial^2 I}{\partial y^2}(y - y_0)^2$$

Problem, küçük bir \mathfrak{R} bölgesi için (64) ifadesinin enküçüklenmesidir:

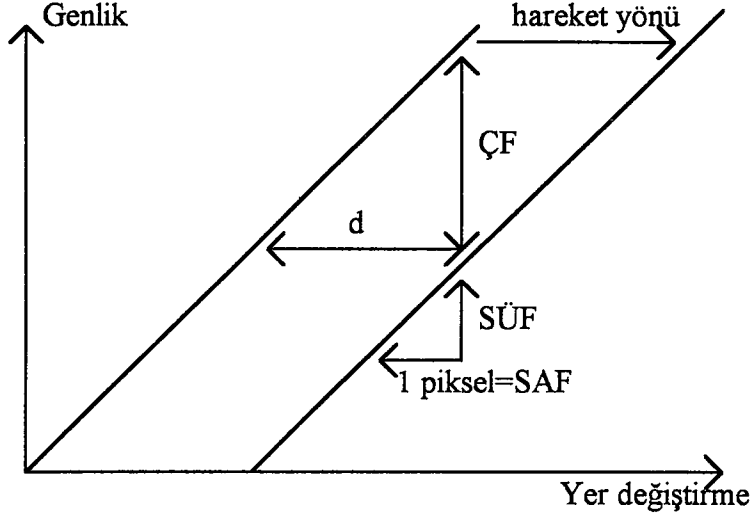
$$\iint \left[I(x, y) - I(x_0, y_0) - \frac{\partial I}{\partial x}(x - v_1) - \frac{\partial I}{\partial y}(y - v_2) - \frac{1}{2} \frac{\partial^2 I}{\partial x^2} (x - v_1)^2 - \frac{\partial^2 I}{\partial x \partial y} (x - v_1)(y - v_2) - \frac{1}{2} \frac{\partial^2 I}{\partial y^2} (y - v_2)^2 \right]^2 dx dy \quad (64)$$

Buradaki enküçükleme, " (x_0, y_0) 'daki (v_1, v_2) hızı ile oluşan yerdeğiştirmeden oluşan parlaklık değişmesi yeni çerçevedeki komşu görüntü elemanları için en iyi parlaklık düzeyi yaklaşımını verir" varsayımını ifade eder. Enküçükleme, hız bileşenlerinin üçüncü kuvvetlerini içeren iki doğrusal olmayan (nonlinear) denklem ortaya çıkarır. Koordinat sistemi, (x_0, y_0) 'da ana eğri yönünde olsun. Bu, $\partial^2 I / \partial x \partial y = 0$ olmasını sağlar. Geri kalan ikinci türevler: $\partial^2 I / \partial x^2$ ve $\partial^2 I / \partial y^2$ dir. Bunlardan en azından biri maksimum gradient elde etmek için (x_0, y_0) 'da sıfırdan geçmelidir. Örneğin, $\partial^2 I / \partial x^2 = 0$, $\partial I / \partial x \rightarrow$ maksimum ve $\partial I / \partial y = 0$. Bu varsayımlar altında integral de toplam olarak yaklaşıtırlırsa, aşağıdaki denklemin enküçüklenmesi yeterli olur:

$$\sum_{x, y \in \mathcal{R}} \left[I(x, y) - I(x_0, y_0) - \frac{\partial I}{\partial x}(x - v_1) - \frac{1}{2} \frac{\partial^2 I}{\partial y^2} (y - v_2)^2 \right]^2 \rightarrow \text{minimum} \quad (65)$$

Yukardaki denklem v_1 ve v_2 'ye göre türetilir ve sifıra eşitlenirse, iki hız bileşenini içeren iki denklem elde edilir. Bu denklemlerin çözümüyle hız bileşenleri bulunur.

5) Bir bölgedeki hareket kestirim değerlerinin doğrusal bir birleşimi kullanılarak düzgünlük kısıtı sağlanmaya çalışılabilir [38], [46], [47]. Sayısal görüntüler için uygulanması kolay olan böyle bir yöntem aşağıda anlatılmaktadır: Parlaklık değişim ölçüleri sütun fark (SÜF) ve satır fark (SAF) olarak yaklaşıtırlı olarak hesaplanmaktadır [38] (Şekil 15).



Şekil 15. Türevsel hareket kestirme teknięi (ÇF: Görüntü çerçeveleri arasındaki parlaklık farkı deęeri, SÜF: iki satırdaki pikseller arasındaki genlik fark deęeri, SAF: aynı satırda iki piksel arasındaki genlik farkı deęeri).

Bir Z uzamsal konumu için çerçeve fark sinyali (ÇF) ařaęıdaki gibi tanımlanmaktadır[38]:

$$\begin{aligned} \text{ÇF}(Z) &= I(Z,t) - I(Z, t - \tau) \\ &= I(Z,t) - I(Z+d,t) \end{aligned} \quad (66)$$

Eřitlięin saę tarafı z noktası etrafında Taylor serisine aılıp sadece birinci dereceden terimler alınırsa ortaya ıkan optik akı denklemi doęrusal regresyon (linear regression) [47] ile özölürse d hareket vektörünün kestirimi ařaęıdaki yaklařtırılmıř ifadeyle bulunabilir[38]:

$$d = - \begin{bmatrix} \sum \text{SÜF}^2(Z) & \sum \text{SÜF}(Z) \cdot \text{SAF}(Z) \\ \sum \text{SAF}(Z) \cdot \text{SÜF}(Z) & \sum \text{SAF}^2(Z) \end{bmatrix}^{-1} \begin{bmatrix} \sum \text{ÇF}(Z) \cdot \text{SÜF}(Z) \\ \sum \text{ÇF}(Z) \cdot \text{SAF}(Z) \end{bmatrix} \quad (67)$$

Buradaki toplamlar, hareket içerięi bulunan sınırları belirlenmiř alanlar için geçerlidir. Yukardaki eřitlik,

$$\sum \text{SÜF}(Z) \cdot \text{SAF}(Z) = 0 \quad (68)$$

varsayılarak basitleştirilebilir [46]. Çarpma işlemlerinden kurtulmak için daha da basitleştirilerek aşağıdaki gibi yazılabilir:

$$d = - \begin{bmatrix} \sum \text{ÇF}(Z) \text{sign}(\text{SÜF}(Z)) / \sum |\text{SÜF}(Z)| \\ \sum \text{ÇF}(Z) \text{sign}(\text{SÜF}(Z)) / \sum |\text{SAF}(Z)| \end{bmatrix} \quad (69)$$

burada,

$$\text{sign}(z) = \begin{cases} 0, & z = 0 \text{ ise} \\ z/|z|, & \text{diğer durumlarda} \end{cases}$$

olarak tanımlanmıştır.

Yukardaki hareket kestirme algoritması, d küçük olduğu sürece iyi sonuç vermektedir. d büyüdükçe yapılan yaklaşımların geçerliliği azalacağından kestirimin doğruluğu azalır. Bu hata, parlaklık işlevini d^i nin bir ilk kestirim değeri etrafında doğrusallaştırarak azaltılabilir. İnci görüntü çerçevesindeki hareket vektörünün kestirim değeri d^i , parlaklık işlevini $(i-1)$ 'inci görüntü çerçevesi için kestirilen yerdeğiştirme etrafında doğrusallaştırarak bulunabilir. Bu işlem aşağıdaki özyineli ifadeyi ortaya çıkarır:

$$d^i = d^{i-1} + U^i \quad (70)$$

burada d^{i-1} , d^i 'nin bir ilk kestirimi ve U^i , d^{i-1} 'in bir düzeltilmiş değeridir (örneğin: $d - d^{i-1}$ 'in bir kestirimi).

ÇF işlevine benzer olarak, yerdeğiştirmiş çerçeveler farkı (YÇF) işlevi de aşağıdaki gibi tanımlanabilir:

$$\text{YÇF}(Z, d^{i-1}) = I(Z, t) - I(Z - d^{i-1}, t - \tau) \quad (71)$$

Hareket vektörü aşağıdaki gibi elde edilir:

$$d^i = d^{i-1} - \begin{bmatrix} \sum \text{YÇF}(Z, d^{i-1}) \text{sign}(\text{SÜF}(Z)) / \sum |\text{SÜF}(Z)| \\ \sum \text{YÇF}(Z, d^{i-1}) \text{sign}(\text{SÜF}(Z)) / \sum |\text{SAF}(Z)| \end{bmatrix} \quad (72)$$

Yine buradaki toplamlar da hareket içerikli alanlarda geçerlidir. Eğer d^{i-1} in ilk kestirimi bir tamsayı değer ise, $Y\check{C}F(.,.)$ işlevi aradeğerleme işlemine gerek duyulmadan hesaplanabilir. d vektörünün bir tamsayı yaklaşığı, bu vektörün bileşenlerini birer tamsayıya yuvarlatmakla yapılabilir. $[d]$ sembolü, d 'nin tamsayı yaklaşımını gösterebilir. Böylece (72) ifadesi aşağıdaki gibi yazılabilir:

$$d^i = [d^{i-1}] - \left[\frac{\sum Y\check{C}F(Z, [d^{i-1}]) \text{sign}(S\check{U}F(Z)) / \sum |S\check{U}F(Z)|}{\sum Y\check{C}F(Z, [d^{i-1}]) \text{sign}(S\check{A}F(Z)) / \sum |S\check{A}F(Z)|} \right] \quad (73)$$

Görüntüde sadece yeknesak bir öteleme hareketi olması çoğu zaman geçerli değildir. Örneğin, cisimlerin görüntülerindeki dönme hareketleri, büyüme/küçülme, arka planların ortaya çıkması veya farklı hızlarda hareket eden cisimlerin olması kestirimin doğruluğunu olumsuz etkiler. Yeknesak öteleme, hareketin sadece yerel olduğu varsayılarak, hareket parametreleri her görüntü elemanı için veya küçük bölgeler için daha doğru kestirilebilir [48], [49]. Ayrıca, hareket kestiriminin kesin olarak yapılamadığı bölgeler belirlenerek bu bölgelerde hareket dengeleme işlemi durdurulabilir.

1.3.2. Konum Özyineli Hareket Kestirme Teknikleri (Pel Recursive Motion Estimation Techniques)

Konum özyineli yerdeğişim kestirimi tekniklerinde görüntü dizilerindeki uzamsal ve zamansal değişim ölçüleri arasındaki bağımlılık ilişkileri kullanılır [38], [50], [51], [52], [53]. Bu tekniklerde de hareket vektörleri bulunurken yine optik akı kısıtı ve düzgünlük kısıtı göz önüne alınır.

Konum özyineli yerdeğişim kestiriminde, görüntü üzerindeki hareketli alanlara ait her görüntü elemanı için bir yerdeğiştirmiş çerçeveler farkı ($Y\check{C}F$) işlevinin karesi enküçüklenir. Elde edilen yerdeğiştirme bilgisi sıradaki görüntü elemanının yerdeğiştirmesini bulmak için yapılacak tekrarlar ilk değer olarak alınarak işlem sayısı azaltılmış olur. Alıcıya sadece başarılı olarak kestirilemeyen görüntü elemanı değerleri için hata ve adres bilgilerini göndermek yeterlidir.

Konum özyineli hareket kestirimi için çerçeveler farkı işlevi ($\check{C}F$) ve yerdeğiştirmiş çerçeveler farkı işlevi ($Y\check{C}F$) aşağıdaki gibi tanımlanabilir [53]:

$$\check{C}F = I(Z, t) - I(Z, t - \tau) \quad (74)$$

$$Y\check{C}F(Z, d) = I(Z, t) - I(Z - d, t - \tau) \quad (75)$$

$I(Z,t)$: $Z=(x,y)^T$ koordinatlarındaki görüntü elemanın parlaklık değeridir. $d = (dx, dy)^T$ ise hareket vektörü d' nin bir kestirim değeridir.

Kestirme işlemi, $YÇF^i$ yi sıfır yapan veya sıfıra yaklaştıran bir d hareket vektörü aranmasıyla başlar. Z' deki görüntü elemanı d^{i-1} yerdeğişimi (i : tekrar indisi) ve $I(Z-d^{i-1}, t - \tau)$ parlaklık değeri ile kestirildiğinde, (76)'daki kestirim hatası $YÇF$ ortaya çıkar:

$$YÇF(Z, d^{i-1}) = I(Z, t) - I(Z - d^{i-1}, t - \tau) \quad (76)$$

Burada, $d^{i-1} = (u^{i-1}, v^{i-1})^T$ dir.

Her görüntü elemanı için, $|YÇF(Z, d^i)| \leq |YÇF(Z, d^{i-1})|$ olacak şekilde yeni bir d^i değeri aranır. Bu amaçla $[YÇF(Z, d^{i-1})]^2$ işlevi, hareket içerikli alanlardaki görüntü elemanları için değişim ölçüsünü belirlemeye dayalı bir yaklaşımla aşağıdaki denklem kullanılarak özyineli olarak enküçüklenir:

$$d^i = d^{i-1} - k \cdot YÇF(Z, d^{i-1}) \cdot \Delta I(Z - d^{i-1}, t - \tau) \quad (77)$$

Burada k , pozitif küçük bir sayı, I ise $I(t - \tau)$ ' nun uzamsal değişim ölçüsüdür. Her tekrarda (iterasyonda) yeni bir d^i elde edilir. Eğer $|YÇF|$ önceden belirlenmiş bir eşik seviyesinden küçükse yer değiştirme vektörü belirlenmiş olur ve tekrar işlemi sona erer. Bir noktada tekrarlamalı olarak elde edilen yerdeğişim kestirimi gelecek noktadaki (yeni görüntü elemanı üzerindeki) tekrarlar için ilk değer olarak alınabilir. Bu noktada hesaplanan $|YÇF|$ eşik seviyesinden büyük değilse bu görüntü elemanı için yeni bir tekrar işlemi başlamaz. Eşik seviyesi aşılana görüntü elemanı için, yakınsama tekrar bulununcaya kadar tekrar işlemi sürdürülür. Böylece her görüntü elemanına ilişkin bir yer değiştirme vektörü az sayıda tekrar işlemiyle bulunur. Bu şekilde yapılan kestirime "konum özyineli yerdeğişim kestirimi" adı verilir. Yukardaki (76) eşitliğinin basitleştirilmiş bir şekli [38] aşağıdaki gibidir:

$$d^i = d^{i-1} - k \cdot \text{sign}\{YÇF(Z, d^{i-1})\} \cdot \text{sign}\{\text{grad } I(Z - d^{i-1}, t - \tau)\} \quad (78)$$

$\text{sign}(\cdot)$ işlevi sadece 0, 1 ve -1 değerlerini alabildiği için düzeltme vektörü sadece 45° nin katlarındaki açılarda olabilir. Bu düşünce kestirim hatasını fazla arttırmaz [38]. Buna göre, her tekrarda d vektörü k kadar düzeltilmektedir. Küçük k değerleri için daha doğru kestirimler yapılabilir, fakat yöntemin yakınsama hızı yavaşlar. Yukardaki (78) eşitliği, yakınsamanın bulunmasındaki tekrar sayısı hakkında bilgi vermesi bakımından da

yararlıdır. Örneğin, d^i ile d^{i-1} arasında iki görüntü elemanı aralığı kadar bir fark varsa ve $k=1/16$ seçilmişse, gerçek yerdeğiştirmeyi bulmak için 32 tekrar gerekir. Bu ise görüntü elemanı başına işlem sayısının fazla olacağını göstermektedir. Yakınsama hızını arttırmak ve yer değiştirmeyi doğru olarak kestirebilmek için düşünülen iki çözüm şunlardır:

1) Uygun bir k değeri seçmek ve her hareketli alana ait görüntü elemanları için tekrar sayısını 1 veya 2 olarak sınırlamak. Bu şekilde yapılan kestirimin yine de iyi sonuçlar verdiği [38] belirtilmektedir. Bunun nedeni, video telefonlardaki tipik görüntülerin istatistikleri nedeniyle, komşu iki hareket vektörü arasında çok büyük farklar ortaya çıkmamasıdır.

2) Bir görüntü elemanına komşu olan diğer bir görüntü elemanının yer değiştirmesi de yaklaşık aynıdır. Fakat cisim kenarlarına rastlayan yerlerde veya cisim üzerinde şekil bozulmaları olması durumunda bu bölgelerdeki yerdeğişim vektörlerinin benzerliği azalır. k katsayısını değişken seçerek bu sorun giderilebilir. Yukarıda elde edilen (76) eşitliğinin sağ tarafındaki ikinci terime "düzeltme terimi" [54] adı verilebilir. $|YÇF|$ büyük ve $|\text{grad } I|$ küçükse vektör düzeltim değeri büyük seçilir. Eğer $|YÇF|$ küçük ve $|\text{grad } I|$ büyükse burada bir cisim kenarı olduğu anlaşılabilir ve vektör düzeltim değeri küçük seçilir. Walker and Rao [54], yakınsama hızını belirleyen k' ya "ivme" adını vererek aşağıdaki gibi belirlemiştir:

$$k = \frac{1}{2(|\text{grad } I(Z - d^{i-1}, t - \tau)|)^2} \quad (79)$$

Cafforio and Rocca [47] k için ilişki tekniğine dayanan benzer bir bağıntı bulmuştur:

$$k = \frac{1}{\sigma^2 + |\text{grad } I(Z - d^{i-1}, t - \tau^2)|} \quad (80)$$

Burada $|\text{grad } I|$ çok küçük veya sıfır olduğunda σ' 'nin varlığı gerekli olmaktadır. k , değişim ölçüsü değerinin genliğiyle ters orantılı olduğu için düzeltme terimi kenar eğimlerinden (contours slopes) bağımsız olur ve yakınsama hızı artar.

Yer değiştirme vektörü \mathbf{d} , tamsayı (d^i) ve kesirli (d^f) parçaların toplamı olarak gösterilebilir. Böylece bir görüntü elemanına ait I parlaklık değeri ve $YÇF$, bir önceki yer değiştirme vektörünün bir görüntü üzerine rastlamayan (kesirli) bileşenlerini hesaba katarak standart iki boyutlu doğrusal aradeğerlemeyle bulunabilir [38]:

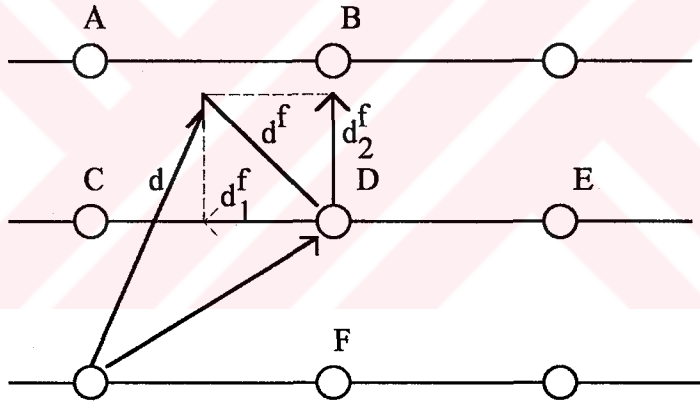
$$I(Z - d^{i-1}, t - \tau) = (1 - d_2^f) \left[(1 - d_1^f) I_D + d_1^f I_C \right] + d_2^f \left[(1 - d_1^f) I_B + d_1^f I_A \right] \quad (81)$$

Buradaki aradeğerleme işlemi yerine işlem sayısını azaltmak için basitleştirilmiş bir aradeğerleme formülü kullanılabilir:

$$I(Z - d^{i-1}, t - \tau) = I_D + d_1^f (I_C - I_D) + d_2^f (I_B - I_D) \quad (82)$$

gradI, yaklaşık bir hesaplamayla bulunur. Bunun için sütun fark (SÜF) ve satır fark (SAF) işlevleri tanımlanır. Örneğin, (83) eşitliği ile verilen yaklaşımda E ve C konumundaki görüntü elemanlarının parlaklık farkı ile B ve F konumundaki görüntü elemanlarının parlaklık farkı kullanılmaktadır.

$$\text{SÜF} = \frac{(I_B - I_A + I_D - I_C)}{2}, \quad \text{SAF} = \frac{(I_A - I_C + I_B - I_D)}{2} \quad (83)$$

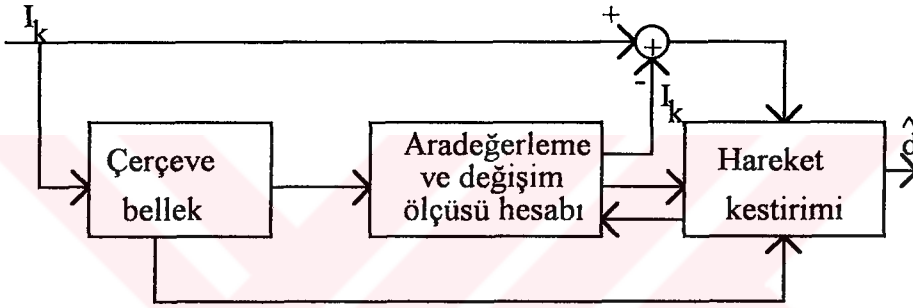


Şekil 16. Aradeğerleme işlemi.

Yöntemin sayısal video sinyallerine uygulanmasında, her görüntü elemanı üzerinde yapılacak tekrar sayısı sınırlandırılır. Bu sınırdaki $|YÇF|$ yine de eşikten büyükse d^i sıfırlanır. Değişim ölçüsü ise (83) bağıntısı kullanılarak hesaplanır. Burada d^i , yerdeğişim kestirimi için bir anlık değer olup, bir önceki konumdaki görüntü elemanının son kestirilen değeri veya üzerinde çalışılan görüntü elemanının en son kestirim değerini gösterir. Eğer her görüntü elemanı üzerinde sadece bir tekrar yapılırsa d^i 'nin bir önceki görüntü elemanına ait yerdeğişim vektörü olacağı kolayca görülebilir. Yeni bir d^i hesaplandığında eski $d^i = d^{i-1}$ olur. Yerdeğişim vektörünün boyu genellikle sınırlandırılır ve bu değer aşıldığında yerdeğişim vektörü sıfırlanır.

Her görüntü elemanı için alıcıya bir YÇF gönderilir. $|YÇF|$ 'nin eşik seviyesinden küçük olması durumunda ise ortaya çıkan sıfırların sayısı kodlanarak alıcıya gönderilir.

k katsayısı (79) eşitliğindeki gibi alınır, tipik video telefon görüntüleri için bir veya iki tekrar işlemi yeterli sayılabilmektedir [54]. Ancak birkaç tekrar için bile işlem sayısının görüntü elemanı başına oldukça fazla olacağı açıktır. Kestirim hatasını fazla arttırmayan bazı yöntemlerle işlem sayısı azaltılabilmektedir. Örneğin, [55]'de 9 çarpıcı ve 48 toplayıcı kullanılırken [53]'de 6 çarpıcı ve 11 toplayıcı kullanılmaktadır. $176 \times 144 \times 10$ (10: saniyedeki görüntü sayısı) değerlerine sahip bir video telefon görüntüsündeki görüntü elemanlarının ortalama %30'unun hareketli olduğu düşünülürse saniyedeki çarpma sayısı ortalama 450000 olmaktadır.



Şekil 17. Bir konum özyineli hareket kestiricinin ilkesel şeması.

1.3.3. Fourier Dönüşümüne Dayalı Hareket Kestirme Teknikleri

Her görüntü için iki boyutlu Fourier dönüşümü hesaplanır ve dönüşümün fazından yararlanarak ardışıl çerçeveler arasındaki yerdeğiştirme kestirilir [56], [57], [58], [59], [60], [61], [62]. I_1 ve I_2 , ardışıl iki görüntünün aynı bölgesine rastlayan iki boyutlu görüntü elemanı (piksel) parlaklık matrisleri olsun. Bir görüntü dönemi süresinde \mathbf{d} : (d_x, d_y) kadar bir öteleme hareketi olduğu varsayalım. Kenar etkileri önemsenmezse, iki bölge için aşağıdaki eşitlik yazılabilir[56]:

$$I_2(x, y) = I_1(x - d_x, y - d_y) \quad (84)$$

bu eşitliğin her iki yanının Fourier dönüşümü alınır, aşağıdaki eşitlik elde edilir:

$$F_2(m, n) = F_1(m, n)e^{-\pi j(md_x + nd_y)} \quad (85)$$

burada F_1 , I_1 'nin Fourier dönüşümü, (m,n) : uzamsal zamansal frekansları gösteren bir kolon vektördür.

Yazılıştta kolaylık olması için m,n indisleri ihmal edilirse, görüntünün çapraz ilişki işlevinin (circular) Fourier dönüşümü,

$$f(c) = F_1 \cdot F_2 = F_1 \cdot F_1^* \cdot e^{2\pi j(md_x + nd_y)} \quad (86)$$

olarak gösterilebilir. Eğer bu ifade ters dönüşüm almadan önce $F_1 \cdot F_1^*$ 'e bölünürse, gerekli olan yararlı sonuç elde edilmiş olur:

$$C(x, y) = \delta(x - d_x, y - d_y) \quad (87)$$

yani ilişki işlevi, yeri istenen yerdeğiştirmeyi belirten bir birim vuruş işlevi haline gelmiştir. Daha genel olarak, eğer I_2 I_1 'in sadece ötelenmiş değilse ve parlaklığı da değişiyorsa, aşağıdaki denklem hesaplanır:

$$C(Z) = f^{-1} \left\{ \frac{F_1 \cdot F_2^*}{|F_1 \cdot F_2^*|} \right\} \quad (88)$$

burada f^{-1} ters Fourier dönüşümünü gösterir. Burada yapılan, çapraz ilişki hesaplama işleminden önce I_1 ve I_2 'nin tayflarının normalize edilmesidir. Böyle bir ölçekleme işlemi, dönüşümden faz bilgisini ayıklamak için yararlıdır. Bu tip ilişki faz ilişkisi (phase correlation) olarak adlandırılır.

Faz ilişkisi tekniğinin çapraz ilişki tekniğine nazaran üstünlükleri şunlardır:

1) Tepelerin keskin olması nedeniyle bir çok farklı hareketten kaynaklanan çok sayıda tepeyi ayırtetmek mümkündür.

2) Aradeğerleme yaparak bir görüntü elemanı boyundan daha küçük yerdeğiştirmeler de bulunabilir.

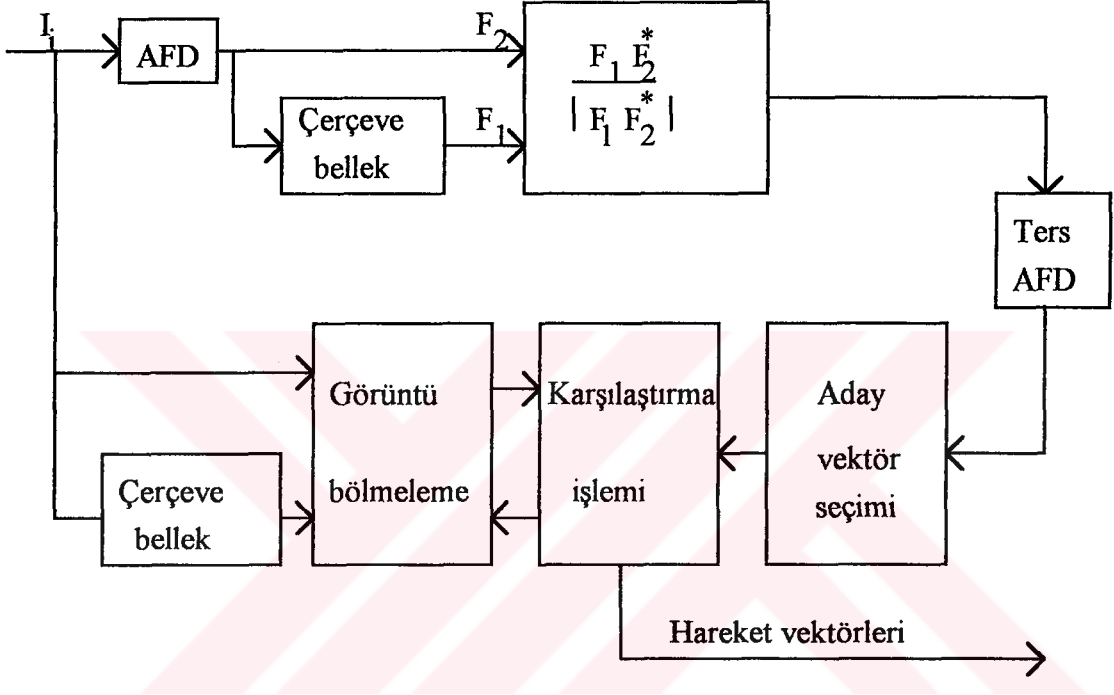
3) Görüntüde dar band genişliğine sahip olan parlaklık değişimleri (tüm görüntüde aydınlık değişmesi, büyük yayılmış gölgelerin hareketleri) ölçmelerin doğruluğunu etkilemez.

4) Hesaplamalarda FFT kullanmak, özellikle büyük yerdeğiştirmeler için çapraz ilişki hesaplamasından daha az çarpma gerektirir.

Bir faz ilişkisi işlevi [57]' de tanımlanmaktadır. Faz ilişkisi tekniği için, girişe gelen her görüntü bloklara bölünür. Faz ilişkisi işlevleri (veya yüzeyleri), baskın tepeleri bulmak için incelenir. Görüntünün her bölgesi için bir dizi aday yerdeğiştirme seçilir. Bu

yerdeřtirmelere baęlı olarak řimdiki ve bir önceki görüntüler arasındaki fark sinyalinin genlięi hesaplanır ve küçük bir alan üzerinden toplanır. Bu işlem den, en küçük hatayı veren vektör seçilir ve deęerlendirilir. Bu teknięin bir sakıncası, dönmeler ve hızlı ölçeklemeler (fast zooms) bir çok aday vektör ortaya çıkarır ve bu durumda iliřki yüzeyindeki tepeler birbirinden ayırtedilemeyebilir.

řekil 18'de faz iliřkisi teknięi kullanarak hareket kestirimi ilkesel olarak gösterilmektedir.



řekil 18. Faz iliřkisi teknięi ile hareket kestirimi.

1.3.4. Blok Karşılařtırmaya Dayalı Hareket Kestirme Teknikleri

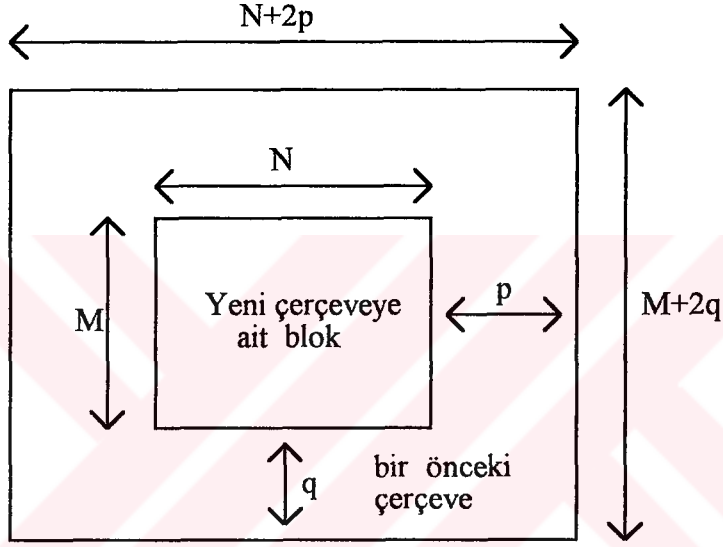
1.3.4.1. Sabit Boyutlu Blok Karşılařtırma

Blok karşılařtırma tekniklerinde, bir referans görüntü çerçevesi ile üzerinde bulunan görüntü çerçevesi (current frame) arasında bir $D(p,q)$ bozulma işlevinin (distortion function) minimum yapılmasıyla bulunabilen bir $d=[p,q]^T$ yerdeęiřtirme vektörü (displacement vector) aranır [63], [64], [65], [66], [67], [68], [69], [70], [71]. Buradaki p ve q , yatay ve düşey yöndeki yerdeęiřtirmeleri gösteren tamsayılardır.

Görüntü çerçeveleri, cisimlerin görüntü boyutlarından daha küçük boyutlu dikdörtgen bloklara bölünürse, cisimlerin hareketleri bu görüntü bloklarının yerdeęiřtirmelerinden bulunabilir. cisimlerin görüntü boyutlarının dikdörtgen blokların

boyutlarına oranı ne kadar büyük olursa, bu düşünce o kadar geçerli olur. Ardışıl görüntü çerçevelerindeki parlaklık değerleri blok blok karşılaştırılarak yerdeğiştirme miktarı arandığı için bu işlemlere "blok karşılaştırma" adı verilir.

Blok karşılaştırmada, bir görüntü çerçevesindeki herhangi bir $M \times N$ boyutlu parlaklık blokunun bir önceki görüntü çerçevesinin aynı merkezli $(M+2q, N+2p)$ boyutlu bir bölgesindeki yeri aranır. Burada M , bir bloktaki satır sayısı, N ise bir satırdaki görüntü elemanı sayısıdır. $2q+1$, bir blokun düşey yönde (eleman sayısı olarak) maksimum araştırma alanı genişliği, $2p+1$ ise yatay yöndeki maksimum araştırma alanı genişliği olarak tanımlanabilir



Şekil 19. Blok karşılaştırma.

Yerdeğiştirmenin bulunmasında, genellikle çapraz ilişki (cross-correlation) işlevi veya ortalama kare hata: OKH (mean square error) ve ortalama mutlak hata: OMH (mean absolute error) gibi bozulma işlevleri (distortion functions) karşılaştırma ölçütü olarak kullanılırlar. Bir ortalama bozulma işlevi (mean distortion function) aşağıdaki gibi gösterilebilir[63]:

$$D(i, j) = \frac{1}{MN} \sum_{m=1}^M \sum_{n=1}^N f[|I_{k+1}(m, n) - I_k(m-i, n-j)|] \quad (89)$$

Burada I_{k+1} bir görüntü çerçevesindeki (current frame) $M \times N$ boyutlu parlaklık blokunu, I_k ise bir önceki görüntü çerçevesindeki I_{k+1} ile aynı merkezli $(M+2q, N+2p)$ boyutlu parlaklık blokunu (referans blok) göstermektedir. $-q \leq i \leq +q$ ve $-p \leq j \leq +p$ ve $f(x)$ pozitif monoton artan bir işlevdir. Örneğin $f(x)=x^2$ OKH ölçütü için, $f(x)=|x|$ OMH

ölçütü için kullanılır. OKH veya OMH ölçütüyle $D(i,j)$ ' yi minimum yapan (i,j) değerlerinden minimum bozulmanın yönü bulunur.

Yerdeğiştirme değeri, bir çapraz ilişki işlevi ile maksimum ilişkinin yönünü bularak kestirilebilir. Bir çapraz ilişki işlevi $C(i,j)$ aşağıdaki gibi gösterilebilir[63]:

$$C(i,j) = \frac{\sum_{m=1}^M \sum_{n=1}^N I_{k+1}(m,n) \cdot I_k(m-i, n-j)}{\left[\sum_{m=1}^M \sum_{n=1}^N I_{k+1}^2(m,n) \right]^2 \cdot \left[\sum_{m=1}^M \sum_{n=1}^N I_k^2(m-i, n-j) \right]^{1/2}} \quad (90)$$

$D(i,j)$ ' yi veya $C(i,j)$ ' yi arama alanı içerisindeki her i,j değeri için hesaplamak (brute force search techniques) oldukça zaman alabilir. Çünkü bir blokun yerdeğişiminin bulunması için $(2p+1) \cdot (2q+1)$ kere hesaplama yapılması gerekmektedir. Bir görüntü çerçevesi için ise $(2p+1) \cdot (2q+1) \cdot H \cdot Y$ kere hesaplama yapılması gerekmektedir. Burada H , bir görüntü çerçevesinde yatay yöndeki blok sayısı, Y ise düşey yöndeki blok sayısıdır. Tablo 2' de bir blok için gereken araştırma sayısı verilmektedir. Minimum bozulmanın yönünü bulmada sıralı arama yapılırsa örneğin, $p=q=5$ için 121 yönün araştırılması gerekmektedir. Tablo 3' de ise CIF (352x288) ve QCIF (176x144) görüntü formatları için

Tablo 2. Arama uzaklığına göre bozulma işlevinin (veya ilişki işlevinin) hesaplanma sayısı

Arama uzaklığı	0	1	2	3	4	5	6	7	8	9	10	11	12
İşlem sayısı	1	9	25	49	81	121	169	225	289	361	441	529	625

Tablo 3. Blok boyutuna göre hareket vektörü sayısı (CIF ve QCIF boyutlu bir görüntü için)

Blok boyutları	4x4	8x8	16x16	32x32
Hareket vektörü sayısı (352x288)	6336	1584	396	99
Hareket vektörü sayısı (176x144)	1584	396	99	kesirli

değişik blok boyutları kullanılması durumunda ortaya çıkan hareket vektörü sayısı gösterilmektedir. Blok boyutlarının iki kat küçülmesi hareket vektörü sayısını dört kat arttırdığı görülmektedir. QCIF formatındaki bir görüntü 32x32'lik bloklara tam olarak bölünemediği için hareket vektörü sayısı tabloya yazılmamıştır.

Sıralı arama tekniğinde karşılaştırma sayısının çok fazla olması hesapsal bakımından güçlükler ortaya çıkarabilir. Bu nedenle minimum bozulmanın yönünü kısa zamanda ve az miktarda işlem yaparak bulabilmek için hızlı arama teknikleri kullanılabilir. Bunlardan önemlileri aşağıda anlatılmaktadır:

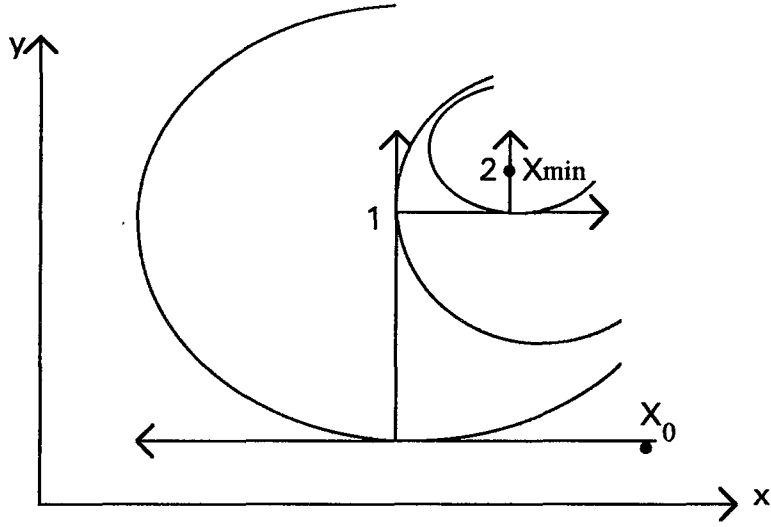
1.3.4.2. Hızlı Blok Karşılaştırma Yöntemleri

Hızlı blok karşılaştırma yöntemlerinin uygulanabilmesi için şu varsayımlardan yararlanılır: İlişki işlevinin tek bir tepe değeri vardır. Bu tepe değeri maksimum ilişki yönündedir. Arama alanındaki tepe noktasından uzaklaştıkça bu ilişki monoton olarak azalır.

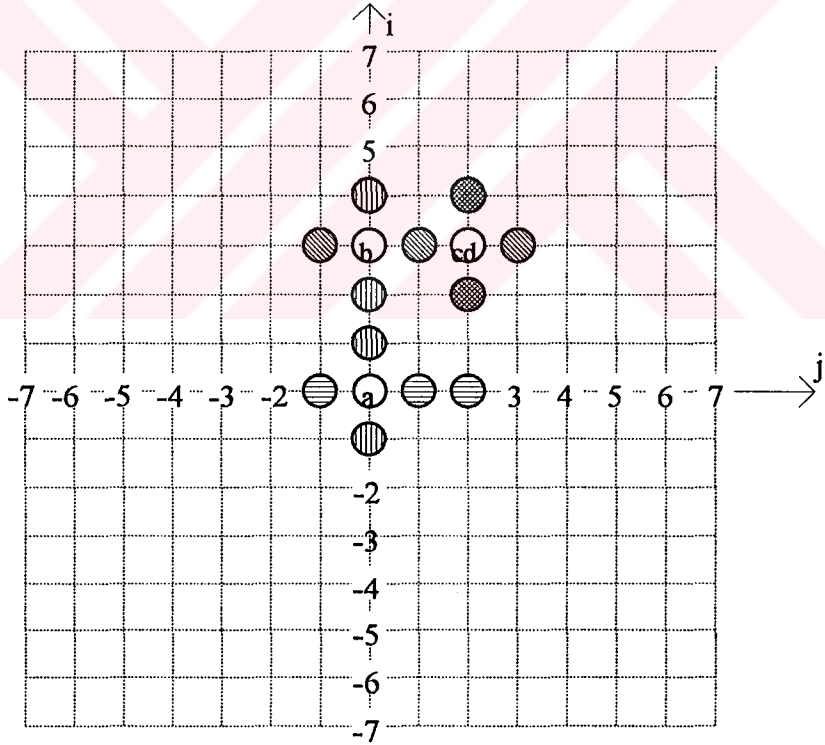
İlişki işlevi video sinyalinin içeriğine büyük ölçüde bağımlıdır. Bu nedenle yüksek ayrıntı ve hareket içeriği bulunan karmaşık görüntülerde (dar şeritler, ızgara şekilleri) arama doğrultusu bazen yanlış yöne kayarak yanlış tepeye varılabilir. Bu durum, hareketlerin yanlış kestirilmesine neden olur. Bu nedenle karşılaştırma ölçütünün ve karşılaştırma şeklinin seçimi önemlidir. Karşılaştırma ölçütü olarak OMH ve OKH genellikle birbirine yakın performans gösterirler. Blok karşılaştırma işleminde, amaç işlevin en iyi (optimal) çözümünü elde etmek için önce kestirilen bir ilk değerden başlanır. Amaç işlevin değerinin birtakım testler uygulanarak indirgenmesi ve en iyi çözümün bulunması bir optimizasyon problemidir. En iyi çözüm için kullanılacak yöntemler aşağıda anlatılmaktadır. Hızlı arama tekniklerinin uygulanmasında, doğru yerdegiştirmeye yakınsayabilmek için bozulma işlevinin dışbükey (convex) olması gerekmektedir. Örneğin, bir ortalama kare hata işlevi (mean square error) veya ortalama mutlak hata (mean absolute error) bu şartı sağlar. İki yöntem de birbirine yakın performans göstermektedir. Yöntemler iki boyutlu sayısal görüntü verilerine uygulandığında iki görüntü çerçevesi arasındaki parlaklık farklarını içeren iki değişkenli bozulma işlevleri elde edilir ve bu işlevler aşağıdaki tekniklerden biriyle incelenirler.

Bir-seferde-bir arama (BSBA) yöntemi (one-at-a-time search) [72]: Minimum noktanın üzerinde bulunduğu varsayılan yön üzerinde bir minimum aramakla işleme başlanır. Her adımda, minimum bozulmayı bulmak amacıyla iki değişkenden biri sabit tutulur, diğeri değiştirilir. Her adımdaki araştırma yönü koordinat eksenlerinden birine paralel olarak seçilir. Bir adımda bulunan minimum gerçek yönü belirtmediği için ardışıl olarak işlemler tekrarlanır. BSBA yönteminin iki değişkenli bir dışbükey işleve uygulanışı Şekil 20' de gösterilmektedir. Şekil 21 'de ise iki boyutlu görüntü verilerini içeren bir

bölge için BSBA yöntemiyle en iyi çözüme doğru gitmek için gereken adımlar gösterilmektedir.



Şekil 20. BSBA yönteminin iki değişkenli bir dışbükey işleve uygulanışı.



- ⊙ : Birinci adımdaki arama noktaları , ⊙ : İkinci adımdaki arama noktaları
- ⊙ : Üçüncü adımdaki arama noktaları , ⊙ : Dördüncü adımdaki arama noktaları
- ⊙ : Birinci adımdaki minimum noktası , ⊙ : İkinci adımdaki minimum noktası
- ⊙ : Üçüncü adımdaki minimum noktası , ⊙ : Dördüncü adımdaki minimum noktası

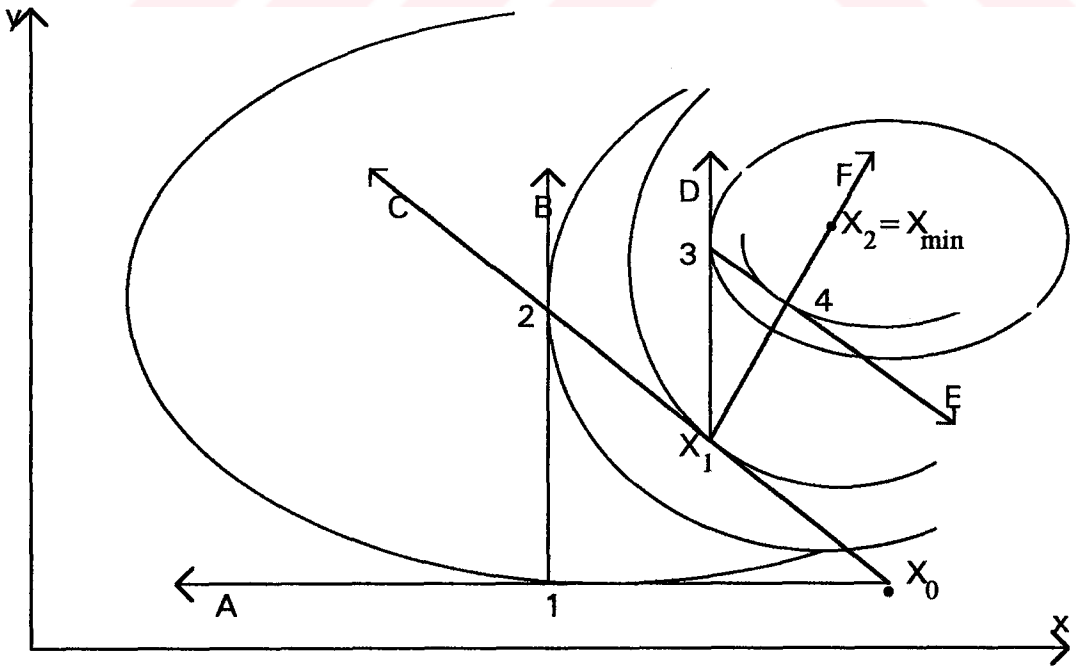
Şekil 21. Bir seferde bir arama yöntemiyle blok karşılaştırma.

Herhangi bir adımda bir minimum bulunduktan sonra, gerekli ölçütlerle karşılaştırma yapılır ve istenen ölçütleri sağlayan minimum elde edilinceye kadar araştırmaya devam edilir. En son bulunan noktada yatay ve düşey yöndeki minimumlar aynı olur. Ortalama 7 arama yapılmaktadır. Bozulma fonksiyonu D, sürekli kısmi türevlere sahipse BSBA yöntemi genellikle yakınsar.

Eşlenik yönlerde arama (EYA) yöntemi (conjugate direction search)[72]:

Eşlenik yönlerde aramada araştırmanın yönü sabit tutulmayıp, araştırmanın sonucunu iletilecek şekilde devamlı olarak değiştirilir. Amaç işlevindeki değişkenlerin sayısı kadar doğrusal bağımsız eşlenik yön vektörü bulunmaktadır. Örneğin, iki değişkenli oval (elliptic) bir D amaç işlevi için sadece iki adımda sonuca varılmaktadır. EYA yönteminin eliptik çevreli konveks bir işleve uygulanışı Şekil 22' de gösterilmektedir.

Araştırmaya önce eksnelere paralel yönde başlanır. x eksenine paralel yönde bulunan minimum noktasından başlayarak y eksenini boyunca ilerlenir. y eksenine paralel yöndeki minimum noktası (2) ile X_0 başlangıç noktası birleştirilir (c doğrusu). Bu doğru üzerindeki (X_1) minimum noktasından başlayarak y eksenine paralel yönde araştırmaya devam edilir. Bu adımda x yönünden vazgeçilir ve y eksenine paralel yönde bulunan bir minimum noktasından itibaren c doğrusuna paralel yönde araştırmaya devam edilerek bir minimum noktası (4) bulunur. Yeni araştırma yönü, X_1 noktası ile 4 noktasını birleştiren doğru üzerindedir. Burada elde edilen minimum noktası (X_2) varılması gereken noktadır.

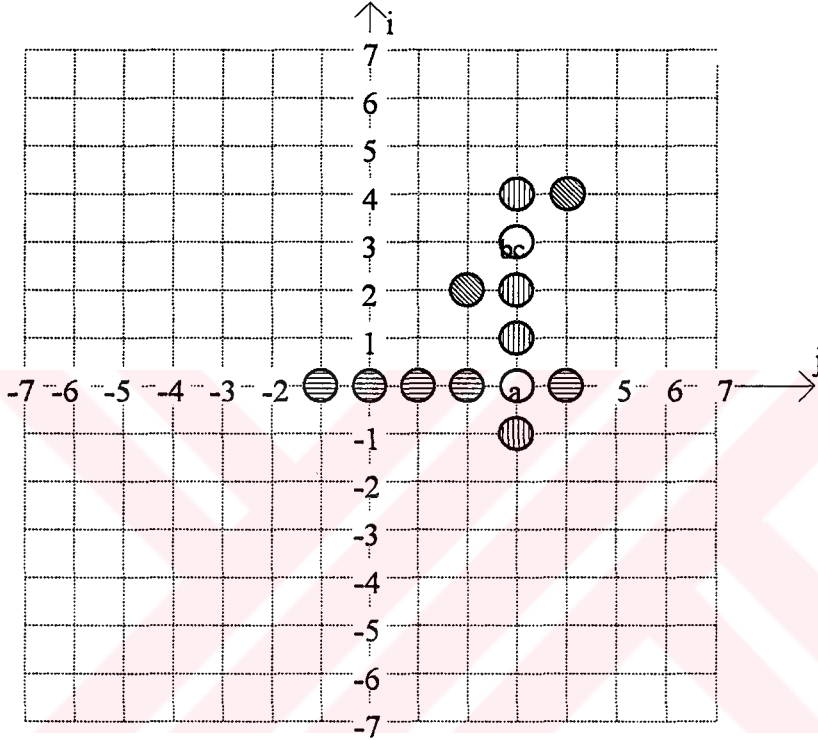


Şekil 22. EYA yönteminin iki değişkenli bir dışbükey işleve uygulanışı.

EYA yönteminin blok karşılaştırma işleminde kullanılması aşağıda anlatılmaktadır:

1.adım: $D(i,j)$, $D(i,j+1)$, $D(i,j-1)$ hesaplanarak aralarından minimum olan seçilir (Şekil 23). Eğer minimum $D(i,j+1)$ de ise, $D(i,j+2)$ noktası da hesaplanır ve $D(i,j)$, $D(i,j+1)$, $D(i,j+2)$ den minimum olan bulunur. Bu işlemler ardışıl olarak sürdürülerek i eksenini yönündeki minimum bulunur.

2. adım: Birinci adıma benzer olarak araştırma i eksenini yönünde sürdürülür.



- : Birinci adımdaki arama noktaları, ● : İkinci adımdaki arama noktaları
 ● : Üçüncü adımdaki arama noktaları ● : Birinci adımdaki minimum noktası
 ● : İkinci adımdaki minimum noktası, ● : Üçüncü adımdaki minimum noktası

Şekil 23.EYA yöntemiyle blok karşılaştırma.

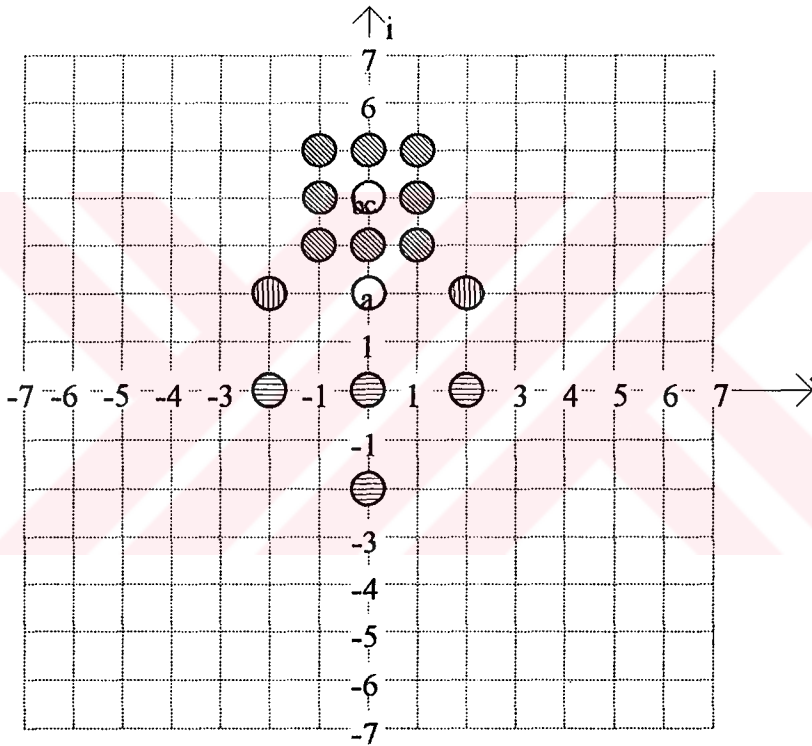
3.Adım: Araştırmanın yönü, (i,j) başlangıç noktasını en son bulunan minimum noktaya birleştiren doğru üzerindedir. Eğer 2. adımda bulunan minimum, $D(i+3,j+3)$ de ise, $D(i+2,j+2)$ ve $D(i+4,j+4)$ noktaları hesaplanır. 1. ve 2. adımlarda olduğu gibi bir minimuma yakınsanana kadar işlemlere devam edilir. 1. ve 2. adımlarda elde edilen i ve j vektörleri bir kare oluşturmadığı zaman, örneğin, 2. adımda $(i+2,j+3)$ noktasında bir minimum bulunduğunda, (i,j) ve $(i+2, j+3)$ noktalarını birleştiren doğru üzerinde yanından geçilen en yakın noktalar araştırılır. Bunlar, Şekil 23'de $(i+1,j+2)$ ve $(i+3,j+4)$ noktalarıdır. Burada, araştırma yönünde küçük sapmalar olmaktadır. Fakat gerçek minimum noktasından sapma, bir görüntü elemanı genişliğinden daha azdır ve bu

hata önemsenmeyebilir. Araştırmaya aynı yön üzerinde 1. ve 2. adımlarda olduğu gibi bir minimum buluncaya kadar devam edilir.

Yukardaki araştırmada BSBA yöntemi kullanılıyorsa, ilk iki adımda işlemler sona erecekti ve bulunan en uygun noktalar da daha farklı olacaktı.

Genellikle EYA, BSBA' dan daha hızlı yakınsama gösterir. Eğer D karesel bir işlevse (quadratic) ve oval kenarlara sahipse, yakınsama hızı daha da artar. $p=q=5$ için en fazla 13 arama yapılır. Genellikle BSBA den daha hızlı yakınsama gösterir. Arama sayısı $= 3+2(p=q)$.

Logaritmik arama [63]: Bozulmanın yönünü araştırma işlemi, araştırma alanın ardışıl olarak her adımda küçültülmesi prensibine dayanır (Şekil 24).



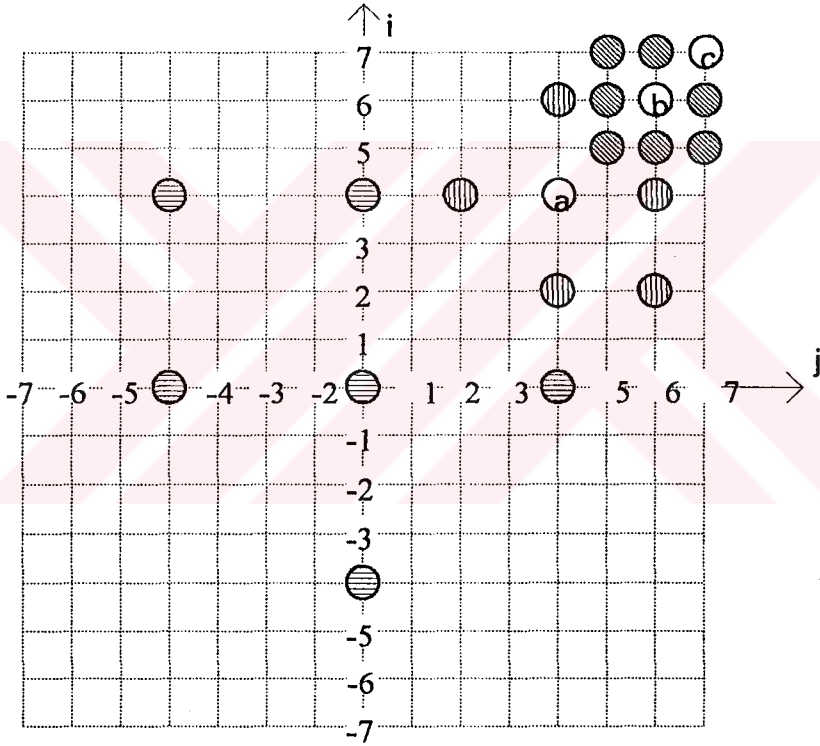
- ⊙ : Birinci adımdaki arama noktaları, ⊚ : İkinci adımdaki arama noktaları
 ⊛ : Üçüncü adımdaki arama noktaları ⊙ : Birinci adımdaki minimum noktası
 ⊖ : İkinci adımdaki minimum noktası, ⊕ : Üçüncü adımdaki minimum noktası

Şekil 24. Logaritmik arama yöntemiyle blok karşılaştırma.

İlk adımda araştırma alanındaki merkez nokta ve merkezden geçen eksenler üzerinde, herbiri merkezden adım büyüklüğü kadar uzaklıkta olan dört nokta için bozulma işlevi ayrı ayrı hesaplanır (ilk adım büyüklüğü $(p=q)/2$). İkinci adımda minimum değere sahip nokta etrafında, birinci adımda bulunan minimum bozulma yönünde üç yeni nokta daha araştırılır ve minimum bozulmayı sağlayan bir aday nokta bulunur. Yeni araştırma alanının merkezi olarak bu aday nokta seçilir. Araştırma adım büyüklüğü yarıya

indirilir ve ilk iki adımda yapılan işlemler, adım büyüklüğü 1'e inene kadar tekrarlanır. Her adımda araştırılan alanın boyutları logaritmik olarak azalır. Her blok için hesaplanan bozulma işlevi sayısı da maksimum yerdeğiştirme (p) 'nin artması ile logaritmik olarak artar. En son adımda (adım büyüklüğü 1 iken) araştırma yapılan alanın boyutları 3x3 'e iner ve 9 komşu nokta için minimum bozulma işlevi hesaplanır. Minimize karşılık gelen nokta minimum bozulmanın yönünü (ve yerini) verir. Burada ise en az 13, en fazla 21 yönün (noktanın) araştırılması yeterli olmaktadır.

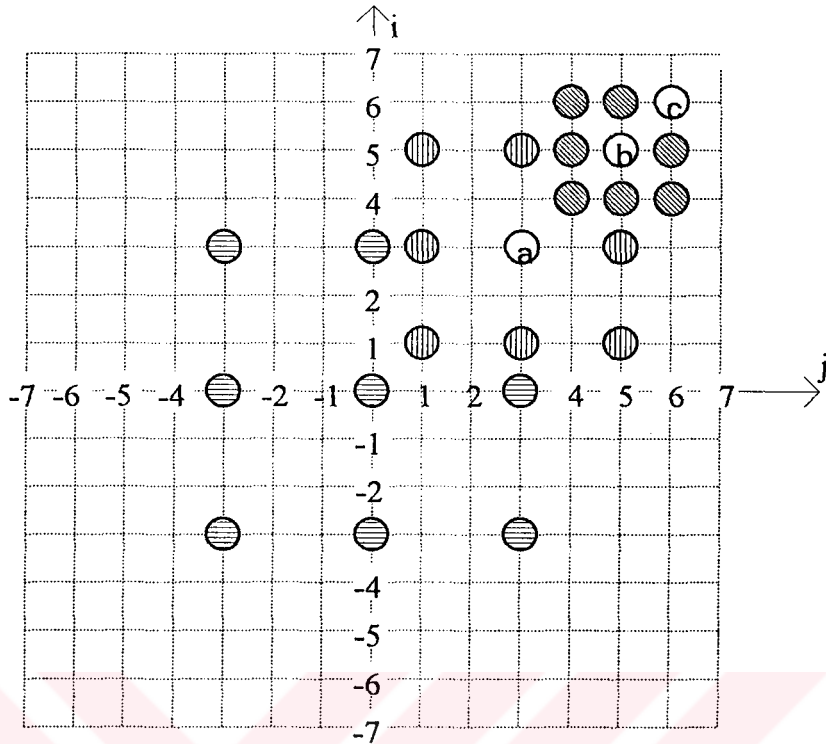
Düzeltilmiş Logaritmik Arama [72]: Logaritmik aramadan farkı, adım büyüklüğünü yarıya indirmeden önce, Üç yerine iki nokta araştırılır (Şekil 25). 1, 4+2, 4+2, 4+2 olmak üzere toplam 1 ile 19 noktanın araştırılması gerekir. Arama sayısı $=1+6\log_2(p=q)$ olarak hesaplanır.



- ⊖ : Birinci adımdaki arama noktaları, ⊕ : İkinci adımdaki arama noktaları
- ⊗ : Üçüncü adımdaki arama noktaları ⊙ : Birinci adımdaki minimum noktası
- ⓐ : İkinci adımdaki minimum noktası, ⓑ : Üçüncü adımdaki minimum noktası

Şekil 25. Düzeltilmiş logaritmik arama yöntemiyle blok karşılaştırma.

Üç Adımlı Yönlü Arama (Three-Step Directed Search) [72], [73]: 6 piksele kadar olan yerdeğiştirmeler hesaplanabilir. Önce koordinat merkezi etrafında $(p=q)/2$ aralıklı (adım büyüklüğü $= (p=q)/2$) 8 nokta araştırılır (Şekil 26). Bulunan minimum nokta merkez seçilip adım büyüklüğü bir azaltılır ve 8 yeni nokta araştırılır. Üç adımda toplam 25 nokta araştırılmış olur.



- : Birinci adımdaki arama noktaları, ○ : İkinci adımdaki arama noktaları
 ○ : Üçüncü adımdaki arama noktaları, a : Birinci adımdaki minimum noktası
 b : İkinci adımdaki minimum noktası, c : Üçüncü adımdaki minimum noktası

Şekil 26. Üç adımlı yönlü arama yöntemiyle blok karşılaştırma.

Çapraz arama (Cross search) [74]: Temel düşünce yine logaritmik araştırmadır. Ancak her adımda eksenler üzerindeki 4 nokta yerine eksenlere çapraz konumda olan 4 nokta araştırılır. En son adımda, araştırılan noktalar birbirlerine göre çapraz konumda (St. Andrew's cross) veya eksenler üzerinde (Greek cross) olabilir. Örneğin 8 görüntü elemanı kadar olan yerdeğişirmeler için 17 arama yapılır.

Dikgen arama (Orthogonal search)[74], [75]: Adım büyüklüğü yine logaritmiktir ve her iterasyonda 4 nokta araştırılır. Her adımda sırasıyla düşey ve yatay yönde iki konum araştırılır.

Bu yöntemlerin herbiri değişik yerdeğişirmeler üzerine kurulmuştur (Örneğin, $p=5,6,7,8$ gibi). Bu nedenle işlem sayısı bakımından verimlerini bir blok için doğrudan karşılaştırmak zordur. Bunun yerine, görüntü çerçevesinin boyutlarına göre yaklaşık hesap sayısı bulunabilir. Dikgen aramada ortalama arama sayısı en az olmasına rağmen performans bakımından üç adımlı yönlü arama, düzeltilmiş logaritmik arama ve eşlenik yönlü arama en iyi sonucu vermektedirler [74].

2. YAPILAN ÇALIŞMALAR

2.1. Giriş

Bloksal hareket kestirme tekniklerinin bir zayıf tarafı, yer değiştirmenin bir blok için sabit kabul edilmesidir. Blok boyutları hareket eden cisimin görüntü boyutlarından çok küçük değilse, bir blok içerisinde birbirinden bağımsız hareket eden görüntü parçaları olacağından, hareketler iyi kestirilemez. Bu problemi çözenin bir yolu, çok küçük blok boyutları kullanmaktır. Fakat bu durumda, hareket vektörleri için gereken bit miktarı artacağı için toplam veri hızı azaltılamaz. Görüntünün karmaşık hareketler içeren kısımlarında blok boyutlarının küçük, az hareket içeren kısımlarında ise blok boyutlarının büyük seçilmesiyle hem hareketler daha iyi kestirilebilir hem de toplam veri hızı daha fazla azaltılabilir.

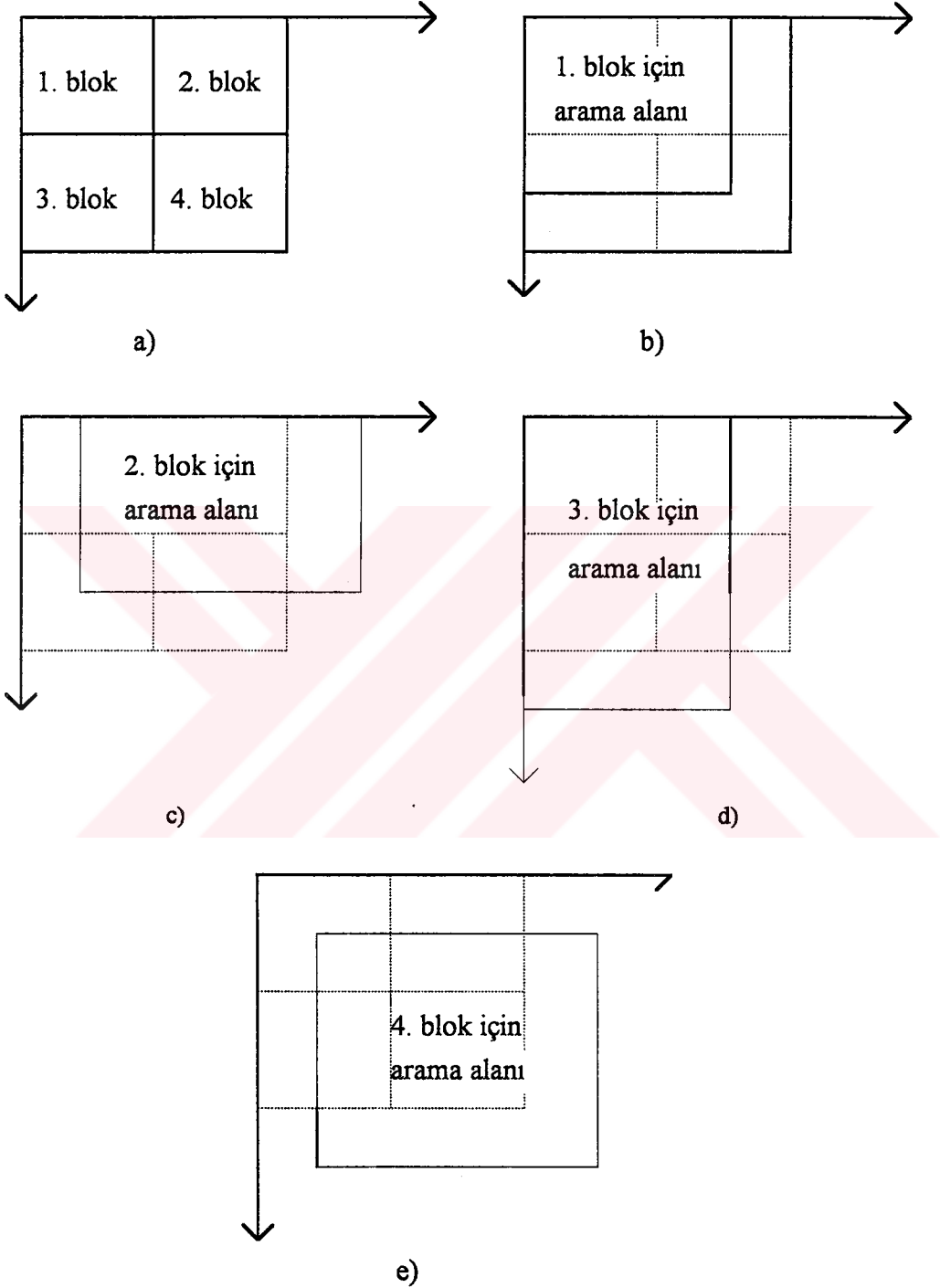
2.2. Bloksal Hareket Kestirime Dayalı Kodlama İçin Yeni bir Yöntem

Bu çalışmada hareket içeriğine göre uyarlanır bir yöntem önerilmekte ve bu yöntem aşağıda adım adım anlatılmaktadır:

1. Adım (Hareket Vektörlerinin Belirlenmesi): Görüntü önce 16x16 piksel (görüntü elemanı) boyutlu sabit boyutlu bloklara ayrılır. Her blok bir kez daha bölünerek 8x8 boyutlu alt bloklara ayrılır. Önce 8x8 boyutlu alt bloklara ait hareket vektörleri birbirlerinden bağımsız olarak bulunurlar. Daha sonra hareket vektörlerinin yatay ve düşey bileşenleri tesbit edilir. Hareket vektörleri bu küçük bloklar için aranır.

Şekil 27'de 16x16 boyutlu herhangi bir blok içerisindeki alt blokların bir önceki görüntü çerçevesinde arandıkları alanlar gösterilmektedir. Görüntünün kenarlarına rastlayan alt bloklar Şekil 27 a)' da görüldüğü gibi aranmaktadır. Örneğin görüntünün en sol üst köşesindeki alt blok, bir önceki görüntü çerçevesinde sadece sağa ve aşağı doğru olan yönlerde aranabilmektedir. Maksimum arama uzaklığı yatay ve düşey yönde ± 6 pikseldir. Diğer yönler çerçevenin dışına çıktığı için aranmamaktadırlar. b)' de arama alanı yatay yönde +6 piksel ve düşey yönde +6 pikseldir. c)'de arama alanı yatay yönde ± 6 piksel ve düşey yönde +6 pikseldir. d)'de arama alanı yatay yönde +6 piksel ve düşey yönde ± 6 pikseldir. e)'de ise her iki yönde de ± 6 piksellik arama yapılmaktadır. Görüntünün ortalarına rastlayan bölgelere ait bütün alt bloklar için arama bölgeleri Şekil

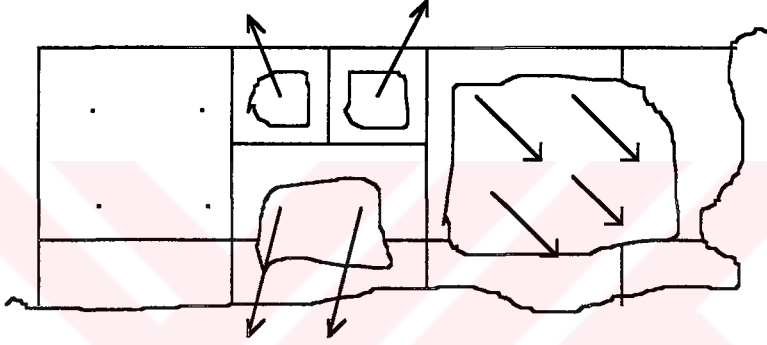
27 e)' deki gibidir. Burada arama alanı yatay ve düşey yönde ± 6 pikseldir. Ortaya çıkan hareket vektörünün boyu ise en fazla $\sqrt{2} \times 6$ piksel olabilmektedir.



Şekil 27. Görüntünün bir bölgesi için alt blokların bir önceki görüntüde arandıkları alanlar. a): Görüntünün sol üst köşesinde 4 adet alt bloka ayrılmış bir bölge. b)-d): sırasıyla 1., 2., 3. ve 4. alt blokun bir önceki görüntü çerçevesindeki arandıkları alanlar.

Blok karşılaştırmada hızlı arama yöntemlerinden üç adımlı yönlü arama kullanılmaktadır. Üç adımlı yönlü arama, sıralı arama yöntemine yakın performans göstermesi ve kolay uygulanabilir olması nedeniyle tercih edilmiştir.

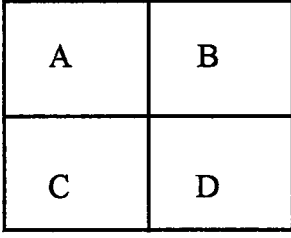
2. Adım (Alt Bölgelerin Bulunması): Bütün alt bloklar için hareket vektörleri belirlendikten sonra, her bir alt bloğa ait hareket vektörleri birbirleriyle karşılaştırılırlar. Hareket vektörleri eşit olan alt bloklar; aynı yönde hareket eden bir cismin parçaları olarak kabul edilebileceğinden dolayı, birleştirilirler. Bunun sonucunda, görüntünün içerdiği harekete göre şekillenen görüntü bölgeleri elde edilir. Örneğin, aynı hareket vektörlerine sahip alt blokların birleştirilmesi şekil 28'de gösterilmektedir.



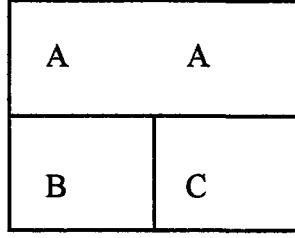
Şekil 28. Aynı hareket vektörüne sahip alt blokların birleştirilmesi.

Görüntünün bir bölgesindeki alt bloklara ait bazı hareket vektörlerinin eşit olması durumunda eşit hareket vektörlerinden sadece bir tanesinin alıcıya gönderilmesi yeterlidir. 16x16 boyutlu blok, düzgün öteleme hareketi yapan bir cismin görüntü boyutlarından daha küçük ise bu blok içerisindeki hareket vektörlerinin eşit olma olasılığı yüksektir. Bu kural bozulduğu ölçüde hareket vektörü sayısı artacaktır. Hiç hareket olmadığı durumda da yine tek bir hareket vektörü ortaya çıkacaktır. Şekil 28'den görüldüğü gibi, alt bloklardan elde edilen hareket vektörleri farklı olduğu sürece blok boyutları küçük tutulmaktadır. Diğer durumda blok boyutları büyümektedir.

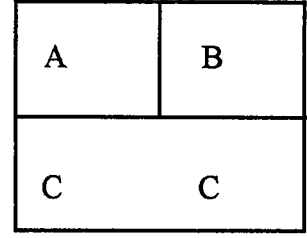
Şekil 29'da bir blokun dört alt bloka ayrılması sonucunda ortaya çıkabilecek bölge biçimleri gösterilmiştir. Bu işlem sonucunda ortaya çıkan hareket vektörleri, her 16x16 boyutlu blok içerisinde değişik bölge biçimlerini temsil etmektedirler. Dört adet alt blok kullanılması durumunda, 15 farklı bölge biçimi ortaya çıkmaktadır. Şekil 29'da birleştirilen bloklar aynı harflerle gösterilmişlerdir. Örneğin dört farklı harf dört farklı bölge oluştuğunu ifade etmektedir (Şekilde 1 numaralı blok). 2, 3, 4, 5, 6 ve 7 numaralı blokların herbirinde üçer tane bölge, 8, 9, 10, 11, 12, 13 ve 14 numaralı bloklarda ikişer tane bölge ve 15 numaralı blokta bir tane bölge ortaya çıkmaktadır.



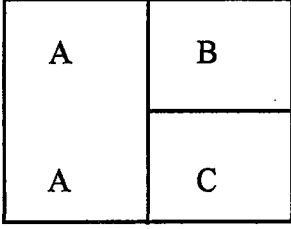
1



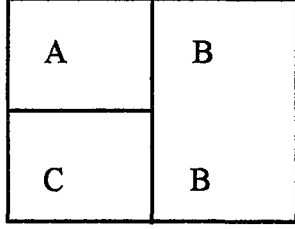
2



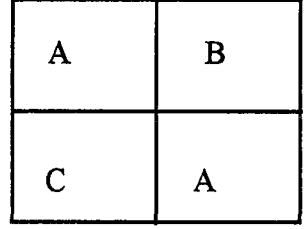
3



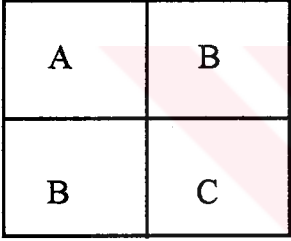
4



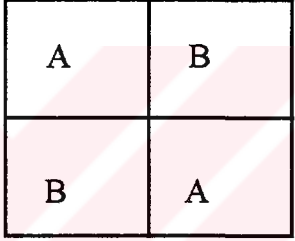
5



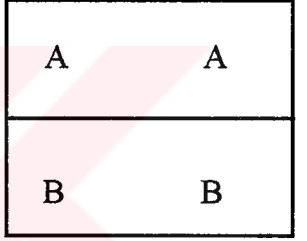
6



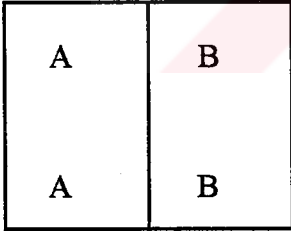
7



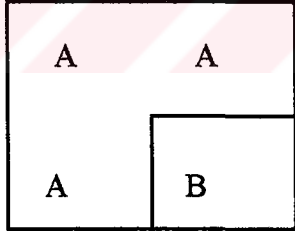
8



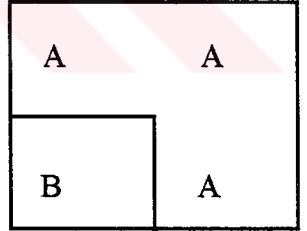
9



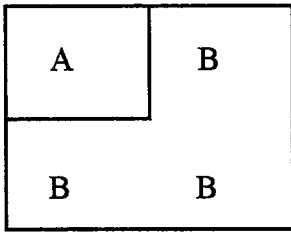
10



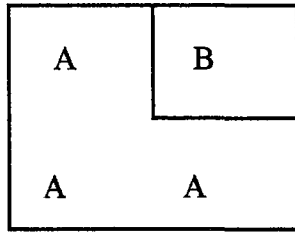
11



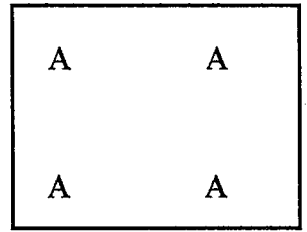
12



13



14



15

Şekil 29. Dört adet alt blok kullanılması durumunda ortaya çıkması mümkün olan bölge biçimleri.

Alt blokların birleştirilmeleri sonucunda ortaya çıkabilecek bölge biçimleri bir liste halinde önceden belirlenmektedirler. Alt blok sayısı arttığında ortaya çıkabilecek bölge biçimi sayısı üstel olarak artmaktadır. Ortaya çıkabilecek bölge biçimi sayısının alt blok sayısı ile değişimi şu ifadeyle hesaplanmaktadır:

$$\begin{aligned} \text{Farklı Bölge Sayısı} = & \text{alt bloklara ait hareket vektörleri arasındaki} \\ & \text{birer birer benzerlikler} + \text{ikişer ikişer benzerlikler} + \dots + \\ & \text{n'inci dereceden benzerlikler} + \text{bütün alt benzerlik grupları.} \end{aligned} \quad (91)$$

Burada n, alt blok sayısıdır. Örneğin n=4 için farklı bölge sayısı 15 olarak bulunmaktadır.

Tablo 4'te kullanılan alt blok sayısına göre ortaya çıkan farklı bölge sayısı verilmektedir. Tablodan görüldüğü gibi, alt blok sayısı arttığında bölge sayısı üstel olarak artmaktadır.

Tablo 4. Alt blok sayısına göre ortaya çıkan farklı bölge sayısı ve bölge başına düşen maksimum entropi

Alt Blok Sayısı	Farklı Bölge Sayısı	Maksimum Entropi (bit / bölge)
2	2	1
3	5	3
4	15	4
5	53	6
6	213	8

3. Adım (İletilecek Hareket Vektörlerinin Seçilmesi): Ortaya çıkabilecek bölge biçimlerini belirten bir liste düzenlenerek alıcıya hareket vektörleri ve hareket vektörlerinin bu listedeki adresleri gönderilmektedir. Eğer adres bilgisi istatistiksel kodlama yapılmazsa, bu işlem her 16x16 boyutlu görüntü bloku için fazladan 4 bitin alıcıya gönderilmesini gerektirmektedir. Tablo 4'te, bir bölgenin biçimini belirten bilginin iletilmesi için gereken maksimum entropi miktarları verilmektedir. Bölge biçimi bilgisinin istatistiksel kodlanması durumunda ortaya çıkan entropi tabloda verilenlerden daha az olmaktadır. Çünkü, video telefon görüntülerinde küçük boyutlu blokların hareketlerinin farklı yönde çıkma olasılığı azdır. Bu nedenle sık rastlanılan bölge biçimleri daha kısa kod sözcüğüyle kodlanırlar.

Bu yöntemin, televizyon sinyallerinin blok sal hareket kestirimine dayalı kodlanmasında toplam veri hızını sabit boyutlu bloklar kullanılması durumundan

daha fazla azaltacağı söylenebilir. Ancak videotelefon gibi kesin bir hız kısıtı altında çalışan sistemlerde de bundan yararlanmak mümkündür. Videotelefondaki kanal sığası 64 kbit/s ile sınırlı olduğu için, bit hızından yapılan bu tasarruf, AKD katsayılarını daha ince seviyelendirmekle kompanse edilebilir. Çünkü, iletimden hemen önce kullanılan yastık bellek daha az dolma eğiliminde olacaktır. Yastık belleği aynı dolulukta tutacak şekilde ince seviyelendirme yapılabilir. Eğer AKD katsayılarına uygulanan eşik düzeyi yastık belleğin doluluğuyla orantılı olarak uyarlamalı seçiliyorsa, yastık bellek daha az dolma eğiliminde olacağı için eşik düzeyleri daha düşük olacaktır. Eşik düzeylerinin düşük olması sonucunda da alıcıya daha fazla bilgi gönderilecek ve kodu çözülen görüntülerin sinyal/gürültü oranı iyileşecektir.

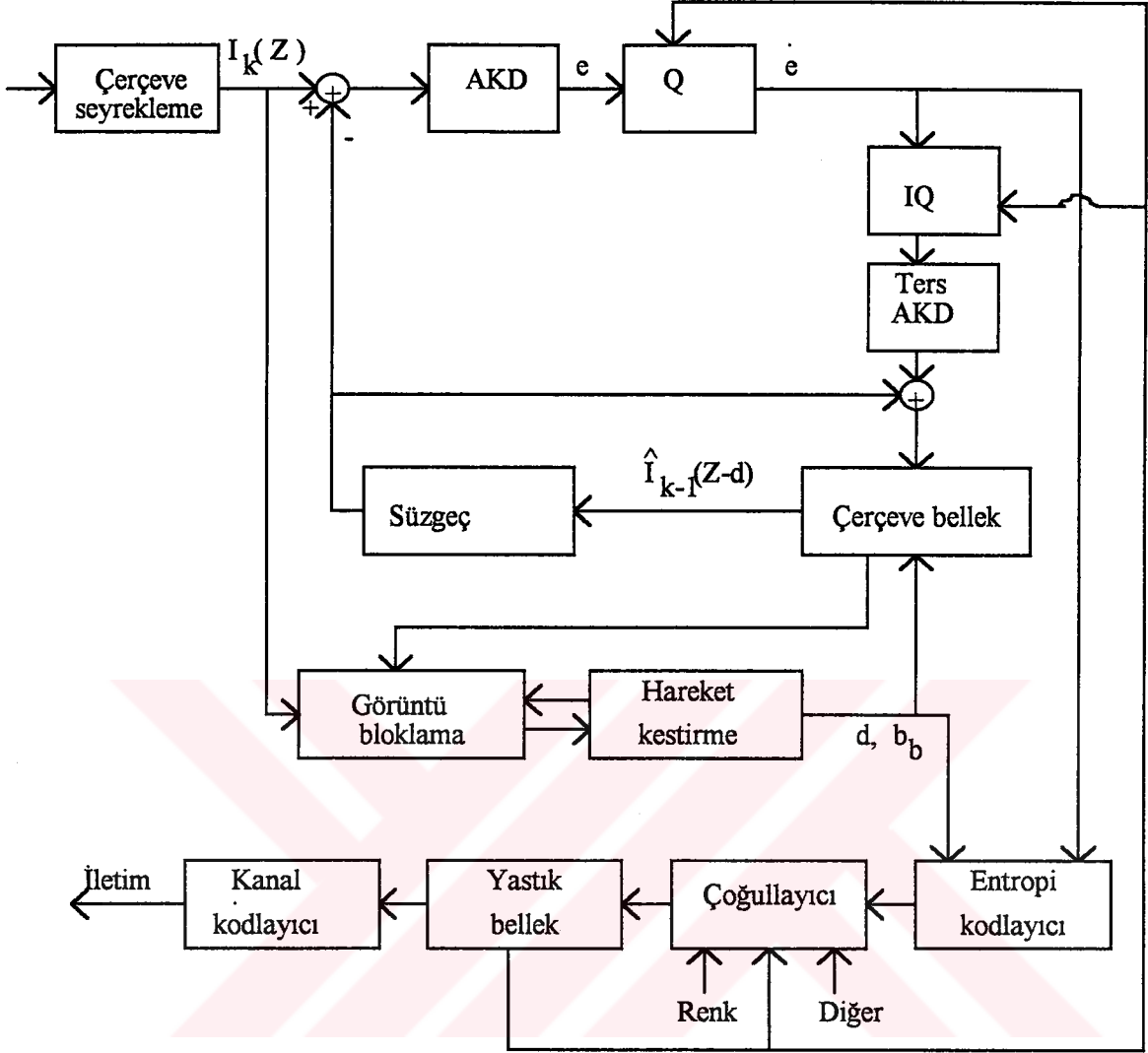
Önerilen yöntem, video telefon sinyallerinin kodlanmasında görüntü kalitesini arttırmak amacıyla incelenmiştir.

2.3. Önerilen Yöntemin Video Telefon Sinyallerinin Kodlanmasında Kullanılması

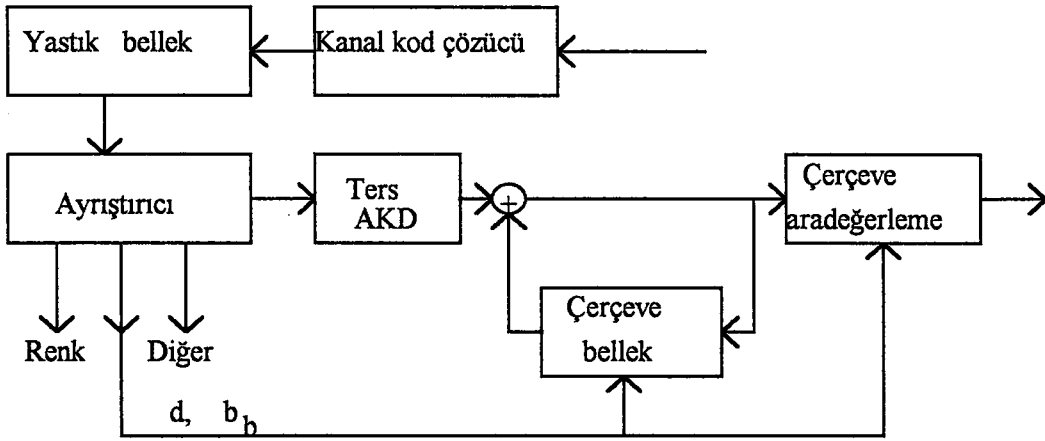
Video telefon sinyallerinin iletimi için kodlayıcı ve kod çözücü Şekil 30'daki gibi tasarlanmıştır. Bu tasarımda önerilen yöntem, vericide görüntü bloklama ve hareket kestirme kısmında kullanılmaktadır. Alıcıda ise yeni bir blok gösterilmemiştir. Çünkü alıcıda yapılan yenilik, sadece bellekten yapılan okuma sırasında olmaktadır. Alınan bölge biçimi adresine göre hareket vektörleri dört alt blok için yeniden düzenlenmekte ve bir önceki kodçözülmüş görüntü (bellekte tutulan) ile yeni alınan YÇF görüntüsü toplanmaktadır.

Burada, Şekil 12'den farklı olarak hareket kestirme işlemi kodçözülmüş veriler üzerinden yapılmaktadır. Bunun yararı hataların yığılmasını önlemesidir. Blok şemadaki birimlerin görevleri ve yapılan işlemler aşağıda anlatılmaktadır:

Şekil 30'da çerçeve seyrekleyici biriminde, hareket kestirimini daha az görüntü çerçevesi kullanarak yapabilmek için saniyedeki görüntü sayısı azaltılır. Bu birimdeki seyrekleme miktarı, sistemin kullanım alanına göre seçilir. Görüntü çerçeveleri seyreklediği zaman aralarındaki zamansal ilişki azalır. Televizyon görüntülerinde zamansal ilişki çok az olabilirken, insan başı ve omuzlarından oluşan ve yavaş hareketleri içeren videotelefon görüntülerinde zamansal ilişki çok fazla olabilir. Bu nedenle video telefon görüntü çerçeveleri genellikle seyrekleme yapmak için elverişlidirler. Tipik olarak saniyede 8-15 görüntü çerçevesi kullanılması yeterli olabilmektedir.



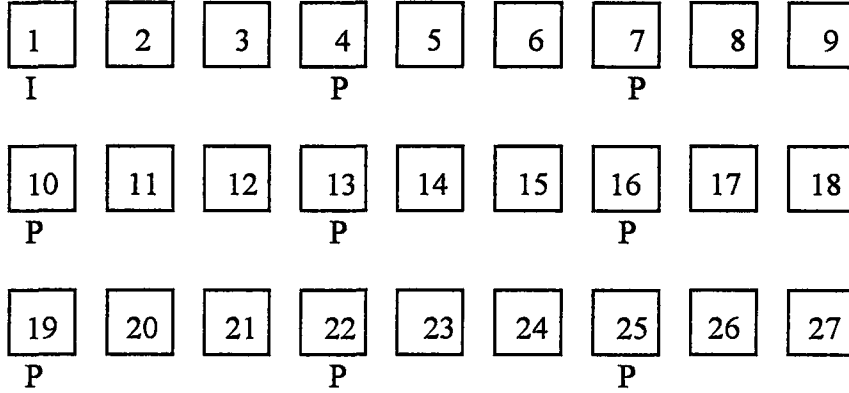
KODLAYICI



KOD ÇÖZÜCÜ

Şekil 30. Video telefon sinyalleri kodlayıcı ve kodçözücü blok şeması
 (AKD : Ayrık kosinüs dönüşümü, Q : Seviyelendirici, IQ : Ters seviyelendirici , d :Hareket vektörü bilgisi, b_b : bölge biçimi bilgisi).

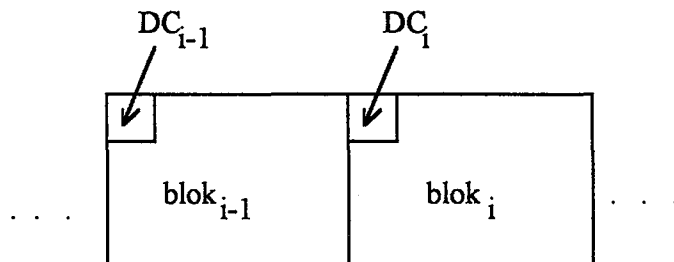
Video telefon görüntü dizisinin kodlanması Şekil 31' deki formata göre seçilmiştir. 25 görüntü çerçevesi 1 saniyeye karşılık gelmektedir. Dizideki 1. ve 28. görüntüler "I" tipi (çerçeve-içi) kodlanmaktadır. 4., 7., 10., ... görüntü çerçeveleri "P" tipi (çerçeveler-arası) kodlanmaktadır. Arada kalan 2.,3., 5.,6., ... görüntü çerçeveleri ise istenirse aradeğerleme işlemiyle alıcıda elde edilebilmektedirler [2].



Şekil 31. "I" tipi (çerçeve-içi) ve "P" tipi (çerçeveler-arası) kodlanan görüntü çerçeveleri.

Hareket kestirimi yapıldıktan sonra, bir çerçevedeki görüntü elemanları ile daha önceki çerçeveye ait olan ve hareketleri yok edilen görüntü elemanları arasındaki farklar (YÇF sinyali) AKD işleminden geçirilirler. AKD için "I" tipi görüntü çerçeveleri ve "P" tipi görüntü çerçeveleri (YÇF) 8×8 boyutlu bloklara ayrılmaktadırlar. Kosinüs dönüşümü için 8×8 boyutlu hızlı ayırık kosinüs dönüşümü algoritması kullanılmaktadır [30]. Görüntü ayrılabilir (separable) işlemlerle modellenemediği için bu algorithmada bir blok için 16×13 çarpma ve 16×32 toplama işlemi yapılmaktadır. Burada $N \times N$ boyutlu bir blok için toplam hesap sayısı, N boyutlu vektör için gereken hesap sayısı $\times 2 \times N$ olmaktadır. Eğer görüntü ayrılabilir işlemlerle modellenemeseydi $2 \times N$ yerine $N^2=64$ kere N boyutlu AKD işlemi yapılacaktı.

AKD katsayılarının seviyelendirilmesi çerçeve-içi veya çerçevelerarası olmasına göre değişik olmaktadır. Çerçeve-içi sinyale ait bir blok için ortalama değer (DC) bileşeni (bloğun sol üst tarafındaki katsayı) fark kodlamasıyla kodlanır. Bu kodlama şekli aşağıdaki gibidir:



Şekil 32.Çerçeve-içi blokların DC bileşenlerinin fark kodlaması yapılması.

İki komşu blok için ortalama değer bileşenlerinin farkı alınmaktadır. Bu fark değeri:

$$\Delta DC_i = DC_i - DC_{i-1} \quad (92)$$

ifadesiyle hesaplanmaktadır. Elde edilen fark değerleri adım büyüklüğü 8 olan bir seviyelendiriciyle seviyelendirilmektedirler. DC bileşeninin diğerlerinden daha küçük adım büyüklüğü ile seviyelendirilmesinin nedeni, bu bileşenin genelde diğerlerinden daha yüksek olması ama buna rağmen 8x8 bloklaşma olmaması için yeterince yüksek hassaslıkta bırakılması gereğidir.

Diğer dönüşüm katsayıları Bölüm 1'de anlatılan biçimde seviyelendirilmektedirler. Çerçeve I tipi ise Şekil 2 b)'deki, P tipi ise Şekil 2 a)'daki seviyelendirme formu kullanılmaktadır. Çerçeve-içi DC bileşeni için 1, diğer bütün katsayılar için 31 ayrı seviyelendirici mevcuttur. DC dışındaki bileşenler için seviyelendirici adım büyüklüğü 2 ile 62 arasında değişen çift sayılardan seçilmektedir.

Hareket içeriğinden arındırılmış (bağımsız) katsayılar (kestirim hataları veya dönüşüm katsayıları) seviyelendirildikten sonra bu katsayılar kayıpsız kodlama işlemine sokularak veri hızı daha da azaltılır. Burada seğırtim boyu (RL: Run-length) kodlama ve istatistiksel kodlama kullanılmaktadır. Bu tez çalışmasında seğırtim boyu kodlama ve istatistiksel kodlama tabloları CCITT Rec. H261 [2]'e uygun olarak seçilmiştir.

Seğırtim boyu kodlaması aşağıdaki gibidir: Tarama sırasında peşpeşe gelen sıfırların sayısını artırmak amacıyla kosinüs dönüşümü sonucunda ortaya çıkan katsayılar zigzag taranmaktadır (Şekil 33). Şekil 33' de kutuların içerisine taramadaki sıra numaraları yazılmıştır.

DC →	1	2	6	7	15	16	28	29
	3	5	8	14	17	27	30	43
	4	9	13	18	26	31	42	44
	10	12	19	25	32	41	45	54
	11	20	24	33	40	46	53	55
	21	23	34	39	47	52	56	61
	22	35	38	48	51	57	60	62
	36	37	49	50	58	59	63	64

Şekil 33. AKD katsayılarının zigzag taranması.

Zigzag taranan katsayılar, (seğirtim boyu, boyut) ve (genlik) olmak üzere iki tip sembole ayrılırlar. Seğirtim boyu, 0 ile 15 arasında değer alabilir ve sıfır olmayan herhangi bir katsayıdan sonra peşpeşe dizilen sıfırların sayısını belirtir. Seğirtim boyu 15'den daha fazla ise (15,0) değeri seğirtim boyu=16 için kullanılır ve bu en fazla üç kere devam edebilir. Bundan sonra sonlandırıcı bir sembol kullanılır. Bundan sonra da devamlı bir (genlik) sembolü kullanılır. Boyut ise genlik değerini kodlamak için kullanılan bit sayısıdır.

Elde edilen (uzunluk, genlik) çiftleri istatistiksel bir kodlama şekli olan Huffman kodlamasıyla kodlanmaktadır [2].

Hareket vektörleri ve bölge biçimi bilgisi de istatistiksel olarak kodlanmaktadır. Hareket vektörlerinin yatay bileşenleri ve düşey bileşenleri için değişken uzunluklu kod sözcükleri (variable length codeword) kaynak [2]'ye göre düzenlenmiştir.

Çoğullayıcı kısımda parlaklık sinyali, renk sinyalleri, hareket vektörleri ve zamandaşlama sinyalleri çoğullanmaktadır. Çoğullayıcı hiyerarşik olarak çalışan dört kısımdan oluşmaktadır. Bunlar: görüntü çerçevesi (picture), blok grubu (group of block), makroblok (macroblock) ve blok olarak adlandırılmaktadır [2]. Her kısım için başlangıç kodları mevcuttur. QCIF boyutlu bir görüntü çerçevesi için üç adet blok grubu kullanılmaktadır. Her blok grubu 33 makrobloka bölünmüştür.

CCITT'nin H.261 tavsiyesine göre Tümleşik hizmetler sayısal şebekesi kanalları kullanılarak 64kbit/s veya 112kbit/s gibi kesin bir hız kısıtı altında video telefon sinyali iletimi yapılması istenmektedir. Bu kısıtı sağlamak için çoğullayıcıdan gelen değişik hızdaki bit grupları bir yastık bellekte (buffer memory) toplanmakta ve sabit hızla kanala verilmektedir. Yastık belleğin dolmasının ve tam boşalmasının getireceği sakıncaları önlemek için de yastık belleğin boşluğu ve doluluğu izlenerek seviyelendiricideki seviye sayısı (seviyelendirici adım büyüklüğü) değiştirilmektedir. Bu, çoğullayıcı ve yastık bellekten yapılan geri besleme ile kontrol edilir. Böylece boşalma eğilimindeki belleğe daha hızlı, taşma eğilimindeki belleğe de daha yavaş bit akışı sağlanmaktadır. Hareket ve fazla detay içeren görüntülerde kestirim yapma ve veri azaltma başarımının daha az olması nedeniyle yastık bellek daha çabuk dolmakta ve kodu çözülen görüntülerdeki bozulmalar, görüntünün hareket ve detay içeriğine göre değişmektedir.

Kanal kodlayıcı bölümünde hata düzelten kodlama olarak BCH (Bose-Chaudhuri-Hocquenghem) (511,493) ileri yönde hata düzelten kodlama (forward error correcting code) kullanılmaktadır [2]. Generatör polinomu (generator polynomial) aşağıdaki gibidir:

$$g(x) = (x^9 + x^4 + 1)(x^9 + x^6 + x^4 + x^3 + 1) \quad (93)$$

Örneğin, "01111...11" (493 bit) bir giriş verisi için ortaya çıkan düzeltici pariti bitler (correction parity bits): "011011010100011011" (18 bit) olarak ortaya çıkar.

Kod çözme aşamaları ise aşağıdaki sırada yapılmaktadır:

- 1 Kanal Kodçözme
- 2 Yastık bellekte verilerin biriktirilmesi
- 3 Renk, parlaklık ve zamandaşlama sinyallerinin ayrılması
- 4 Entropi kodunun çözülmesi
- 5 Zigzag sıranın düzeltilmesi
- 6 Normalize edilmiş katsayıların eski durumlarına getirilmeleri için normalizasyon katsayılarıyla çarpılmaları
- 7 Ters AKD işlemi
- 8 Fark sinyali ve bellekte tutulan bir önceki kod çözülen görüntü çerçevesine ait verilerin hareket vektörlerinin gösterdiği adreslerden alınarak toplanması
- 9 Görüntü çerçeveleri arasına yeni çerçeveler yerleştirmek amacıyla aradeğerleme yapılması.

2.4. Önerilen Yöntemin İşlem Karmaşıklığı

Önerilen yöntemdeki işlem sayısı kullanılan alt blok sayısına göre değişmektedir. Bu sayı, (91) eşitliğinde verilen farklı bölge sayısı kadardır. Örneğin 4 adet alt blok kullanılması durumunda 15 karşılaştırma işlemi yapılmaktadır. Sabit boyutlu blok karşılaştırmalarda yapılan karşılaştırma sayısı yanında bu sayı küçümsenir düzeydedir.

Örnek olarak, 16x16 boyutlu blokların kullanıldığı bir blok karşılaştırma işleminde üç adımlı yönlü arama ve ortalama mutlak hata ölçütleri kullanıldığında, 25x256 sayısı kadar karşılaştırma işlemi yapılmaktadır. Burada 25 sayısı, üç adımlı yönlü aramada arama sayısı, 256 sayısı ise bir blok için yapılan karşılaştırma sayısıdır. Önerilen yöntemde, yapılan karşılaştırma sayısı fazladan 15 olduğuna göre işlem sayısı $15/(25 \times 256) = 0.00234$ kat artmaktadır. Alt blok sayısı arttırılırsa işlem sayısı buna bağlı olarak artar.

2.5. Yapılan Deneysel Çalışmalar ve Kullanılan Görüntü Dizileri

Deneylerde video telefon amaçlı kaydedilmiş, QCIF formatına uygun (176x144 boyutlu) ve 256 parlaklık seviyesine sahip görüntü dizileri kullanılmaktadır. Bu görüntü dizileri, hareket içeriği bakımından üçe ayrılmaktadırlar:

Birinci tür görüntü dizisi sentetik olarak üretilmiş görüntülerden oluşmaktadır. Bu görüntülerde, değişik geometrik şekilli ve değişik boyutlu cisimler bilinen hızlarda ve yönlerde hareket etmektedirler. Sol üst köşede büyük boyutlu cisimler, sağ ve alt köşede ise küçük boyutlu cisimler bulunmaktadır. Zemin ise uzamsal koordinatlara göre

parlaklık deęerleri deęişen bir fondan oluşmaktadır. Bu görüntü dizisi tezde "sentetik görüntü dizisi" olarak adlandırılmaktadır. Sentetik görüntüler yardımıyla hem yazılım programlarının doğrulukları test edilmekte hem de kullanılan yöntemlerin performansları incelenmektedir.

İkinci tür görüntü dizileri "Claire", "Miss America" ve "Trevor" adlarıyla anılan standart görüntü dizileridir. Bu dizilerde, yavaş hareketler içeren baş ve omuz görüntüleri bulunmaktadır.

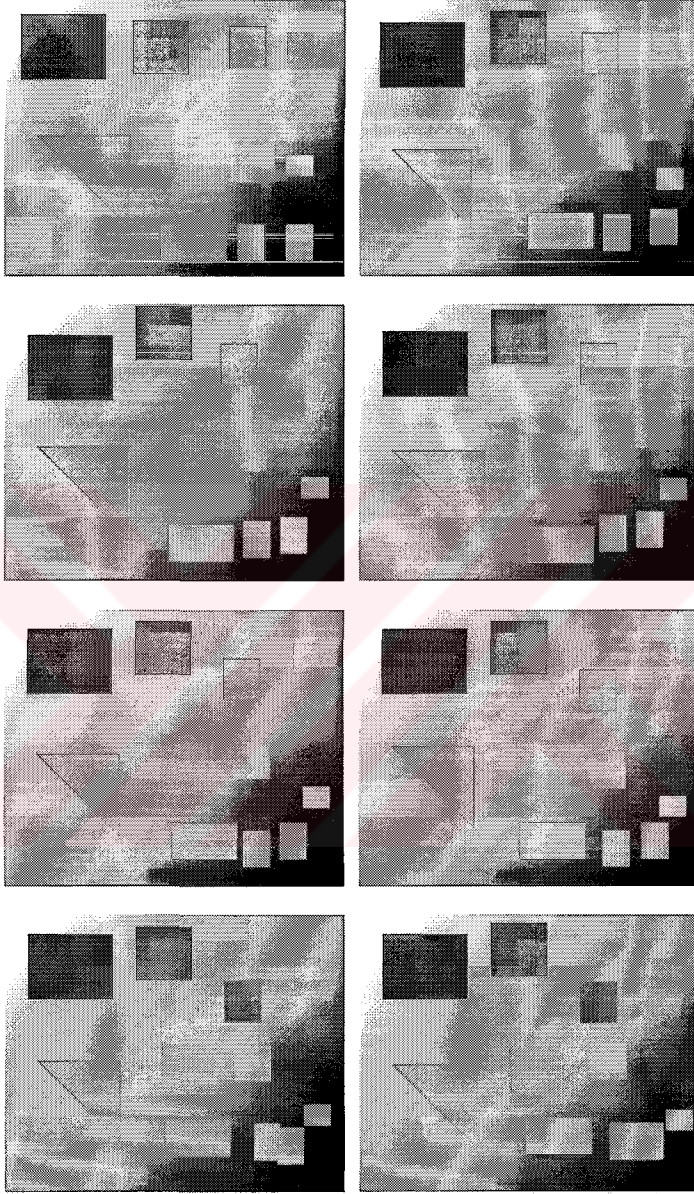
Üçüncü tür görüntü dizisi, bu tez yazarının bir video kamerayla kaydettięi görüntü dizisidir. Bu görüntü dizisindeki hareketler, ikinci tür görüntü dizilerinden biraz daha fazladır.

Kullanılan algoritmaların gürültüye karşı duyarlılıklarını incelemek amacıyla bütün görüntülere deęişik oranlarda Gauss gürültüsü eklenmiştir.

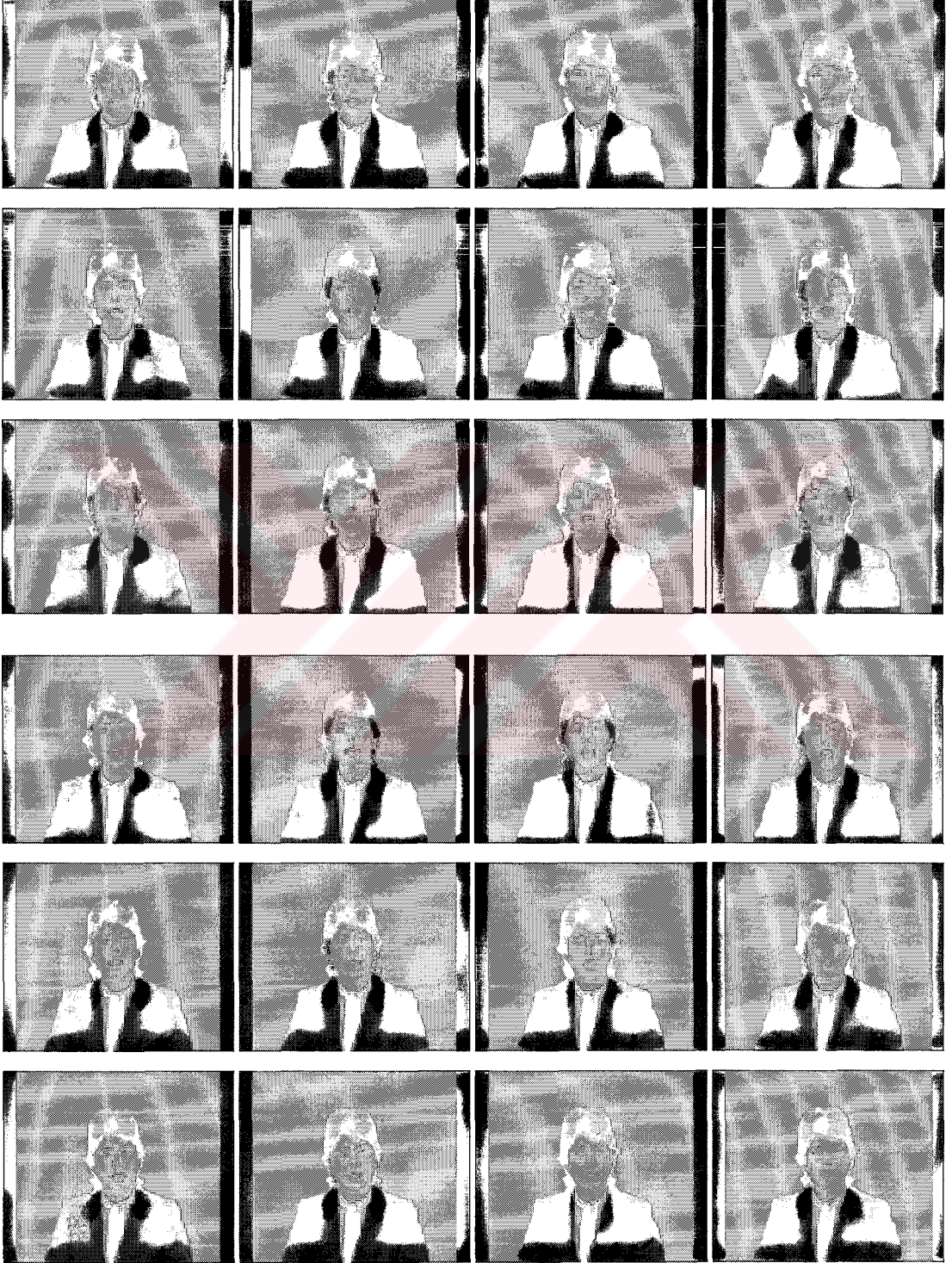
Bilgisayar programlama dili olarak PASCAL kullanılmaktadır.

Bilgisayar programı ile elde edilen sentetik görüntü dizisinin ilk 8 çerçevesi Şekil 34'de görülmektedir. Şekil 35, Şekil 36 ve Şekil 37'de sırasıyla "Claire", "Miss America" ve "Trevor" standart görüntü dizilerinin ilk 24'er görüntü çerçeveleri gösterilmiştir. Şekil 38'de ise tez yazarının görüntüsünden oluşan 8 görüntü çerçevesi görülmektedir.

Yazarın görüntüleri ve sentetik görüntüler saniyede 9 görüntü çerçevesi, standart görüntü dizileri ise saniyede 25 görüntü çerçevesi gösterilecek şekilde seçilmiştir.



Şekil 34. Sentetik görüntü dizisinin ilk 8 çerçevesi.



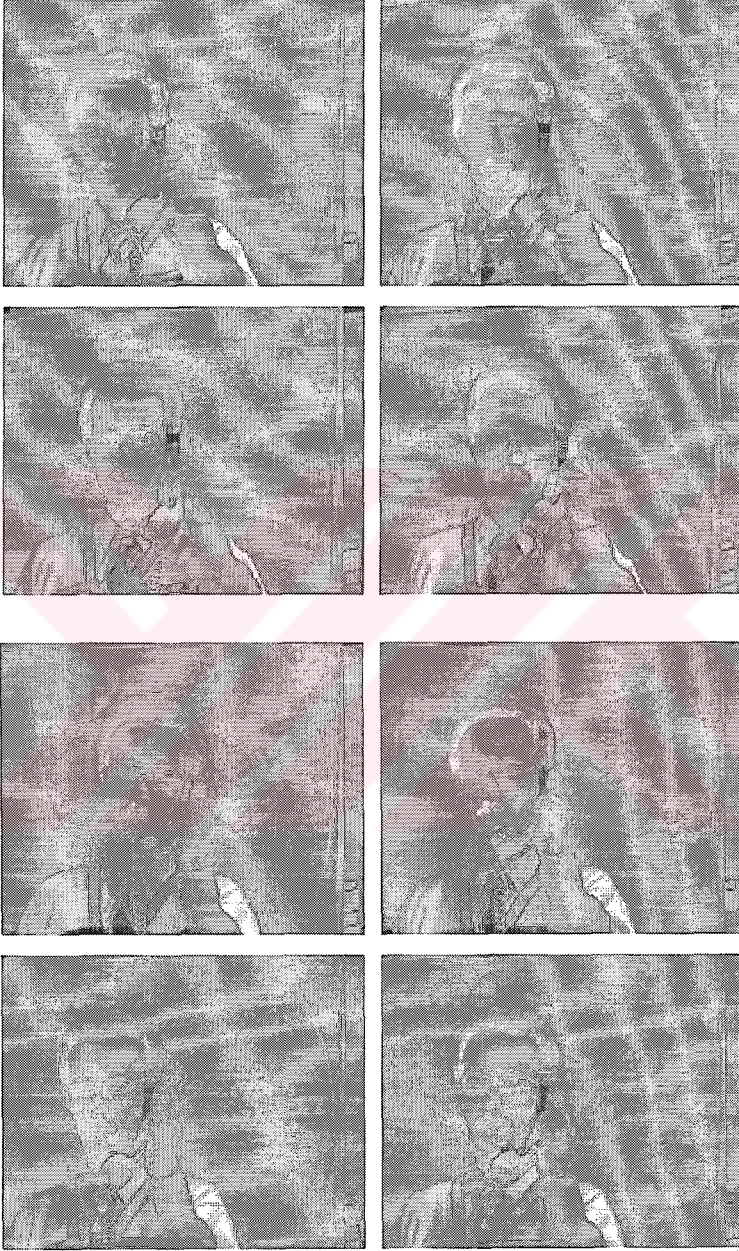
Şekil 35. "Claire" görüntü dizisinin ilk 24 çerçevesi.



Şekil 36. "Miss America" görüntü dizisinin ilk 24 çerçevesi.



Şekil 37. "Trevor " görüntü dizisinin ilk 24 çerçevesi.



Şekil 38. Yazarın görüntü dizisinin ilk 8 çerçevesi.

3. BULGULAR

3.1. Blok Karşılaştırmada Uygun Blok Boyutlarının Araştırılması

Simülasyon için kullanılan sentetik görüntülerde değişik boyutlara ve geometrik şekillere sahip sentetik cisim görüntüleri değişik yönlerde ve değişik hızlarda hareket ettirilmektedir (Şekil 34). Bütün cisimlerin hız ve yön bilgileri önceden bilinmekte ve bundan yararlanarak hareket kestirme yeteneği test edilmektedir. Bloksal hareket kestirme tekniklerinde uygun blok boyutları araştırılmaktadır. Cisimlerin hızlarının değişik olması ise uygun bir arama uzaklığı belirlenmesine yardım etmektedir.

Yapılan çalışmalarda arama uzaklığı ve blok boyutları değiştirilerek blok karşılaştırmalı hareket kestirme tekniklerinin performansı ölçülmeye çalışılmıştır.

Tablo 5, Tablo 6 ve Tablo 7'de sıralı aramalı blok karşılaştırma ile hareket kestirimi yapıldığında iki görüntü çerçevesi arasındaki mutlak hatanın ne kadar azaltılabildiği gösterilmektedir. Tablolardan da görüldüğü gibi, arama uzaklığı arttırıldığında YÇF'nin (kestirim hatası) entropisi (E_f) ve YÇF'nin değışintisi (σ^2 : variance) azalmakta, hareket vektörleri entropisi (E_h) ve mutlak hatanın azalması (% olarak) artmaktadır. Mutlak hatanın azalma yüzdesi aşağıdaki gibi hesaplanmaktadır [72]:

$$\% \text{ olarak azalma} = \frac{\sum(\text{HKY YÇF}) \sum(\text{HKV YÇF})}{\sum(\text{HKV YÇF})} \times 100 \quad (94)$$

Burada, HKY YÇF: Hareket kestirimi yokken yerdeğıştirmiş çerçeveler farkı ve HKV YÇF: Hareket kestirimi varken YÇF değerleridir. YÇF'nin entropisi de aşağıdaki gibi hesaplanmaktadır [33]:

$$E_f = - \sum_{k=0}^{255} P(k) \log_2 P(k) \quad (95)$$

Burada $p(k)$, herhangi bir YÇF değerinin k seviyesini alma olasılığıdır.

Hareket vektörleri entropisi aşağıdaki gibi hesaplanmaktadır[33]:

$$E_h = - \sum_{k=0}^P P_1(k) \log_2 P_1(k) - \sum_{k=0}^Q P_2(k) \log_2 P_2(k) \quad (96)$$

Burada P ve Q, sırasıyla yatay ve düşey yönlerdeki maksimum arama uzaklıklarıdır (her hareket vektörünün iki bileşeni vardır).

Tablo 5. Arama uzaklığına göre ortalama mutlak hatanın azalma yüzdesinin (%), YÇF değişimisinin (σ^2 : variance), YÇF entropisinin (E_f), hareket vektörleri entropisinin (E_h) ve toplam entropinin (E_t) değişimi (sentetik dizi).

Blok boyutları		Arama uzaklığı								
		0	1	2	3	4	5	6	7	
32x32	σ^2	1230.6	1038.8	874.2	728.3	583.8	493.8	460.8	439.3	
	%	0.000	14.080	25.820	36.350	45.960	51.660	54.130	55.570	
	E_f	2.107	2.311	2.194	2.121	2.113	2.085	2.030	1.994	
	E_h	0.000	0.003	0.004	0.005	0.005	0.006	0.006	0.006	
	E_t	2.107	2.314	2.198	2.126	2.118	2.091	2.036	1.950	
16x16	σ^2	1230.6	1026.4	848.3	690.8	532.3	424.4	378.5	347.4	
	%	0.000	15.310	29.310	41.560	52.980	60.500	64.470	67.010	
	E_f	2.107	2.250	2.062	1.918	1.882	1.829	1.733	1.665	
	E_h	0.000	0.006	0.007	0.008	0.009	0.010	0.011	0.012	
	E_t	2.107	2.256	2.069	1.926	1.891	1.839	1.744	1.677	
8x8	σ^2	1230.6	921.1	654.3	437.9	234.2	109.1	53.6	21.3	
	%	0.000	24.540	45.990	63.250	79.080	88.340	93.390	96.430	
	E_f	2.107	1.866	1.501	1.228	1.022	0.865	0.719	0.613	
	E_h	0.000	0.019	0.025	0.028	0.030	0.034	0.037	0.040	
	E_t	2.107	1.885	1.526	1.256	1.052	0.899	0.756	0.653	
4x4	σ^2	1230.6	908.0	638.6	422.6	225.4	103.2	43.3	9.3	
	%	0.000	25.550	47.420	64.820	80.260	89.650	95.270	98.370	
	E_f	2.107	1.768	1.402	1.056	0.832	0.608	0.429	0.325	
	E_h	0.000	0.059	0.071	0.081	0.092	0.101	0.112	0.118	
	E_t	2.107	1.827	1.473	1.137	0.924	0.709	0.541	0.443	

Değişinti değerleri de aşağıdaki gibi hesaplanmaktadır:

$$\sigma^2 = \frac{\sum_{m=0}^{175} \sum_{n=0}^{143} (I_{YCF} - I_{YCF_{ort}})^2}{(176 \times 144)} \quad (97)$$

Burada I_{YCF} : YÇF parlaklık değeri, $I_{YCF_{ort}}$: Bir görüntü çerçevesi için YÇF parlaklık ortalama değeridir.

Toplam entropi (E_t), her üç tabloda da arama uzaklığı arttıkça azalmaktadır. Blok büyüklüğü azaldıkça YÇF entropisi de azalmakta, hareket vektörleri entropisi artmaktadır. Her üç tabloda da hareket vektörleri entropisinin YÇF entropisinden çok daha az olduğu görülmektedir. Blok boyutlarının 4x4 seçilmesi en iyi sonucu vermektedir.

Burada E_h 'nin E_f yanında küçümsenir düzeyde olduğu görülmektedir. Ancak kodlama sırasında YÇF sinyali AKD işleminden sonra kodlanarak E_f yaklaşık on kat daha azaltılabilmektedir. Bu durumda E_h E_f 'nin yanında küçümsenemez duruma gelmektedir.

Tablo 6. Arama uzaklığına göre ortalama mutlak hatanın azalma yüzdesinin (%), YÇF değişiminin (σ^2 : variance), YÇF entropisinin (E_f), hareket vektörleri entropisinin (E_h) ve toplam entropinin (E_t) değişimi ("Claire" dizisi).

Blok boyutları		Arama uzaklığı							
		0	1	2	3	4	5	6	7
32x32	σ^2	71.94	47.58	41.28	41.28	41.28	41.28	41.28	41.28
	%	0.000	17.75	23.24	23.24	23.24	23.24	23.24	23.24
	E_f	2.750	2.687	2.663	2.663	2.663	2.663	2.663	1.663
	E_h	0.000	0.001	0.001	0.001	0.001	0.001	0.001	0.001
	E_t	2.750	2.688	2.664	2.664	2.664	2.664	2.664	2.664
16x16	σ^2	71.94	38.91	25.66	24.48	24.48	24.48	24.48	24.48
	%	0.000	23.90	34.62	36.14	36.14	36.14	36.14	36.14
	E_f	2.750	2.649	2.596	2.590	2.590	2.590	2.590	2.590
	E_h	0.000	0.002	0.003	0.003	0.003	0.003	0.003	0.003
	E_t	2.750	2.651	2.599	2.593	2.593	2.593	2.593	2.593
8x8	σ^2	71.94	31.15	15.33	13.49	13.49	13.49	13.49	13.49
	%	0.000	31.35	44.32	46.75	46.77	46.77	46.77	46.77
	E_f	2.750	2.587	2.510	2.496	2.494	2.494	2.494	2.494
	E_h	0.000	0.013	0.015	0.017	0.018	0.018	0.018	0.018
	E_t	2.750	2.600	2.525	2.513	2.512	2.512	2.512	2.512
4x4	σ^2	71.94	26.71	12.19	9.37	9.31	9.16	9.03	8.95
	%	0.000	39.11	52.89	56.96	57.30	57.51	57.77	58.07
	E_f	2.750	2.455	2.333	2.285	2.272	2.266	2.261	2.252
	E_h	0.000	0.069	0.093	0.111	0.122	0.131	0.138	0.144
	E_t	2.750	2.524	2.426	2.396	2.394	2.397	2.399	2.396

"Claire" dizisi video telefona uygun bir görüntü dizisidir. Burada ardışıl iki görüntü arasındaki hareket miktarı çok az olduğundan arama uzaklığını fazla arttırmanın gereksiz olduğu görülmektedir. Hareket kestirme 1., 4., 7. görüntü çerçeveleri için yapılmakta, böylece saniyede 9 görüntü çerçevesi iletilmektedir. Aradaki görüntüler ise aradeğerleme teknikleriyle yeniden elde edilebilmektedirler.

Yazarın kaydettiği görüntü dizisi de video telefon amaçlı kaydedilmiş bir görüntü dizisidir. Bu dizide arka plan düzgün bir parlaklık değerine sahip değildir. Hareketler ise "Claire" dizisine göre daha hızlıdır. Bu görüntü dizisi için arama uzaklığının "Claire" dizisinden daha fazla olması gerektiği görülmektedir.

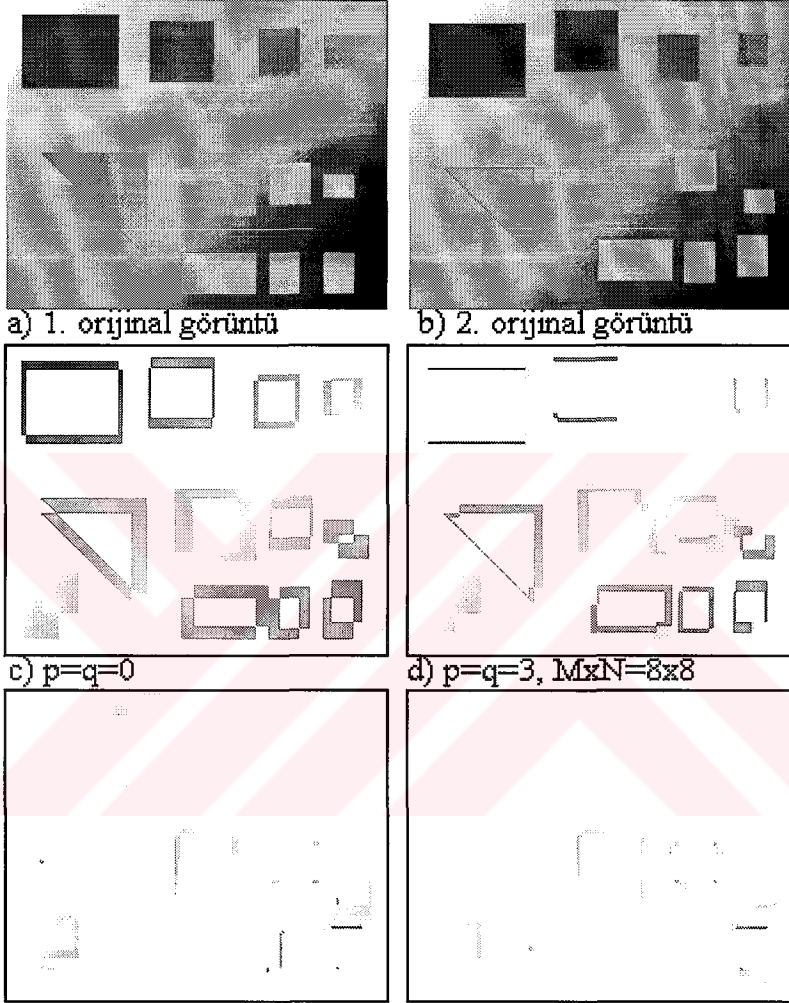
Bu sonuçlara göre, kestirim hatası entropisi ve hareket vektörleri entropisi arasında bir optimizasyon yaparak uygun bir blok boyutu saptamak uygun bir yol olarak

görülebilir. Ancak değişken görüntü istatistiği ve hareket içeriğine sahip sistemlerde sabit blok büyüklüğü kullanmak genellikle uygun değildir. Çünkü, YÇF'nin entropisi ve hareket vektörlerinin entropisi görüntü istatistiği ve hareket içeriğine bağlı olarak değişik oranlarda ortaya çıkmaktadır. Bu nedenle, toplam entropinin kararlı bir seviyede tutulması için değişken blok büyüklüğü kullanmak uygun bir yoldur.

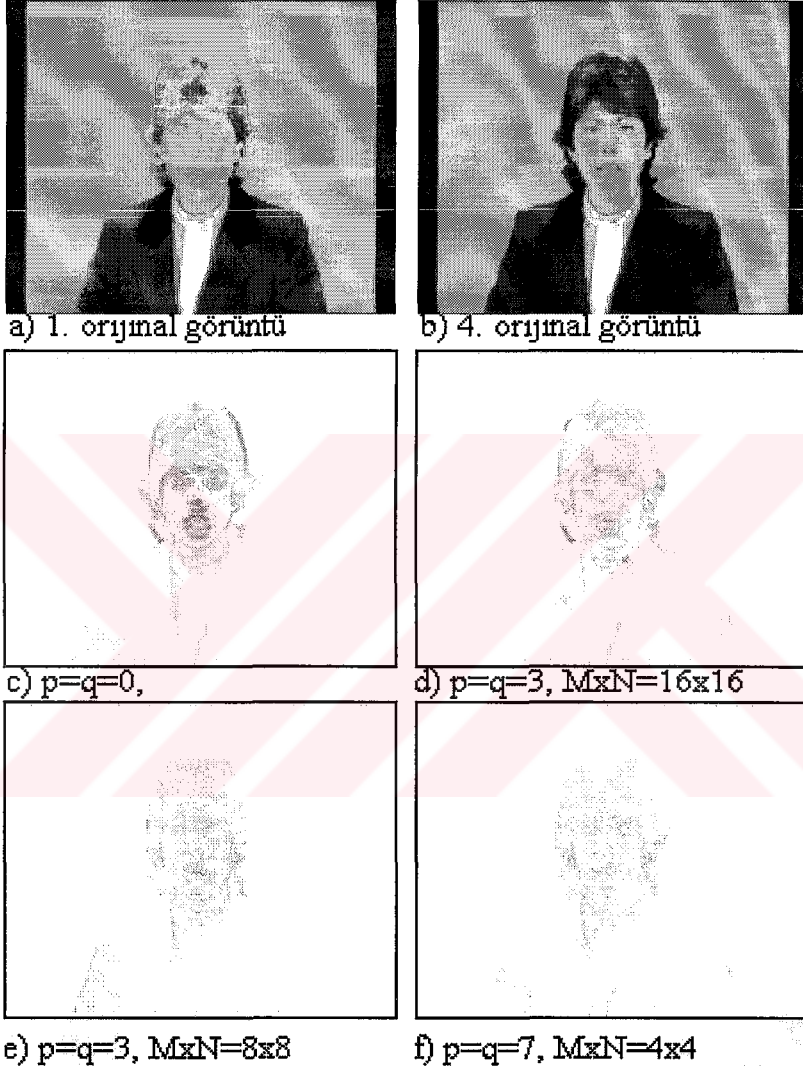
Tablo 7. Arama uzaklığına göre ortalama mutlak hatanın azalma yüzdesinin (%), YÇF değişim oranının (σ^2 : variance), YÇF entropisinin (E_f), hareket vektörleri entropisinin (E_h) ve toplam entropinin (E_t) değişimi (yazarın dizisi).

Blok boyutları		Arama uzaklığı							
		0	1	2	3	4	5	6	7
32x32	σ^2	482.73	312.03	224.61	163.18	125.02	120.90	120.64	120.64
	%	0.000	19.730	29.690	38.090	42.961	44.390	44.722	44.722
	E_f	5.072	4.918	4.836	4.699	4.631	4.591	4.582	4.582
	E_h	0.000	0.002	0.002	0.003	0.003	0.003	0.003	0.003
	E_t	5.072	4.920	4.838	4.702	4.634	4.594	4.585	4.585
16x16	σ^2	482.73	290.93	199.51	134.84	106.85	96.19	94.24	94.04
	%	0.000	22.860	34.750	43.200	48.290	50.630	51.370	51.420
	E_f	5.072	4.857	4.698	4.580	4.479	4.414	4.393	4.392
	E_h	0.000	0.004	0.005	0.006	0.006	0.007	0.007	0.007
	E_t	5.072	4.861	4.703	4.586	4.485	4.421	4.400	4.399
8x8	σ^2	482.73	265.91	163.58	99.54	72.63	62.62	59.52	59.25
	%	0.000	27.100	40.470	49.920	55.220	57.730	58.790	58.870
	E_f	5.072	4.784	4.606	4.443	4.308	4.230	4.202	4.200
	E_h	0.000	0.018	0.022	0.024	0.025	0.027	0.028	0.028
	E_t	5.072	4.802	4.628	4.467	4.333	4.257	4.230	4.228
4x4	σ^2	482.73	236.72	125.43	67.54	42.46	36.01	33.77	33.08
	%	0.000	32.300	48.240	58.120	63.380	65.400	66.360	66.660
	E_f	5.072	4.702	4.453	4.258	4.119	4.049	4.015	4.007
	E_h	0.000	0.075	0.098	0.109	0.119	0.127	0.133	0.138
	E_t	5.072	4.777	4.551	4.368	4.238	4.176	4.148	4.145

Şekil 39 ve Şekil 40'da sıralı aramalı blok karşılaştırma tekniğiyle hareket kestirme sonucunda elde edilen hata görüntülerinin arama uzaklığına bağlı olarak görünüşleri verilmektedir. Burada 4. ve 1. görüntü çerçevelerinin farkı alınmaktadır. Şekil 39 ve Şekil 40'da görüldüğü gibi, blok boyutları küçük ve arama uzaklığı fazla olduğu zaman hareketler daha iyi kestirilmektedir. Hiç hareket kestirimi yapılmadığında ise fark (YÇF) görüntüsünün daha belirgin olduğu görülmektedir. Fark görüntülerinin, YÇF parlaklık değerlerinin 255 değerinden çıkarılmasıyla $(255 - |Y_{\text{YÇF}}|)$, beyaz zemin üzerinde görülmesi sağlanmıştır.



Şekil 39. Arama uzaklığına ve hareket kestirmede kullanılan blok boyutlarına göre elde edilen fark (YÇF) görüntüleri (sentetik görüntü dizisi).



Şekil 40. Arama uzaklığına ve hareket kestirmede kullanılan blok boyutlarına göre elde edilen fark (YÇF) görüntüleri ("Claire" görüntü dizisi).

3.2. Değişik Hareket Kestirme Tekniklerinin Performanslarının Karşılaştırmalı Olarak İncelenmesi

Bu bölümde, blok karşılaştırma teknikleri ve türevsel teknikler ile hareket kestiriminin performansları karşılaştırmalı olarak incelenmektedir. İncelemeler üç ayrı tip görüntü dizileri (sentetik, standart ve yazarın kaydettiği) üzerinde yapılmaktadır. Önce, kullanılan algoritmaların gürültüye karşı nasıl sonuç verdikleri araştırılmaktadır. Daha sonra bazı performans ölçütleri kullanılarak karşılaştırmalar yapılmaktadır.

3.2.1. Gürültünün Kestirim Hatası Entropisine Olan Etkileri

Bu bölümde, sıralı aramalı blok karşılaştırma, hızlı arama yöntemleri kullanan blok karşılaştırma ve türevsel tekniklerin Gauss gürültüsü eklenmiş görüntüler üzerinde nasıl sonuç verdikleri incelenmektedir. Gauss gürültüsü eklenmiş görüntülerde sinyal/gürültü oranı (SGO) aşağıdaki ifadeyle hesaplanmaktadır:

$$SGO(\text{dB}) = 10 \log_{10} \frac{\sigma_s^2}{\sigma_g^2} \quad (98)$$

Burada, σ_s^2 : Orijinal görüntünün parlaklık değerlerinin değişintisi, σ_g^2 : gürültünün değişintisidir.

3.2.1.1. Sentetik Görüntü Dizisi İçin Yapılan İncelemeler

Sentetik görüntü dizisinde 7. ve 4. görüntü çerçevesi arasındaki hareketler kestirilmeye çalışılmaktadır. Tablo 8'de altı ayrı sinyal/gürültü oranı için değişik hareket kestirme tekniklerinin Gauss gürültüsü eklenmiş görüntüler üzerinde hareketi izleme yetenekleri verilmiştir. Tablodaki değerler kestirim hatasına (YÇF) ilişkin entropi değerleridir.

Entropinin hesaplanması için, önce 176x144 boyutlu bir görüntü çerçevesi için elde edilen YÇF sinyalinin (hata görüntüsü) genliği alınmaktadır. Daha sonra görüntü elemanı başına düşen entropi bulunmaktadır. Entropi, genlik alma işlemi yapılmadan da bulunabilirdi. Fakat buradaki amaç bir karşılaştırma yapmak olduğu için genlik alınarak işlem yapılmaktadır. Entropi (95) ifadesiyle hesaplanmaktadır.

Tablo 8. Değişik hareket kestirme tekniklerinin Gauss gürültüsü eklenmiş sentetik görüntüler üzerinde göreceli hareket izleme performansı, YÇF' nin entropisi (7. çerçeve ve 4. çerçeve arasındaki hareket).

BLOK BOYUT	S/G	HKY	SBK	UAYA	LA	DLA	BSBA	EYA	CA	DA	THK
32x32	∞	1.45	1.37	1.40	1.42	1.40	1.40	1.41	1.41	1.40	1.43
	50	1.83	1.66	1.69	1.71	1.67	1.70	1.77	1.69	1.71	1.73
	40	3.51	3.15	3.26	3.40	3.25	3.36	3.37	3.35	3.32	3.43
	30	4.40	4.05	4.22	4.31	4.22	4.24	4.31	4.24	4.24	4.36
	21	5.77	5.35	5.61	5.68	5.62	5.67	5.68	5.63	5.63	5.72
	10	7.04	6.82	6.98	7.01	6.98	6.99	7.00	6.89	6.96	7.01
16x16	∞	1.45	1.33	1.37	1.42	1.38	1.38	1.40	1.38	1.37	1.41
	50	1.83	1.45	1.49	1.56	1.49	1.50	1.52	1.49	1.47	1.50
	40	3.51	2.83	2.87	2.99	2.87	2.99	2.98	2.91	2.92	3.16
	30	4.40	3.60	3.81	3.82	3.81	3.82	3.83	3.81	3.82	4.10
	20	5.77	5.17	5.31	5.42	5.31	5.38	5.39	5.34	5.35	5.51
	10	7.04	6.72	6.93	6.99	6.93	6.86	6.98	6.89	6.95	6.99
8x8	∞	1.45	0.47	0.47	0.54	0.49	0.52	0.55	0.55	0.54	1.03
	50	1.83	0.56	0.60	0.71	0.58	0.61	0.63	0.63	0.62	1.07
	40	3.51	2.33	2.43	2.48	2.44	2.45	2.46	2.43	2.44	3.01
	30	4.40	3.30	3.42	3.63	3.32	3.64	3.65	3.57	3.58	3.82
	20	5.77	4.93	5.28	5.41	5.29	5.32	5.30	5.33	5.32	5.43
	10	7.04	6.50	6.82	6.93	6.82	6.84	6.93	6.87	6.86	6.91
4x4	∞	1.45	0.17	0.19	0.22	0.20	0.21	0.21	0.20	0.21	0.97
	50	1.83	0.25	0.28	0.31	0.28	0.29	0.29	0.29	0.29	1.02
	40	3.51	1.68	1.77	1.81	1.72	1.73	1.78	1.75	1.74	2.14
	30	4.40	3.03	3.22	3.42	3.20	3.33	3.34	3.25	3.27	3.44
	20	5.77	4.53	5.27	5.40	5.22	5.31	5.30	5.29	5.29	5.42
	10	7.04	6.08	6.52	6.75	6.58	6.65	6.67	6.71	6.73	6.87

Kısaltmaların anlamları aşağıdaki gibidir:

S/G: Görüntüye Gauss gürültüsü eklenmesi sonucu elde edilen sinyal/gürültü oranı

HKY : Hareket kestirimi yok

SBK : Sıralı aramalı blok karşılaştırma

UAYA: Üç adımlı yönlü aramalı hızlı blok karşılaştırma

LA : Logaritmik aramalı blok karşılaştırma

DLA : Düzeltilmiş logaritmik aramalı blok karşılaştırma

BSBA: Bir seferde bir aramalı blok karşılaştırma

EYA : Eşlenik yönlerde aramalı blok karşılaştırma

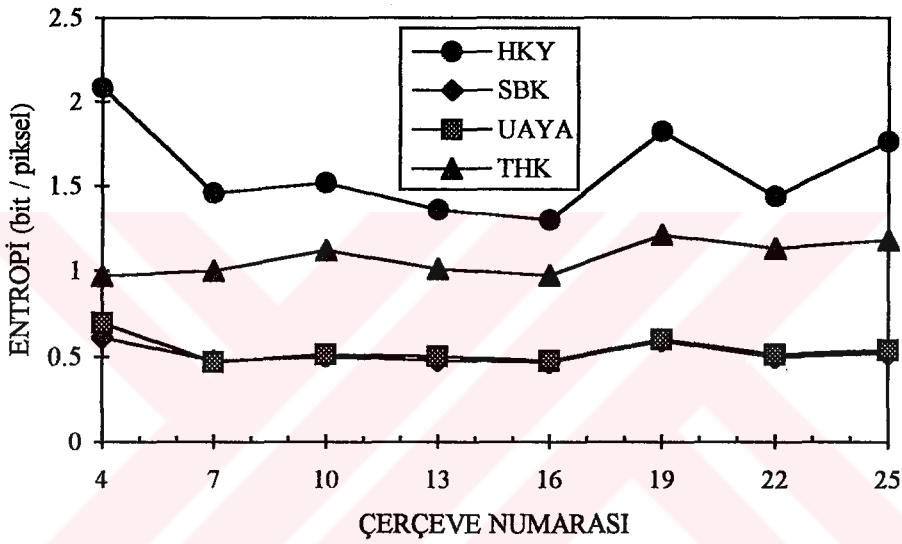
CA : Çapraz aramalı blok karşılaştırma

DA : Dikgen aramalı blok karşılaştırma

THK : Türevsel hareket kestirme

Sinyal/gürültü oranı azaldığı zaman bütün yöntemlerde entropi miktarı artmaktadır. Tablo 8'den görüldüğü gibi, değişik Sinyal/gürültü oranlarında incelenen bütün yöntemler hareket kestirimi yapılmadığı duruma göre entropi azaltımını sağlamaktadırlar. Ancak gürültü miktarı arttıkça hızlı arama teknikleri daha kötü sonuç vermektedir. En iyi sonuç ise sıralı arama tekniği kullanılan blok karşılaştırma ile (SBK) elde edilmektedir.

Şekil 41'de, gürültü eklenmemiş sentetik görüntü dizisinde değişik hareket kestirme yöntemleriyle elde edilmiş YÇF sinyalinin (kestirim hatasının) entropisi 1 saniyelik görüntü dizisi için gösterilmektedir.



Şekil 41. Sentetik görüntü dizisinde kestirim hatasının entropisi.

Burada, $|E_f|$: YÇF sinyalinin genlik değeridir (mutlak hata).

Şekil 41'de görüldüğü gibi, en düşük entropi değeri sıralı arama tekniği kullanılan blok karşılaştırma ile elde edilmektedir. Burada üç adımlı yönlü arama tekniğinin sıralı aramaya yakın performans gösterdiği görülmektedir.

3.2.1.2. Standart Görüntü Dizisi İçin Yapılan İncelemeler

Standart görüntü dizilerinden "Claire" görüntü dizisi üzerinde algoritmaların performansları incelenmiştir. Sentetik görüntüler için yapılan bütün incelemeler bu görüntüler için de yapılmıştır.

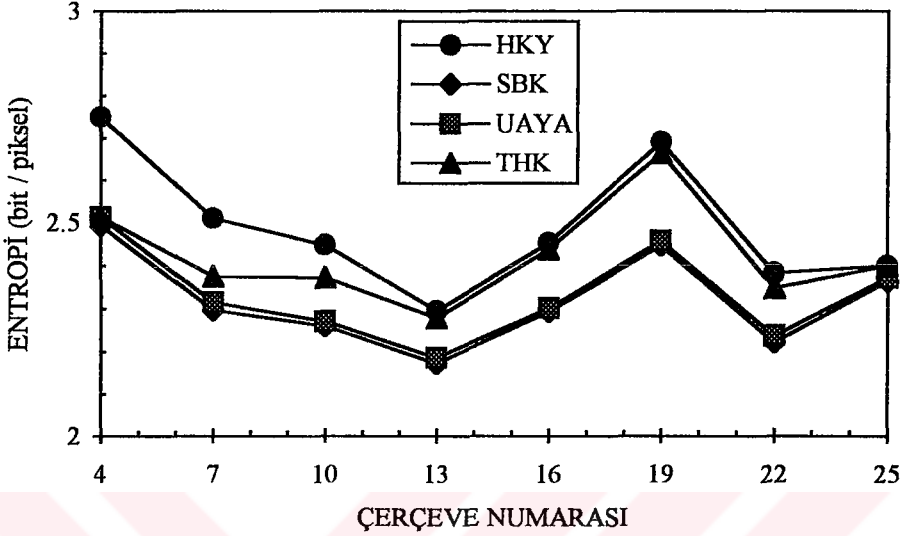
Tablo 9. Değişik hareket kestirme tekniklerinin Gauss gürültüsü eklenmiş "Claire" görüntü dizisi üzerinde göreceli hareket izleme performansı, YÇF' nin entropisi (7. çerçeve ve 4. çerçeve arasındaki hareket).

BLOK BOYUT	S/G	HKY	SBK	UAYA	LA	DLA	BSBA	EYA	CA	DA	THK
32x32	∞	2.51	2.38	2.38	2.41	2.40	2.42	2.42	2.41	2.40	2.47
	50	2.51	2.41	2.46	2.48	2.45	2.46	2.46	2.45	2.44	2.49
	40	3.03	2.88	2.95	2.98	2.96	2.98	2.99	2.97	2.97	3.02
	30	3.92	3.76	3.85	3.88	3.84	3.89	3.89	3.87	3.87	3.90
	21	5.20	5.06	5.15	5.17	5.16	5.17	5.17	5.18	5.18	5.29
	10	6.63	6.52	6.58	6.61	6.60	6.61	6.61	6.60	6.60	6.62
16x16	∞	2.51	2.30	2.30	2.33	2.32	2.34	2.32	2.32	2.33	2.46
	50	2.51	2.36	2.36	2.40	2.39	2.40	2.39	2.39	2.40	2.48
	40	3.03	2.86	2.87	2.90	2.87	2.88	2.88	2.87	2.87	2.93
	30	3.92	3.74	3.77	3.81	3.78	3.79	3.79	3.80	3.80	3.88
	20	5.20	5.04	5.10	5.12	5.10	5.11	5.12	5.11	5.11	5.19
	10	6.63	6.46	6.54	6.59	6.54	6.57	6.58	6.56	6.56	6.60
8x8	∞	2.51	2.30	2.32	2.33	2.31	2.32	2.34	2.32	2.34	2.36
	50	2.51	2.30	2.32	2.34	2.32	2.33	2.34	2.33	2.33	2.38
	40	3.03	2.77	2.82	2.84	2.82	2.83	2.83	2.82	2.82	2.90
	30	3.92	3.62	3.72	3.78	3.73	3.74	3.75	3.74	3.74	3.75
	20	5.20	4.80	4.96	4.99	4.94	4.98	4.97	4.95	4.95	4.98
	10	6.63	6.17	6.38	6.41	6.39	6.40	6.40	6.39	6.39	6.51
4x4	∞	2.51	2.11	2.24	2.28	2.25	2.26	2.26	2.27	2.27	2.30
	50	2.51	2.12	2.26	2.29	2.26	2.28	2.27	2.28	2.27	2.32
	40	3.03	2.53	2.76	2.80	2.76	2.79	2.79	2.78	2.77	2.85
	30	3.92	3.32	3.58	3.59	3.58	3.61	3.60	3.59	3.58	3.70
	20	5.20	4.25	4.59	4.68	4.62	4.71	4.72	4.69	4.69	4.85
	10	6.63	5.57	6.00	6.12	6.01	6.15	6.16	6.13	6.12	6.31

"Claire" görüntü dizisinde 7. ve 4. görüntü çerçevesi arasındaki hareketler kestirilmeye çalışılmaktadır. Tablo 9'da altı ayrı sinyal/gürültü oranı için değişik hareket kestirme tekniklerinin Gauss gürültüsü eklenmiş görüntüler üzerinde hareketi izleme yetenekleri verilmiştir. Tablodaki değerler kestirim hatasına (YÇF) ilişkin entropi değerleridir.

Sinyal/gürültü oranı azaldığı zaman bütün yöntemlerde entropi miktarı artmaktadır. Tablo 9'dan görüldüğü gibi, değişik Sinyal/gürültü oranlarında incelenen bütün yöntemler hareket kestirimi yapılmadığı duruma göre entropi azaltımını sağlamaktadırlar. Bu görüntü dizisi için de gürültünün hızlı arama tekniklerine olan olumsuz etkileri tablodan görülmektedir. En iyi sonuç yine sıralı arama tekniği kullanılan blok karşılaştırma ile (SBK) elde edilmektedir.

Şekil 42'de, gürültü eklenmemiş "Claire" görüntü dizisinde değişik hareket kestirme yöntemleriyle elde edilmiş YÇF sinyalinin (kestirim hatasının) entropisi 25 görüntü çerçevesi için gösterilmektedir.



Şekil 42. "Claire" görüntü dizisinde kestirim hatasının entropisi.

Şekil 42'de görüldüğü gibi, "Claire" dizisi için de en düşük entropi değeri sıralı arama tekniği kullanılan blok karşılaştırma ile elde edilmektedir. Yine üç adımlı yönlü arama tekniğinin sıralı aramaya yakın performans gösterdiği görülmektedir.

3.2.1.3. Yazarın Kaydettiği Görüntü Dizisi İçin Yapılan İncelemeler

Yazarın kaydettiği görüntü dizisi üzerinde, sentetik görüntüler için yapılan performans incelemeleri aynı şekilde yapılmıştır.

Yazarın görüntü dizisinde 7. ve 4. görüntü çerçevesi arasındaki hareketler kestirilmeye çalışılmaktadır. Tablo 10'da altı ayrı sinyal/gürültü oranı için değişik hareket kestirme tekniklerinin Gauss gürültüsü eklenmiş görüntüler üzerinde hareketi izleme yetenekleri verilmiştir. Tablodaki değerler kestirim hatasına (YÇF) ilişkin entropi değerleridir.

Sinyal/gürültü oranı azaldığı zaman bütün yöntemlerde entropi miktarı artmaktadır. Tablo 10'dan görüldüğü gibi elde edilen sonuçlar sentetik dizi ve "Claire" dizisi için elde edilen sonuçlara benzerdir.

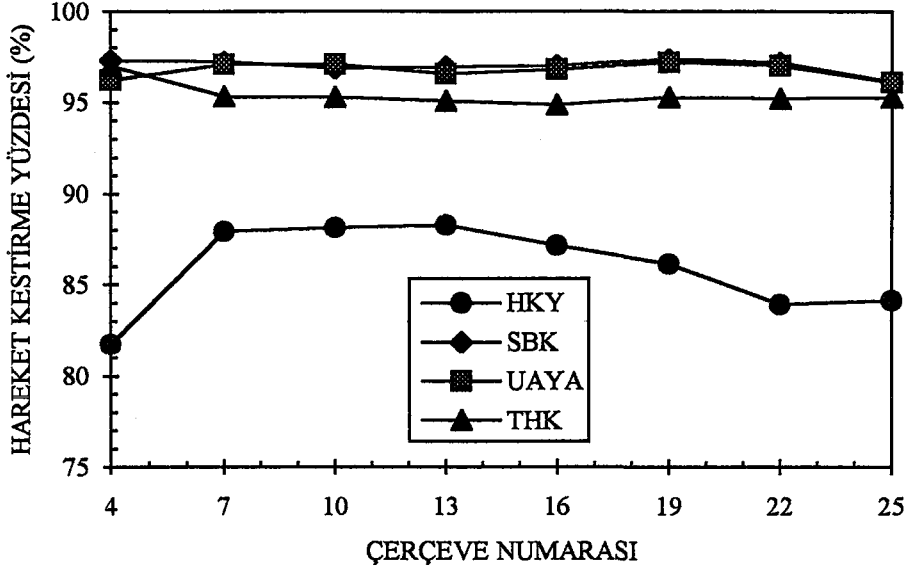
Tablo 10. Değişik hareket kestirme tekniklerinin Gauss gürültüsü eklenmiş yazının görüntü dizisi üzerinde göreceli hareket izleme performansı, YÇF' nin entropisi (7. çerçeve ve 4. çerçeve arasındaki hareket).

BLOK BOYUT	S/G	HKY	SBK	UAYA	LA	DLA	BSBA	EYA	CA	DA	THK
32x32	∞	6.27	5.47	5.49	5.58	5.46	5.57	5.57	5.56	5.53	5.82
	50	6.27	5.48	5.50	5.60	5.48	5.59	5.60	5.58	5.57	5.84
	40	6.30	5.54	5.58	5.67	5.59	5.63	5.64	5.63	5.63	5.92
	30	6.32	5.58	5.67	5.95	5.60	5.92	5.93	5.91	5.91	6.01
	21	6.43	5.74	5.89	6.09	5.88	6.02	6.02	6.03	6.02	6.18
	10	6.84	6.30	6.48	6.62	6.50	6.53	6.54	6.52	6.53	6.63
16x16	∞	6.27	5.40	5.40	5.43	5.41	5.41	5.43	5.41	5.42	5.44
	50	6.27	5.40	5.41	5.45	5.42	5.43	5.44	5.42	5.43	5.47
	40	6.30	5.40	5.42	5.48	5.44	5.46	5.47	5.46	5.47	5.58
	30	6.32	5.43	5.46	5.54	5.50	5.49	5.49	5.50	5.50	5.64
	20	6.43	5.66	5.82	6.95	5.83	5.89	5.87	5.86	5.84	9.99
	10	6.84	6.24	6.42	6.49	6.43	6.48	6.49	6.48	6.47	6.50
8x8	∞	6.27	5.13	5.15	5.26	5.17	5.25	5.26	5.26	5.26	5.39
	50	6.27	5.14	5.16	5.29	5.18	5.28	5.29	5.30	5.29	5.41
	40	6.30	5.14	5.23	5.34	5.23	5.34	5.35	5.37	5.36	5.48
	30	6.32	5.19	5.32	5.41	5.32	5.42	5.43	5.40	5.41	5.61
	20	6.43	5.33	5.46	5.53	5.47	5.51	5.50	5.47	5.49	5.88
	10	6.84	6.12	6.30	6.42	6.32	6.35	6.34	6.34	6.33	6.42
4x4	∞	6.27	4.74	4.75	4.77	4.76	4.77	4.77	4.76	4.76	5.16
	50	6.27	4.78	4.80	4.82	4.80	4.81	4.81	4.80	4.80	5.20
	40	6.30	4.78	4.83	4.92	4.86	4.88	4.87	4.87	4.86	5.26
	30	6.32	4.82	4.89	4.98	4.95	5.12	5.10	5.02	5.01	5.32
	20	6.43	5.07	5.34	5.38	5.35	5.39	5.38	5.37	5.38	5.57
	10	6.84	5.82	6.18	6.36	6.21	6.25	6.23	6.22	6.22	6.30

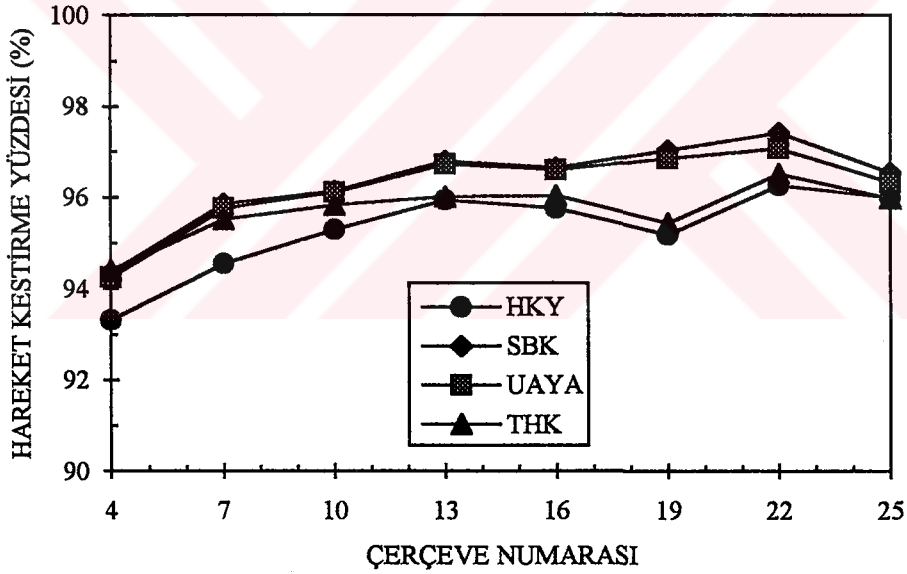
3.2.2. Değişik Yöntemlere Göre Görüntü Elemanlarının Kestirilebilme Oranları

Bir görüntü elemanının kestirilen parlaklık değeri ile gerçek parlaklık değeri arasındaki fark belli bir eşik değerinden küçükse, doğru bir kestirme işlemi yapıldığı kabul edilmektedir. Buradaki çalışmada, eşik seviyesi olarak bazı kaynaklarda [31], [41] kullanılan eşik seviyesi ($|T| \leq 3$) kullanılmaktadır.

Şekil 43'de kestirilebilen görüntü elemanı değerleri % olarak verilmektedir. Sentetik görüntülerde hareket kestirimi yapıldığında, görüntü elemanları parlaklık değerleri yukarıda kullanılan ölçüte göre yaklaşık % 90'ın üzerinde bir başarımla kestirilebilmektedir.



a)



b)

Şekil 43. Kestirilebilen görüntü elemanı değerlerinin yüzdesi ($|\text{eşik}| \leq 3$),

a) sentetik görüntü dizisi, b) "Claire" görüntü dizisi.

En iyi sonucu sıralı aramalı blok karşılaştırma yönteminin verdiği görülmektedir. "Claire" görüntü dizisinde ise bu oran yaklaşık % 95 düzeyinde olmaktadır. Bu dizi için de en iyi sonucu sıralı aramalı blok karşılaştırmanın verdiği görülmektedir.

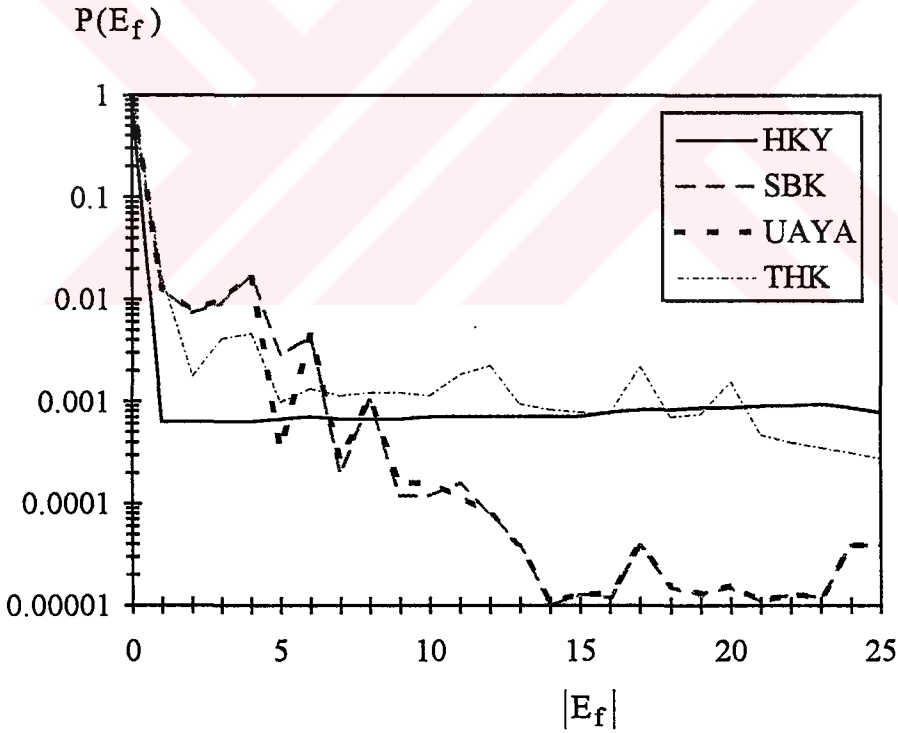
3.2.3. Değişik Yöntemlerin YÇF Olasılık Yoğunluk İşlevlerinin Karşılaştırılması

Değişik algoritmalarla elde edilen kestirim hatalarının (YÇF parlaklık değerlerinin) ortaya çıkma olasılıkları da hareket kestirme performansı için bir ölçü olarak kabul edilebilir [33]. Burada, bir görüntü çerçevesi için YÇF sinyalinin genliği alınmakta ve daha sonra olasılık yoğunluk işlevi (99) ifadesine göre hesaplanmaktadır.

$$P(E_f) = \frac{1}{176 \times 144} S_{E_f}, \quad 0 \leq |E_f| \leq 255 \quad (99)$$

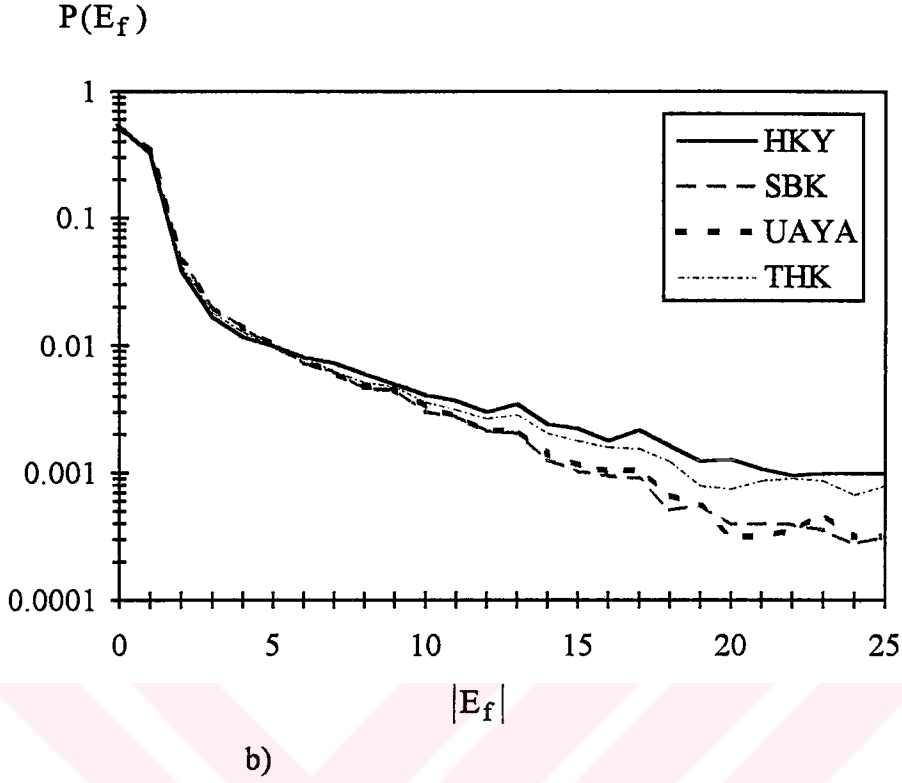
Burada S_{E_f} , bir YÇF görüntüsünde parlaklık değerinin genliği $|E_f|$ olan görüntü elemanlarının sayısıdır.

Şekil 44'de değişik hareket kestirme algoritmalarıyla elde edilen kestirim hatalarının ortaya çıkma olasılıkları verilmektedir.



a)

Şekil 44. Değişik hareket kestirme teknikleri kullanılarak elde edilen YÇF sinyali için olasılık yoğunluk işlevleri, a) sentetik görüntü dizisi, b) "Claire" görüntü dizisi.



Şekil 44. (devam).

Şekilde görüldüğü gibi, hareketlerin iyi kestirildiği yöntemlerde hata olasılığı daha az olmaktadır. Hareket kestirme tekniği kullanılmadığında ise hata olasılığının belirgin olarak daha fazla olduğu görülmektedir.

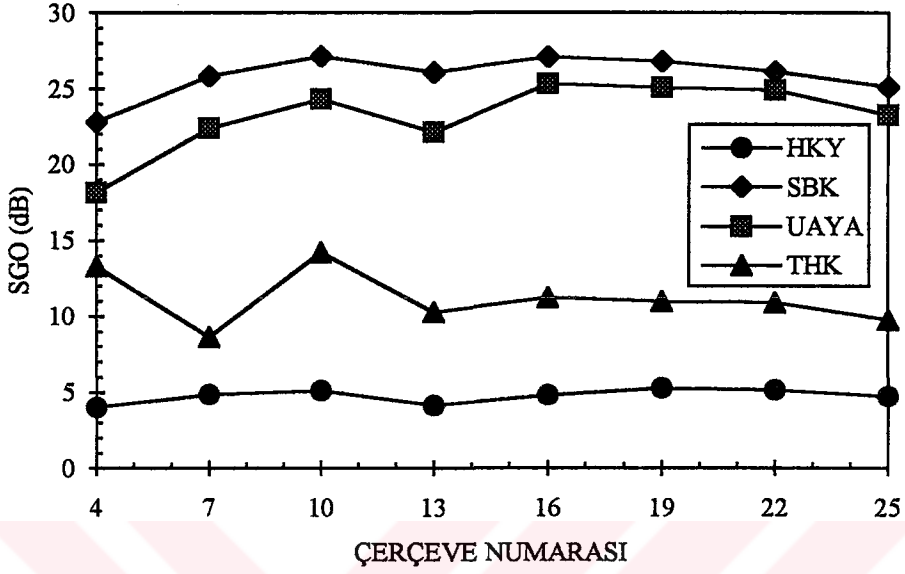
3.2.4. Değişik Yöntemlerde Kestirim Hatasının Bir Önceki Orijinal Görüntüye Göre Sinyal/Gürültü Oranı

Kestirim hatasının bir önceki orijinal görüntüye göre sinyal/gürültü oranı da performans için bir ölçü olarak kabul edilebilir [33]. Bu ölçüt, hareket kestirmeli farkları alınan ardışıl iki görüntüden birincisinin değişintisinin hata görüntüsünün değişintisine oranının logaritması olarak tanımlanmaktadır [33].

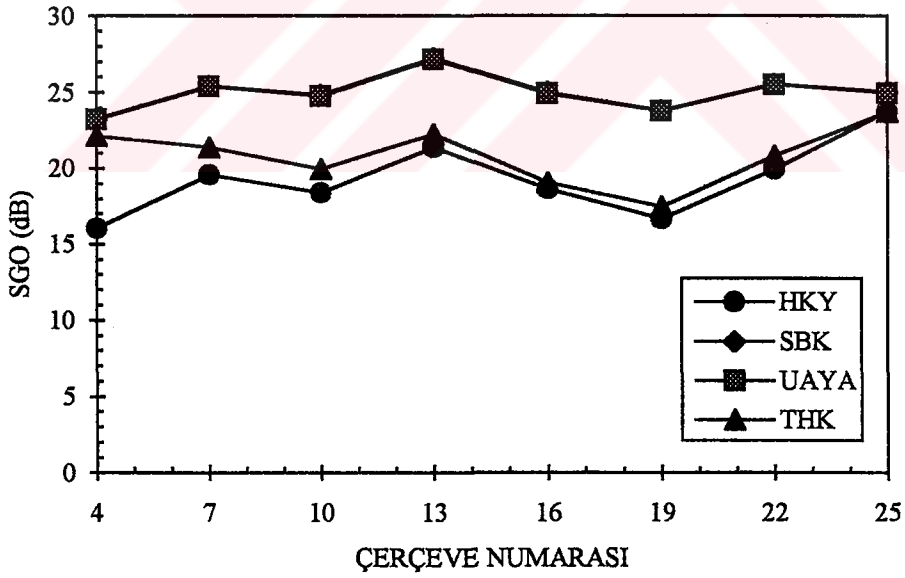
$$SGO(\text{dB}) = 10 \log_{10} \frac{\text{Birinci orijinal görüntünün değişintisi}}{\text{Hata görüntüsünün değişintisi}} \quad (100)$$

Şekil 45'de hareket kestirimi yapılarak elde edilmiş YÇF görüntüleri için sinyal/gürültü oranı gösterilmektedir. Şekilde görüldüğü gibi, sıralı arama tekniğiyle

yapılan blok karşılaştırma en iyi sonucu vermekte, UAYA buna yakın performans göstermektedir. THK tekniği ise burada en kötü sonucu vermektedir.



a)



b)

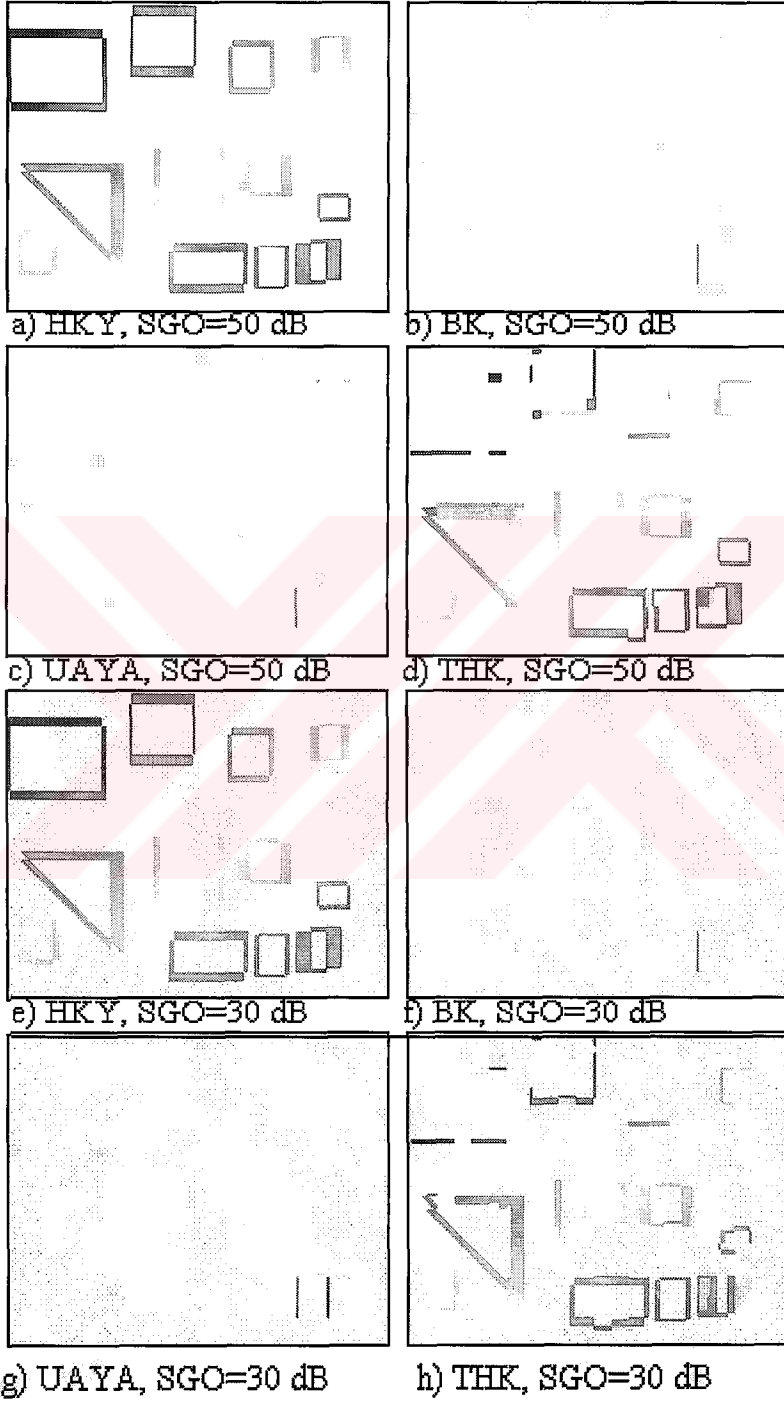
Şekil 45. Kestirim hatasının bir önceki orijinal görüntüye göre sinyal / gürültü oranı (SGO), a) sentetik görüntü dizisi, b) "Claire" görüntü dizisi.

3.2.5. Görsel (subjektif) sonuçlar

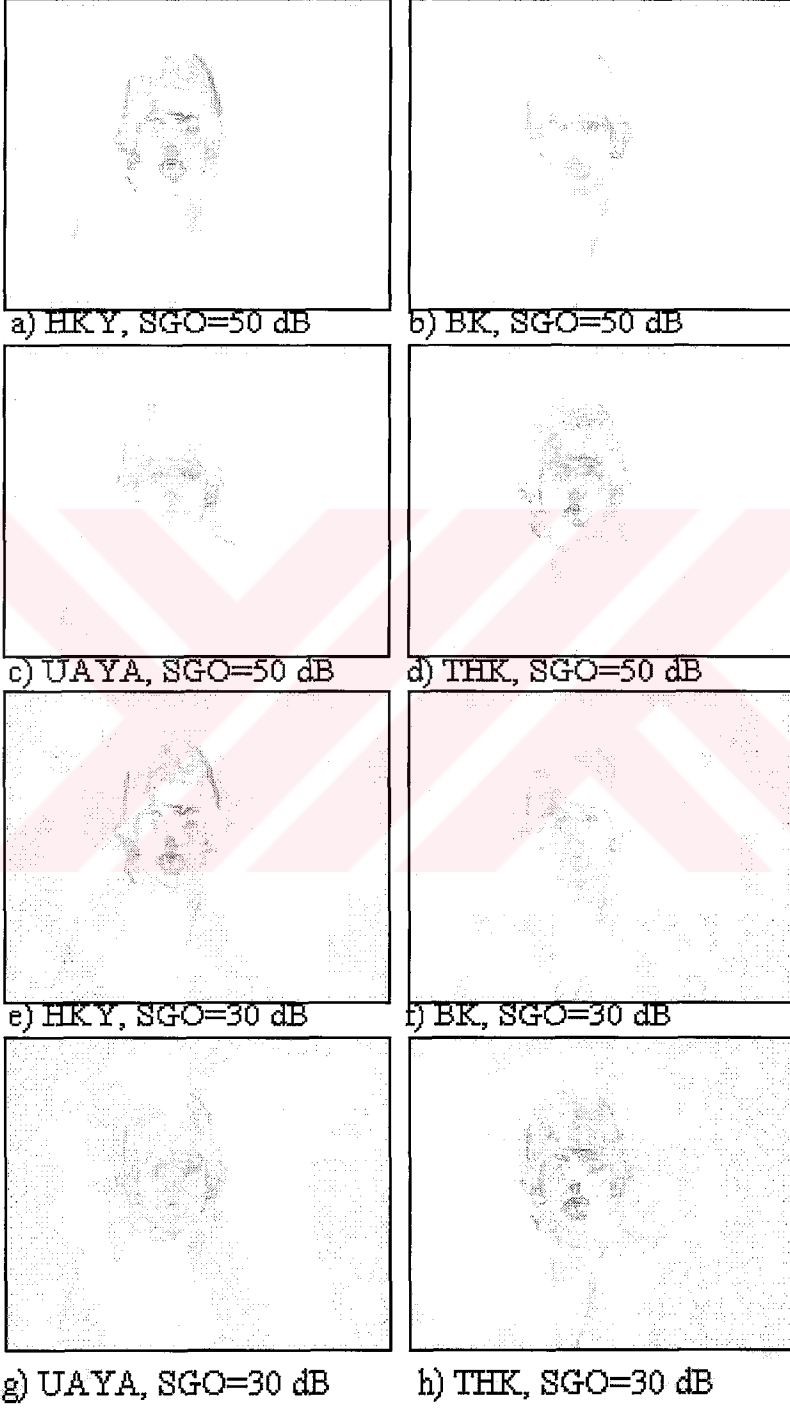
Şekil 46, Şekil 47 ve Şekil 48'de sırasıyla sentetik, "Claire" ve yazarın görüntü dizilerinde değişik hareket kestirme algoritmaları kullanılarak elde edilen fark görüntüleri (YÇF) görülmektedir. Şekillerde HKY: Hareket kestirimi yok, BK: Blok karşılaştırma ile hareket kestirme, UAYA: Üç adım yönlü aramalı blok karşılaştırma ile hareket kestirme ve THK: Türevsel teknikle hareket kestirme anlamında kullanılmıştır. Her üç şekilde de a)-d) görüntüleri için sinyal/gürültü oranı 50 dB, e)-g) görüntüleri için sinyal/gürültü oranı 30 dB olarak seçilmiştir.

Görüntülerin beyaz zemin üzerinde gözükmesi için fark görüntülerindeki parlaklık değerlerinin genlikleri alınarak 255 sayısından çıkarılmıştır ($255 - |I_{YÇF}|$).

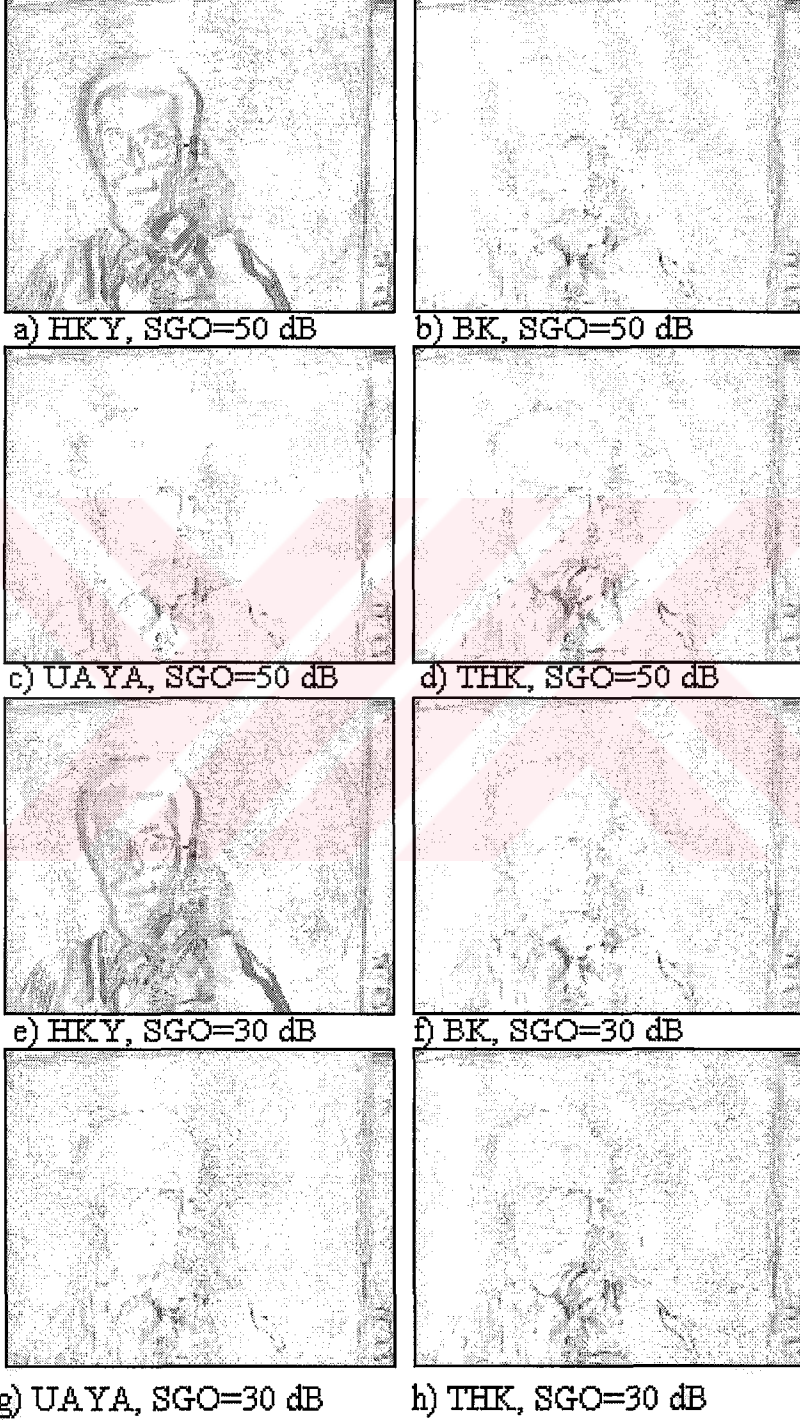
Sentetik görüntü dizisinde sağ alt köşedeki cisimlerin hızları ani ve düzensiz değişmektedir. Bu nedenle türevsel teknik görüntünün bu bölgesinde iyi sonuç vermemektedir. Üç görüntü dizisinde de yine en iyi sonucu blok karşılaştırma tekniklerinin verdiği görülmektedir. Gürültü miktarının artmasıyla her üç görüntü dizisi için de fark görüntülerindeki hataların arttığı görülmektedir.



Şekil 46. Bazı fark görüntüleri (sentetik görüntü dizisi, 7. çerçeve-4. çerçeve).



Şekil 47. Bazı fark görüntüleri ("Claire" görüntü dizisi, 7. çerçeve-4. çerçeve)



Şekil 48. Bazı fark görüntüleri (yazarın görüntü dizisi, 7. çerçeve-4. çerçeve).

3.2.6. Değişik Yöntemlerin İşlem Sayısı Bakımından İncelenmesi

Tablo 11'de değişik hareket kestirme yöntemlerinde bir QCIF formatındaki (176x144) bir görüntü çerçevesi parlaklık değerlerinin kestirimi için gereken ortalama işlem miktarı verilmektedir. Tabloda görüldüğü gibi, sıralı aramalı blok karşılaştırma ve faz ilişkisi teknikleri diğerlerine göre daha fazla işlem gerektirmektedirler. Önerilen yeni yöntemde ise üç adımlı yönlü aramada yapılan işlem sayısına ek olarak 15 toplama işlemi eklenmektedir.

Tablo 11. Değişik hareket kestirme yöntemlerinde 16x16 boyutlu bir bölgenin kestirimi için maksimum işlem sayısı

	OMH		OKH	
	Toplama	Çarpma	Toplama	Çarpma
SBK(±7)	57600	0	57600	57600
UAYA(±6)	6400	0	6400	6400
LA(±5)	5376	0	5376	5376
DLA(±7)	4096	0	4096	4096
BSBA(±7)	5120	0	5120	5120
EYA(±7)	4352	0	4352	4352
DA(±7)	3072	0	4352	3072
ÇA(±8)	4352	0	4352	4352
Önerilen Yöntem	6400+15	0	6400+15	6400
	Toplama	Çarpma		
THK[38]	1028	8		
KÖ[53]	2816	1356		
Faz[56]	6656	6144		

Tabloda sol tarafta hareket kestirme yönteminin adı ve hareket vektörlerinin yatay ve dikey bileşenlerinin maksimum uzunlukları verilmektedir. Tabloda kullanılan ve daha önce tanımlanmamış olan kısaltmaların anlamları aşağıdaki gibidir:

KÖ : Konum özyineli hareket kestirme tekniği.

Faz : Faz ilişkisi tekniğiyle hareket kestirme.

İşlem sayıları ortalama mutlak hata (OMH) ve ortalama kare hata (OKH) ölçütleri için ayrı ayrı verilmiştir. Tablodaki son üç yöntem için OMH veya OKH ölçütü kullanılmadığından bu yöntemler tabloda ayrı olarak gösterilmişlerdir. Tabloda görüldüğü gibi, hızlı arama tekniği kullanılan blok karşılaştırmalarda OMH ölçütü

kullanıldığında çarpma işlemi yapılmamaktadır. Bu nedenle işlem hızı yüksektir. OKH ölçütünde ise toplama sayısı kadar çarpma işlemi yapılmaktadır. Türevsel, konum özyineli ve faz ilişkisi tekniklerinde ise hem toplama hem de çarpma işlemi yapılmaktadır. En az işlem gerektiren yöntem ise OMH ölçütü kullanan dikgen arama yöntemidir. En fazla işlem gerektiren yöntemler ise OKH ölçütü kullanan sıralı aramalı blok karşılaştırma ve faz ilişkisi tekniğidir. Türevsel ve konum özyineli hareket kestirim yöntemlerinde işlem sayısı az görülmesine rağmen, ince duyarlılıkta (bir görüntü elemanı boyundan daha az) hareket kestirmek istendiğinde yapılan tekrar sayıları artacağı için işlem miktarı tekrar sayısı ile orantılı olarak bir kaç kat artar.

3.2.7. Sonuçlar

Hareket kestirme performansı bakımından sıralı aramalı blok karşılaştırmanın diğer hızlı aramalı ve türevsel yöntemlerden daha iyi sonuç verdiği görülmektedir. Sentetik görüntülerde, özellikle düzgün olmayan hareketlerin olduğu bölgelerde türevsel yöntemlerin zayıf performans gösterdiği gözlenmiştir. Hızlı blok karşılaştırma yöntemleri birbirlerine yakın performans göstermekte, üç adımlı yönlü arama ve düzeltilmiş logaritmik arama daha iyi sonuç vermektedir. İşlem sayısı bakımından ise blok karşılaştırma tekniklerinde OMH ölçütü iyi sonuç vermektedir. Hızlı arama tekniği kullanan blok karşılaştırmalarda işlem sayısının çok az olduğu görülmektedir.

3.3. Önerilen Yeni Hareket Kestirme Yönteminin Kodlamadaki Performansının Karşılaştırmalı Olarak İncelenmesi

Daha önceki incelemelerde, hızlı arama tekniği kullanan blok karşılaştırma yönteminin az işlem sayısı gerektirdiği ve hareket kestirme performansının iyi olduğu gözlenmişti. Bu nedenle önerilen yöntem, en iyi sonuçları sağlayan blok karşılaştırma yöntemleriyle karşılaştırılmaktadır. Buradaki blok karşılaştırma işleminde üç adımlı yönlü arama tekniği kullanılmaktadır.

Önce, blok karşılaştırma yöntemiyle hareket kestirme sonucunda YÇF sinyalinin ve hareket vektörlerinin entropisi hesaplanmakta, daha sonra kodlama işlemi sonundaki veri hızı hesaplanmaktadır. Ayrıca görsel sonuçlar verilmektedir.

3.3.1. Yöntemin Veri Azaltma Performansının İncelenmesi

Veri azaltma performansı iki ayrı şekilde hesaplanmaktadır. İlk olarak, hareket kestirme sonunda YÇF sinyalinin ve hareket vektörlerinin entropisi teorik olarak hesaplanmaktadır. İkinci olarak YÇF sinyalinin ayrık kosinüs dönüşümünde

ortaya çıkan katsayıların ve hareket vektörlerinin istatistiksel kodlanmasında ortaya çıkan veri hızı deneysel olarak incelenmektedir.

3.3.1.1. Toplam Entropinin Hareket Vektörleri ve YÇF Sinyali Tarafından Paylaşımı

Tablo 12, Tablo 13, Tablo 14, Tablo 15 ve Tablo 16'da sırasıyla sentetik, "Claire", "Miss America", "Trevor" ve yazarın görüntü dizileri için değişik hareket kestirme yöntemleri sonucunda seviyelendirilmemiş YÇF sinyali ve hareket vektörlerinin entropisi verilmektedir. Tablo 12' de sentetik görüntü dizisi için YÇF entropisi hareket vektörleri entropisinden çok daha fazla olduğu görülmektedir. 4x4 boyutlu blok kullanılması durumunda bile hareket vektörleri entropisi oran olarak azdır. Ancak YÇF sinyali bu haliyle kodlanmayıp AKD işleminden sonra kodlanırsa YÇF entropisi yaklaşık on kat daha azalır ve bu oran büyür (Bölüm 5.4.1.2).

Bu çalışmada incelenen yeni yöntem, tablolarda "YENİ" adıyla belirtilmiştir.

Tablo 12. YÇF ve Hareket vektörlerinin toplam entropisi (Sentetik görüntü dizisi).
E: Entropi (bit / piksel). E_f : YÇF entropisi. E_h : Hareket vektörleri entropisi.
 E_t : Toplam entropi.

BLOK BOY.	E	ÇERÇEVE NUMARASI									
		1	4	7	10	13	16	19	22	25	
4x4	E_f	1.0826	0.2934	0.1905	0.2534	0.2027	0.1935	0.3234	0.2332	0.2886	
	E_h	_____	0.1218	0.1072	0.1174	0.1125	0.1093	0.1332	0.1150	0.1193	
	E_t	1.0826	0.4152	0.2777	0.3708	0.3152	0.3028	0.4566	0.3482	0.4079	
8x8	E_f	1.0826	0.6134	0.4674	0.5012	0.4824	0.4701	0.5928	0.4986	0.5212	
	E_h	_____	0.0401	0.0370	0.0392	0.0380	0.0375	0.0412	0.0386	0.0399	
	E_t	1.0826	0.6535	0.5044	0.5404	0.5204	0.5076	0.6340	0.5372	0.5611	
16x16	E_f	1.0826	1.2575	1.3720	1.2212	1.1824	1.1726	1.3214	1.1870	1.2216	
	E_h	_____	0.0118	0.0125	0.0104	0.0095	0.0112	0.0146	0.0098	0.0122	
	E_t	1.0826	1.2693	1.1815	1.2316	1.1919	1.1838	1.1360	1.1968	1.2338	
YENİ	E_f	1.0826	0.6134	0.4674	0.5012	0.4824	0.4701	0.5928	0.4986	0.5212	
	E_h	_____	0.0123	0.0168	0.0156	0.0215	0.0134	0.0136	0.0124	0.0195	
	E_t	1.0826	0.6257	0.4842	0.5168	0.5039	0.4835	0.6064	0.5110	0.5407	

Tablo 13. YÇF ve Hareket vektörlerinin toplam entropisi ("Claire" görüntü dizisi).
E: entropi (bit / piksel). E_f : YÇF entropisi. E_h : Hareket vektörleri entropisi.
 E_t : Toplam entropi.

BLOK BOY.	E	ÇERÇEVE NUMARASI								
		1	4	7	10	13	16	19	22	25
4x4	E_f	6.3920	2.2520	2.2411	2.0788	2.1251	2.0101	2.2389	2.0580	2.1751
	E_h	_____	0.1443	0.1301	0.1280	0.1284	0.1226	0.1509	0.1258	0.1375
	E_t	6.3920	2.3963	2.2492	2.2069	2.2535	2.1326	2.3897	2.1838	2.3126
8x8	E_f	6.3920	2.4943	2.3242	2.2601	2.1729	2.2928	2.4472	2.2229	2.3654
	E_h	_____	0.0185	0.0136	0.0136	0.0111	0.0118	0.0223	0.0125	0.0183
	E_t	6.3920	2.5128	2.3108	2.2737	2.1840	2.3045	2.4695	2.2354	2.3837
16x16	E_f	6.3920	2.5895	2.3582	2.3167	2.2082	2.3305	2.5335	2.2722	2.3989
	E_h	_____	0.0030	0.0019	0.0018	0.0010	0.0012	0.0033	0.0009	0.0023
	E_t	6.3920	2.5926	2.3601	2.3185	2.2092	2.3317	2.1368	2.2731	2.4013
YENİ	E_f	6.3920	2.4943	2.2972	2.2601	2.1729	2.2928	2.4472	2.2229	2.3654
	E_h	_____	0.0070	0.0055	0.0058	0.0048	0.0052	0.0074	0.0056	0.0071
	E_t	6.3920	2.5013	2.3027	2.2659	2.1777	2.2980	2.4547	2.2285	2.3725

Tablo 14. YÇF ve Hareket vektörlerinin toplam entropisi ("Miss America" görüntü dizisi). E: Entropi (bit/piksel). E_f : YÇF entropisi. E_h : Hareket vektörleri entropisi. E_t : Toplam entropi.

BLOK BOY.	E	ÇERÇEVE NUMARASI								
		1	4	7	10	13	16	19	22	25
4x4	E_f	5.5829	2.5488	2.5904	2.4395	2.4215	2.4970	2.4801	2.4490	2.4523
	E_h	_____	0.2097	0.2070	0.1932	0.1904	0.1964	0.2022	0.1921	0.1887
	E_t	5.5829	2.7585	2.7975	2.6327	2.6119	2.6934	2.6824	2.6411	2.6410
8x8	E_f	5.5824	2.8692	2.9218	2.7065	2.6857	2.7883	2.7551	2.6992	2.7116
	E_h	_____	0.0427	0.0397	0.0344	0.0331	0.0352	0.0371	0.0347	0.0310
	E_t	5.5824	2.9118	2.9616	2.7409	2.7189	2.8235	2.7922	2.7339	2.7426
16x16	E_f	5.5824	3.0053	3.0395	2.7964	2.7757	2.8878	2.8692	2.7891	2.8126
	E_h	_____	0.0063	0.0058	0.0043	0.0035	0.0044	0.0055	0.0043	0.0039
	E_t	5.5824	3.0116	3.0454	2.8006	2.7792	2.8922	2.8747	2.7934	2.8165
YENİ	E_f	5.5824	2.8692	2.9218	2.7065	2.6857	2.7883	2.7551	2.6992	2.7116
	E_h	_____	0.0115	0.0110	0.0106	0.0107	0.0101	0.0104	0.0106	0.0101
	E_t	5.5824	2.8807	2.9328	2.7171	2.6964	2.7984	2.7655	2.7099	2.7217

Tablo 15. YÇF ve Hareket vektörlerinin toplam entropisi ("Trevor" görüntü dizisi).
E: Entropi (bit / piksel). E_f : YÇF entropisi. E_h : Hareket vektörleri entropisi.
 E_t : Toplam entropi.

BLOK BOY.	E	ÇERÇEVE NUMARASI								
		1	4	7	10	13	16	19	22	25
4x4	E_f	6.1801	2.9058	3.1085	3.1477	3.1035	3.1613	3.3132	3.2043	3.0507
	E_h	_____	0.0905	0.1230	0.1372	0.1305	0.1178	0.1247	0.1062	0.0799
	E_t	6.1801	2.9963	3.2316	3.2857	3.2340	3.2791	3.4378	3.3105	3.1307
8x8	E_f	6.1801	3.0982	3.4122	3.4695	3.3789	3.4700	3.6725	3.4786	3.2544
	E_h	_____	0.0170	0.0282	0.0330	0.0311	0.0258	0.0296	0.0236	0.0162
	E_t	6.1801	3.1152	3.4404	3.5025	3.4100	3.4958	3.7021	3.5022	3.2705
16x16	E_f	6.1801	3.2823	3.6467	3.7961	3.6859	3.7425	3.9246	3.6670	3.4042
	E_h	_____	0.0038	0.0063	0.0079	0.0076	0.0061	0.0069	0.0048	0.0033
	E_t	6.1801	3.2861	3.6530	3.8041	3.6935	3.7487	3.9315	3.6719	3.4075
YENİ	E_f	6.1801	3.0982	3.4106	3.4695	3.3789	3.4700	3.6725	3.4786	3.2544
	E_h	_____	0.0066	0.0101	0.0117	0.0107	0.0098	0.0109	0.0093	0.0068
	E_t	6.1801	3.1044	3.4208	3.4811	3.3896	3.4798	3.6834	3.4880	3.2612

Tablo 16. YÇF ve Hareket vektörlerinin toplam entropisi (yazarın görüntü dizisi).
E: Entropi (bit / piksel). E_f : YÇF entropisi. E_h : Hareket vektörleri entropisi.
 E_t : Toplam entropi.

BLOK BOY.	E	ÇERÇEVE NUMARASI								
		1	4	7	10	13	16	19	22	25
4x4	E_f	6.9768	4.0070	4.7797	4.5213	4.9516	4.8985	4.2061	3.7586	3.5268
	E_h	_____	0.1383	0.2216	0.2013	0.2364	0.1903	0.1440	0.1340	0.1018
	E_t	6.9768	4.1453	5.0014	4.7226	5.1880	5.0887	4.3501	3.8926	3.6286
8x8	E_f	6.9768	4.2005	5.1565	4.8236	5.1942	5.2803	4.4981	4.0070	3.7001
	E_h	_____	0.0284	0.0501	0.0438	0.0514	0.0420	0.0312	0.0304	0.0208
	E_t	6.9768	4.2289	5.1866	4.8674	5.2456	5.3223	4.5294	4.0374	3.7209
16x16	E_f	6.9768	4.3917	5.3973	5.0170	5.5064	5.5529	4.7496	4.2301	3.8737
	E_h	_____	0.0068	0.0118	0.0106	0.0092	0.0085	0.0071	0.0076	0.0046
	E_t	6.9768	4.3984	5.4091	5.0275	5.5157	5.5614	4.7567	4.2377	3.8783
YENİ	E_f	6.9768	4.2005	5.1365	4.8236	5.1992	5.2803	4.4981	4.0070	3.7001
	E_h	_____	0.0102	0.0132	0.0121	0.0139	0.0129	0.0112	0.0110	0.0071
	E_t	6.9768	4.2107	5.1497	4.8358	5.2081	5.2933	4.5093	4.0179	3.7072

"Claire" ve "Miss America" görüntü dizilerindeki hareketler çok yavaş, "Trevor" görüntü dizisinde biraz daha fazla, yazarın dizisinde ise daha fazladır. Tablolarda, fazla hareket içeriği olan görüntü dizileri için YÇF entropisinin ve hareket vektörleri entropisinin arttığı görülmektedir. İncelenen bütün görüntü dizileri için blok boyutları büyük seçildiğinde YÇF entropisinin arttığı, hareket vektörleri entropisinin azaldığı görülmektedir.

Tablolardan da görüldüğü gibi, önerilen yeni algoritmada entropi değerleri 8x8 ve 16x16 boyutlu blok kullanılması durumundan daha azdır.

3.3.1.2. AKD Katsayılarının ve Hareket Vektörlerinin İstatistiksel Kodlanması Sonucunda Ortaya Çıkan Veri Hızları

Tablo 17, Tablo 18, Tablo 19, Tablo 20 ve Tablo 21'de 8x8 hızlı ayırık kosinüs dönüşümü alınmış ve seviyelendirilmiş katsayıların kodlanmasıyla ortaya çıkan veri hızları gösterilmektedir. Ayrıca hareket vektörlerinin kodlanmasında ortaya çıkan bit miktarları ve toplam bit miktarları verilmektedir. Burada toplam data hızı çok düşük düzeylere indirildiği için hareket vektörlerinin kodlanmasında ortaya çıkan bit miktarı artık etkili olmaya başlamıştır. Özellikle 4x4 boyutlu bloklarla hareketler kestirildiğinde sadece hareket vektörleri için ortalama 20-50 kbit/sn arasında değişen veri hızları ortaya çıkmaktadır. Sabit bloklarla hareket kestirimi yapıldığında en iyi sonuç 8x8 ve 16x16 boyutlu bloklar kullanarak elde edilmektedir. Değişken boyutlu bloklar kullanıldığında ise (tablolarda YENİ olarak gösterilmiştir) hareket vektörleri entropisinin azaldığı ve toplam data hızının azaldığı görülmektedir.

Tablo 17. Seviyelendirilmiş AKD katsayılarının ve hareket vektörlerinin kodlanması sonucu ortaya çıkan veri hızları (sentetik görüntü dizisi). B_f : Veri hızı (bit/piksel). B_h : AKD katsayılarının iletimi için gereken veri hızı. B_t : Hareket vektörlerinin iletimi için gereken veri hızı. B_t : Toplam veri hızı.

BLOK BOY.	E	ÇERÇEVE NUMARASI								
		1	4	7	10	13	16	19	22	25
4x4	B_f	0.2667	0.0161	0.0000	0.0245	0.0289	0.0356	0.0178	0.0215	0.0342
	B_h	_____	0.3451	0.2840	0.4234	0.4523	0.4612	0.3513	0.3785	0.4321
	B_t	0.2667	0.3611	0.2840	0.4479	0.4812	0.4968	0.3691	0.4000	0.4663
8x8	B_f	0.2667	0.0341	0.0123	0.0426	0.0512	0.0624	0.0357	0.0405	0.0532
	B_h	_____	0.1078	0.0886	0.1342	0.1423	0.1535	0.1126	0.1224	0.1445
	B_t	0.2667	0.1420	0.1009	0.1768	0.1935	0.2159	0.1483	0.1629	0.1977
16x16	B_f	0.2667	0.0899	0.0742	0.1135	0.1235	0.1432	0.0934	0.0986	0.1341
	B_h	_____	0.0343	0.0289	0.0387	0.0432	0.0521	0.0377	0.0402	0.0415
	B_t	0.2667	0.1242	0.1031	0.1512	0.1667	0.1953	0.1311	0.1388	0.1756
YENİ	B_f	0.2667	0.0341	0.0123	0.0426	0.0512	0.0624	0.0357	0.0405	0.0532
	B_h	_____	0.0737	0.0629	0.0532	0.0683	0.0532	0.0513	0.0492	0.0734
	B_t	0.2667	0.1078	0.0752	0.0958	0.1195	0.1156	0.0870	0.0897	0.1266

Tablo 18. Seviyelendirilmiş AKD katsayılarının ve hareket vektörlerinin kodlanması sonucu ortaya çıkan veri hızları ("Claire" görüntü dizisi). B : Veri hızı (bit/piksel).

BLOK BOY.	E	ÇERÇEVE NUMARASI								
		1	4	7	10	13	16	19	22	25
4x4	B_f	0.2915	0.0329	0.0227	0.0204	0.0129	0.0218	0.0321	0.0197	0.0236
	B_h	_____	0.3080	0.2799	0.2805	0.2711	0.2784	0.3202	0.2754	0.2989
	B_t	0.2915	0.3409	0.3026	0.3010	0.2840	0.3003	0.3523	0.2951	0.3225
8x8	B_f	0.2915	0.0417	0.0264	0.0293	0.0171	0.0290	0.0404	0.0267	0.0278
	B_h	_____	0.0482	0.0426	0.0423	0.0394	0.0402	0.0530	0.0411	0.0278
	B_t	0.2915	0.0899	0.0690	0.0716	0.0565	0.0692	0.0936	0.0678	0.0760
16x16	B_f	0.2915	0.0569	0.0363	0.0381	0.0200	0.0337	0.0492	0.0296	0.0303
	B_h	_____	0.0103	0.0092	0.0090	0.0084	0.0085	0.0107	0.0084	0.0095
	B_t	0.2915	0.0672	0.0455	0.0471	0.0283	0.0422	0.0599	0.0380	0.0399
YENİ	B_f	0.2915	0.0417	0.0264	0.0293	0.0171	0.0290	0.0406	0.0267	0.0278
	B_h	_____	0.0288	0.0119	0.0120	0.0091	0.0106	0.0150	0.0118	0.0110
	B_t	0.2915	0.0704	0.0283	0.0413	0.0263	0.0396	0.0555	0.0385	0.0388

Tablo 19. Seviyelendirilmiş AKD katsayılarının ve hareket vektörlerinin kodlanması sonucu ortaya çıkan veri hızları ("Miss America" görüntü dizisi). B: Veri hızı (bit/piksel).

BLOK BOY.	E	ÇERÇEVE NUMARASI								
		1	4	7	10	13	16	19	22	25.
4x4	B _f	0.2088	0.0145	0.0215	0.0124	0.0156	0.0116	0.0137	0.0143	0.0161
	B _h	_____	0.4815	0.4765	0.4415	0.4249	0.4372	0.4596	0.4320	0.4205
	B _t	0.2088	0.4960	0.4980	0.4540	0.4405	0.4487	0.4733	0.4463	0.4366
8x8	B _f	0.2088	0.0213	0.0327	0.0185	0.0211	0.0190	0.0167	0.0182	0.0219
	B _h	_____	0.0903	0.0841	0.0740	0.0720	0.0756	0.0793	0.0750	0.0679
	B _t	0.2088	0.1116	0.1160	0.0926	0.0931	0.0946	0.0960	0.0931	0.0898
16x16	B _f	0.2088	0.0293	0.0389	0.0234	0.0253	0.0245	0.0245	0.0212	0.0274
	B _h	_____	0.0149	0.0139	0.0118	0.0108	0.0119	0.0138	0.0116	0.0116
	B _t	0.2088	0.0442	0.0528	0.0352	0.0360	0.0364	0.0384	0.0328	0.0389
YENİ	B _f	0.2088	0.0213	0.0327	0.0185	0.0211	0.0190	0.0167	0.0182	0.0219
	B _h	_____	0.0160	0.0152	0.0127	0.0199	0.0129	0.0150	0.0138	0.0130
	B _t	0.2088	0.0373	0.0479	0.0312	0.0410	0.0319	0.0317	0.0310	0.0349

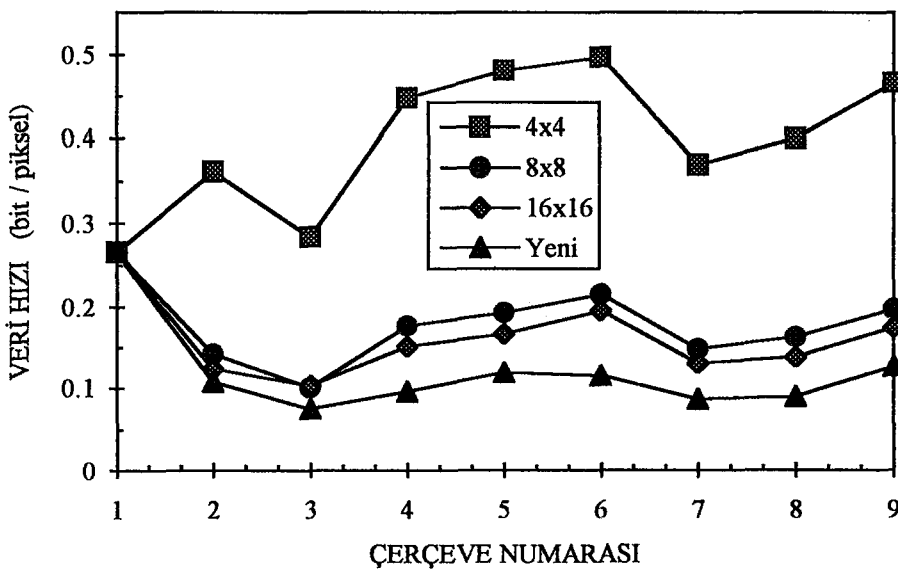
Tablo 20. Seviyelendirilmiş AKD katsayılarının ve hareket vektörlerinin kodlanması sonucu ortaya çıkan veri hızları ("Trevor" görüntü dizisi). B: Veri hızı (bit/piksel).

BLOK BOY.	E	ÇERÇEVE NUMARASI								
		1	4	7	10	13	16	19	22	25
4x4	B _f	0.3284	0.0323	0.0572	0.0589	0.0513	0.0663	0.0932	0.0750	0.0539
	B _h	_____	0.2170	0.2740	0.3135	0.2862	0.2631	0.2874	0.2447	0.2020
	B _t	0.3284	0.2493	0.3311	0.3724	0.3374	0.3294	0.3806	0.3198	0.2560
8x8	B _f	0.3284	0.0559	0.0947	0.0975	0.0862	0.1052	0.1311	0.1053	0.0749
	B _h	_____	0.0457	0.0635	0.0768	0.0695	0.0604	0.0691	0.0562	0.0453
	B _t	0.3284	0.1015	0.1581	0.1743	0.1557	0.1656	0.2007	0.1615	0.1202
16x16	B _f	0.3284	0.0722	0.1202	0.1361	0.1277	0.1340	0.1561	0.1203	0.0926
	B _h	_____	0.0109	0.0148	0.0196	0.0169	0.0146	0.0165	0.0125	0.0104
	B _t	0.3284	0.0831	0.1350	0.1557	0.1446	0.1485	0.1727	0.1328	0.1031
YENİ	B _f	0.3284	0.0559	0.0947	0.0975	0.0862	0.1052	0.1316	0.1053	0.0749
	B _h	_____	0.0134	0.0156	0.0214	0.0201	0.0175	0.0182	0.0143	0.0120
	B _t	0.3284	0.0693	0.1103	0.1189	0.1062	0.1227	0.1498	0.1196	0.0869

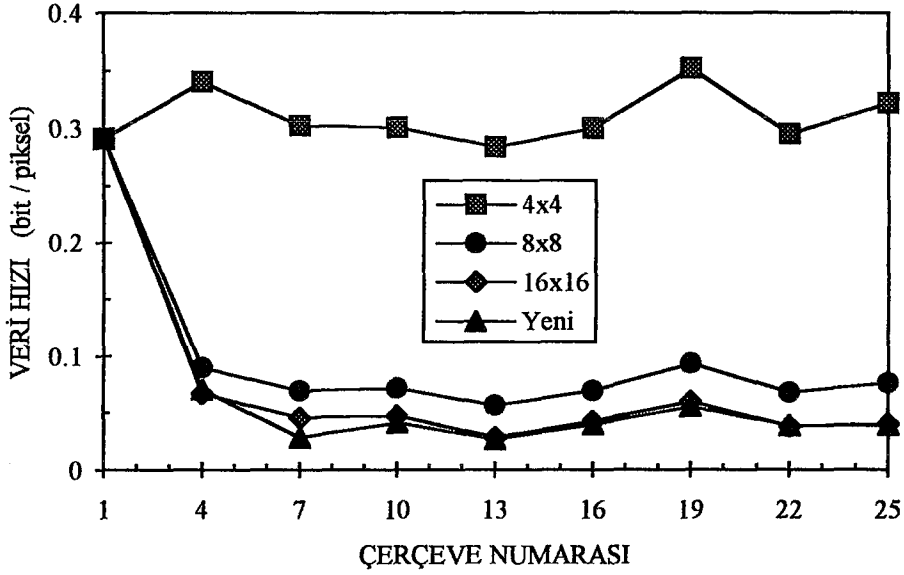
Tablo 21. Seviyelendirilmiş AKD katsayılarının ve hareket vektörlerinin kodlanması sonucu ortaya çıkan veri hızları (yazarın görüntü dizisi). B: Veri hızı (bit/piksel).

BLOK BOY.	E	ÇERÇEVE NUMARASI								
		1	4	7	10	13	16	19	22	25
4x4	B _f	0.5026	0.1131	0.2206	0.1679	0.2674	0.2449	0.1311	0.0996	0.0632
	B _h	_____	0.3141	0.5493	0.4745	0.6377	0.4762	0.3694	0.3385	0.2380
	B _t	0.5026	0.4272	0.7699	0.6424	0.9052	0.7211	0.5005	0.4381	0.3012
8x8	B _f	0.5026	0.1635	0.3043	0.2320	0.3157	0.2972	0.1779	0.1542	0.1016
	B _h	_____	0.0666	0.1175	0.1037	0.1265	0.1091	0.0861	0.0791	0.0538
	B _t	0.5026	0.2301	0.4217	0.3357	0.4422	0.4063	0.2640	0.2333	0.1554
16x16	B _f	0.5026	0.2056	0.3867	0.2891	0.3342	0.3430	0.2278	0.1948	0.1392
	B _h	_____	0.0165	0.0277	0.0254	0.0213	0.0250	0.0207	0.0200	0.0129
	B _t	0.5026	0.2221	0.4144	0.3145	0.3555	0.3681	0.2485	0.2148	0.1522
YENİ	B _f	0.5026	0.1635	0.3043	0.2320	0.3157	0.2972	0.1779	0.1542	0.1016
	B _h	_____	0.0220	0.0314	0.0322	0.0245	0.0281	0.0235	0.0223	0.0201
	B _t	0.5026	0.1855	0.3357	0.2642	0.3402	0.3253	0.2014	0.1765	0.1217

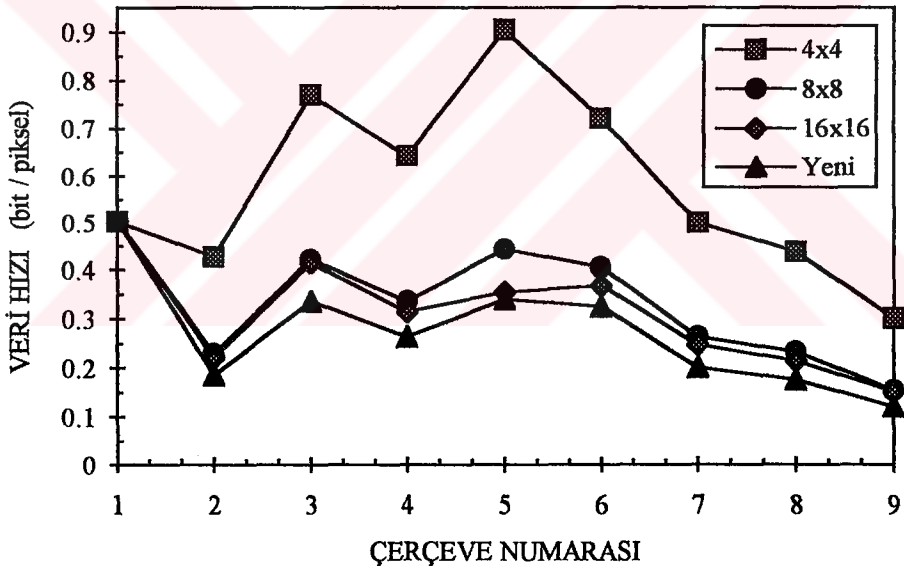
Şekil 49, Şekil 50 ve Şekil 51' de hareket kestiriminde değişik blok boyutları kullanılması durumunda ortaya çıkan toplam veri hızları gösterilmektedir. Bu çalışmadaki yeni yöntem (YENİ) sabit boyutlu bloklar kullanan algoritmalar arasında en iyi sonucu veren (8x8 ve 16x16 boyutlu bloklar) iki yöntemle karşılaştırılmaktadır. Her üç şekilde de YENİ yönteminin en iyi sonucu verdiği görülmektedir.



Şekil 49. Sentetik görüntü dizisinin kodlanmasında ortaya çıkan toplam veri hızı.



Şekil 50. "Claire" görüntü dizisinin kodlanmasında ortaya çıkan toplam veri hızı.



Şekil 51. Yazarın görüntü dizisinin kodlanmasında ortaya çıkan toplam veri hızı.

3.3.2. Kodu Çözülmüş Görüntülerde Kalitenin İncelenmesi

Önerilen yeni yöntemde hareket vektörleri entropisinin daha az olduğu görülmüştü. Bu nedenle, 64 kbit/saniye kapasiteli bir ISDN kanalından sabit veri hızında iletim yapılması için, ayrık kosinüs dönüşümü sırasında ortaya çıkan dönüşüm katsayılarının daha ince seviyelendirilmesi mümkündür. Bu işlem, seviyelendirme

hatasını azaltır. Bunun sonucunda da alıcıda daha kaliteli görüntü elde edilmesi beklenir. Bu bölümde, önerilen algoritmanın sabit iletim hızında alıcı tarafta elde edilen görüntü kalitesi bakımından diğer yöntemlerle karşılaştırmalı olarak incelemesi yapılmıştır. Bu inceleme, kodu çözülen görüntü ile orijinal görüntü arasındaki parlaklık fark değerlerinin değişintisi (hatanın değişintisi) ve kodu çözülen görüntüler için sinyal/gürültü oranının hesaplanması olarak iki aşamada yapılmıştır. Ayrıca kodu çözülen görüntülerden bazıları için görsel sonuçlar verilmiştir.

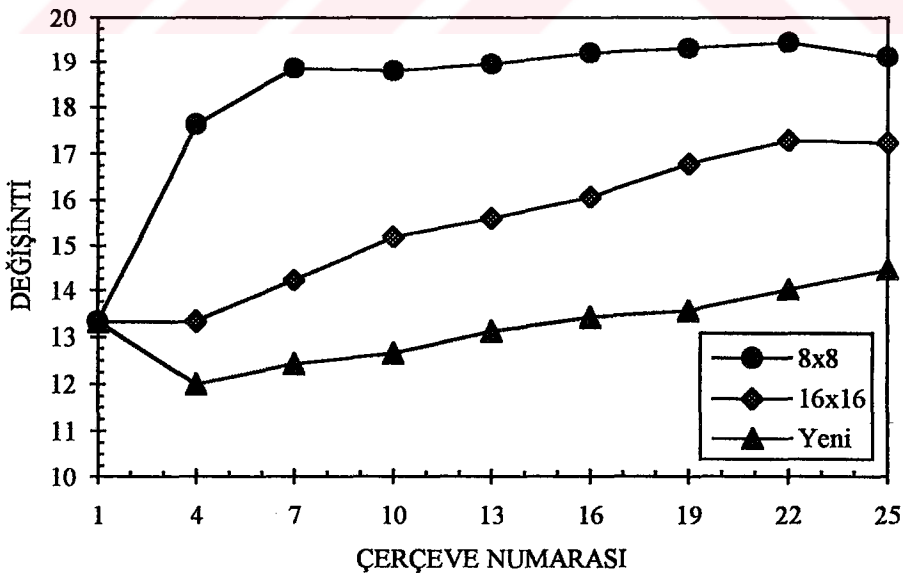
3.3.2.1. Hata Değişintisi

Kodu çözülmüş görüntü ile orijinal görüntü arasındaki hatanın değişintisi,

$$\text{Hatanın değişintisi} = \frac{1}{144 \times 176} \sum_{m=1}^{144} \sum_{n=1}^{176} (I(m,n) - \hat{I}(m,n))^2 \quad (101)$$

olarak hesaplanmaktadır[4]. Burada $\hat{I}(m,n)$: Kodu çözülen görüntü elemanı parlaklık değeridir.

Şekil 52'de "Trevor" dizisi için alıcıda elde edilen görüntüler ile orijinal görüntüler arasındaki hataların değişintileri gösterilmiştir. Şekilde görüldüğü gibi, hata değişintisi azaldıkça sinyal/gürültü oranı artmaktadır.



Şekil 52. Alıcıda elde edilen görüntüler ile orijinal görüntüler arasındaki hataların değişintileri ("Trevor" görüntü dizisi).

Tablo 22'de kod çözülmüş görüntüler ile orijinal görüntüler arasındaki hata değışinti değeri dört ayrı görüntü dizisi için toplu olarak verilmektedir. Tabloda, CL: "Claire", M: "Miss America", T: "Trevor", A: "yazarın görüntü dizisi" sözcüklerinin kısaltılmışı olarak kullanılmıştır.

Tablodan görüldüğü gibi, önerilen algoritma için hata değışinti değeri dört ayrı görüntü dizisi için de daha az çıkmaktadır.

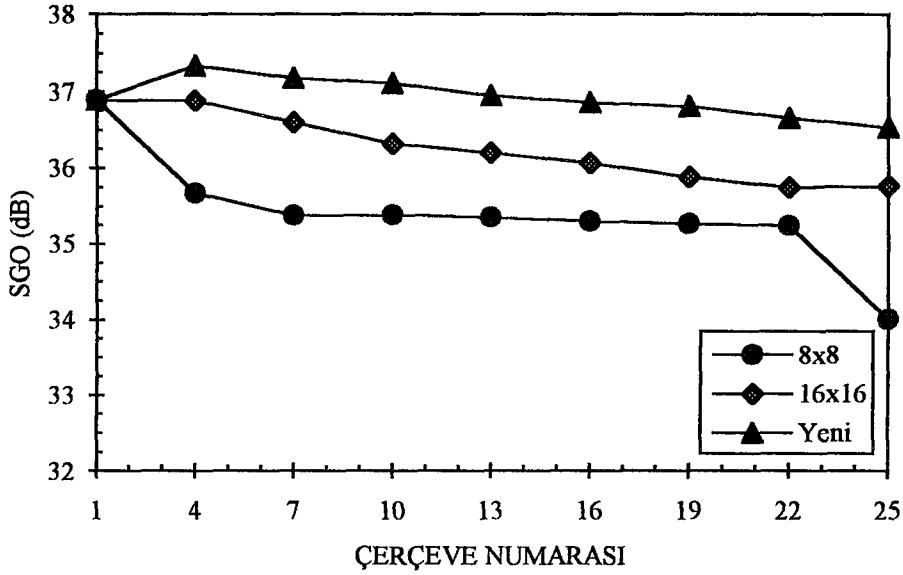
Tablo 22. Kodu çözülmüş görüntüler ile orijinal görüntüler arasındaki hata değışinti (σ^2) değeri.

	BLOK BOY.	ÇERÇEVE NUMARASI								
		1	4	7	10	13	16	19	22	25
CL	8x8	10.92	15.21	16.33	16.98	17.20	17.05	18.06	18.35	18.11
	16x16	10.92	9.47	9.53	9.58	9.79	9.75	9.99	10.10	9.99
	YENİ	10.92	8.37	8.36	8.42	8.51	8.45	8.85	8.95	9.14
M	8x8	7.83	9.73	10.43	9.89	10.52	10.12	10.65	10.24	10.30
	16x16	7.83	7.82	7.97	8.01	8.06	8.05	8.21	8.40	8.48
	YENİ	7.83	6.88	6.79	6.92	6.95	6.85	7.02	7.00	6.93
T	8x8	13.31	17.64	18.85	18.80	18.95	19.21	19.32	19.45	19.12
	16x16	13.31	13.33	14.23	15.19	15.60	16.07	16.79	17.30	17.25
	YENİ	13.31	12.00	12.43	12.65	13.12	13.41	13.55	14.02	14.46
A	8x8	33.86	28.63	27.61	29.60	29.80	29.95	32.10	30.95	29.90
	16x16	33.86	28.58	29.78	29.90	29.46	28.70	29.10	30.76	30.06
	YENİ	33.86	25.20	25.90	26.20	25.99	25.60	25.70	26.80	26.12

3.3.2.2. Sinyal/Gürültü Oranı

Şekil 53 'de "Trevor" dizisi için alıcıda elde edilen görüntülere ait sinyal/gürültü oranları verilmektedir. Sinyal/gürültü oranı aşağıdaki gibi hesaplanmaktadır[4]:

$$SGO = 10 \log_{10} \left(\frac{255^2}{\text{Ha tan ın değışintisi}} \right) \quad (102)$$



Şekil 53. Alıcıda elde edilen görüntülerin sinyal/gürültü oranları ("Trevor" görüntü dizisi).

Tablo 23 'de kodu çözülmüş görüntüler için sinyal/gürültü oranları verilmiştir.

Tablo 23. Kodu çözülmüş görüntülerin sinyal/gürültü oranları (dB).

	BLOK BOY.	ÇERÇEVE NUMARASI									
		1	4	7	10	13	16	19	22	25	
CL	8x8	37.74	36.31	36.00	35.83	35.78	35.81	35.56	35.49	35.92	
	16x16	37.74	38.37	38.34	38.32	38.22	38.24	38.14	38.09	38.14	
	YENİ	37.74	38.90	38.91	38.88	38.83	38.86	38.66	38.61	38.52	
M	8x8	39.19	38.25	37.95	38.18	37.91	38.08	37.86	38.03	38.00	
	16x16	39.19	39.20	39.11	39.10	39.07	39.07	38.99	38.89	38.84	
	YENİ	39.19	39.76	39.81	39.73	39.71	39.77	39.67	39.68	39.72	
T	8x8	36.89	35.67	35.38	35.39	35.35	35.30	35.27	35.24	34.01	
	16x16	36.89	36.88	36.60	36.32	36.20	36.07	35.88	35.75	35.76	
	YENİ	36.89	37.34	37.18	37.11	36.95	36.86	36.81	36.66	36.53	
A	8x8	32.83	33.56	33.72	33.42	33.39	33.37	33.07	33.22	33.37	
	16x16	32.83	33.57	33.39	33.37	33.44	33.55	33.49	32.25	33.35	
	YENİ	32.83	34.12	34.00	33.95	33.98	34.05	34.03	33.85	33.96	

Tablo 23' de değişken blok boyutlu algoritmanın (YENİ) diğer yöntemlerden daha iyi sonuç verdiği görülmektedir. Tablo 23'de "YENİ" olarak numaralanan

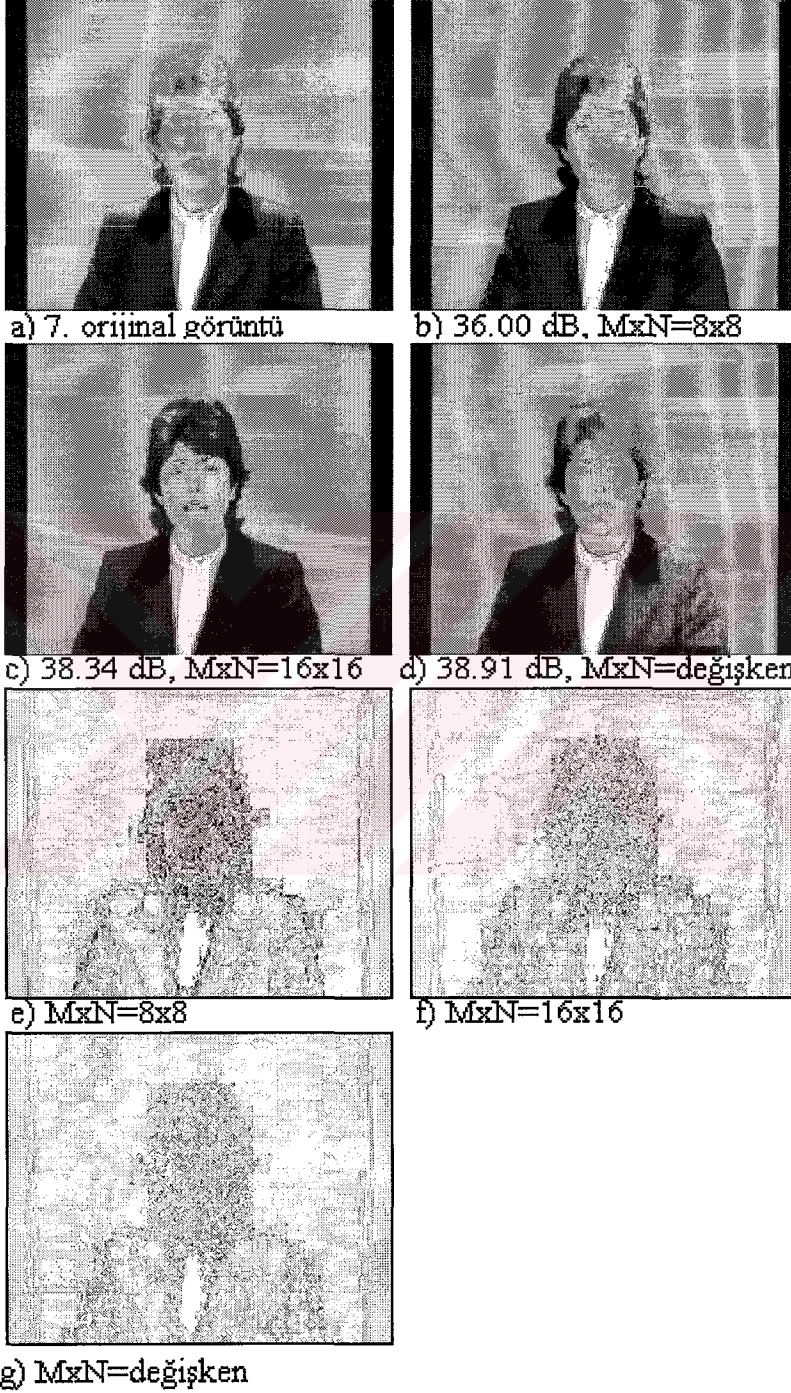
yöntemde alıcıda elde edilen görüntülerin sinyal/gürültü oranlarında ortalama 0 ile 1 dB arasında iyileşmeler görülmüştür.

3.3.2.3. Görsel İncelemeler

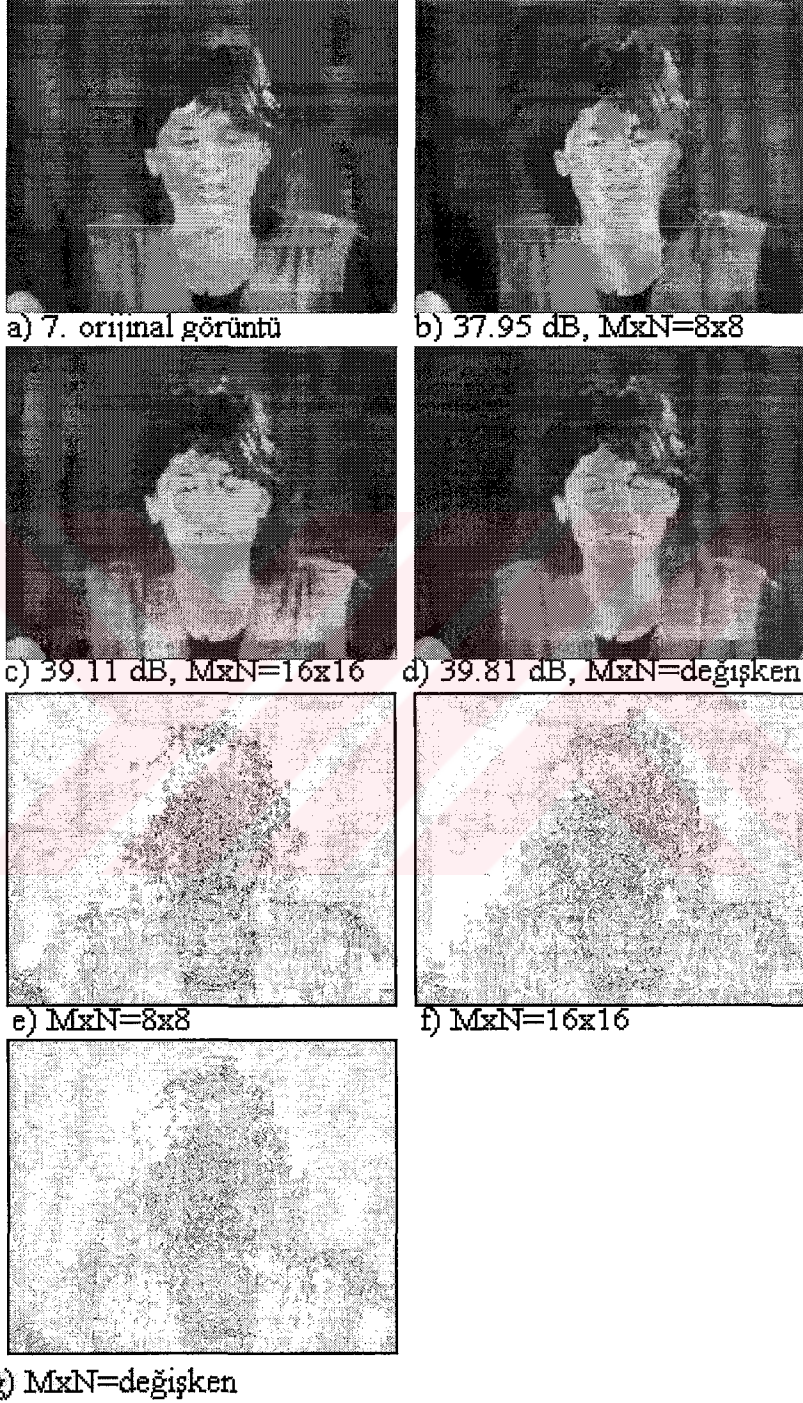
Şekil 54', Şekil 55', Şekil 56', ve Şekil 57'de değişik görüntü dizileri için alıcıda elde edilen bazı görüntüler ve hata görüntüleri gösterilmiştir.

Kodu çözülen görüntüler, alıcı tarafta yeniden elde edilen görüntülerdir. Bu görüntülerin hepsinde sırasıyla aşağıdaki çerçeveler bulunmaktadır: a): orijinal görüntü çerçevesi. b): 8x8 boyutlu sabit bloklar kullanılması durumunda kodu çözülen görüntü. c): 16x16 boyutlu sabit bloklar kullanılması durumunda kodu çözülen görüntü. d): değişken boyutlu bloklar kullanılması durumunda kodu çözülen görüntü. e), f) ve g): sırasıyla 8x8 sabit boyutlu, 16x16 sabit boyutlu ve değişken boyutlu bloklar kullanılması durumunda orijinal görüntüler ile kodu çözülen görüntüler arasındaki parlaklık fark değerlerinin 10 kat kuvvetlendirilmiş görüntüleri.

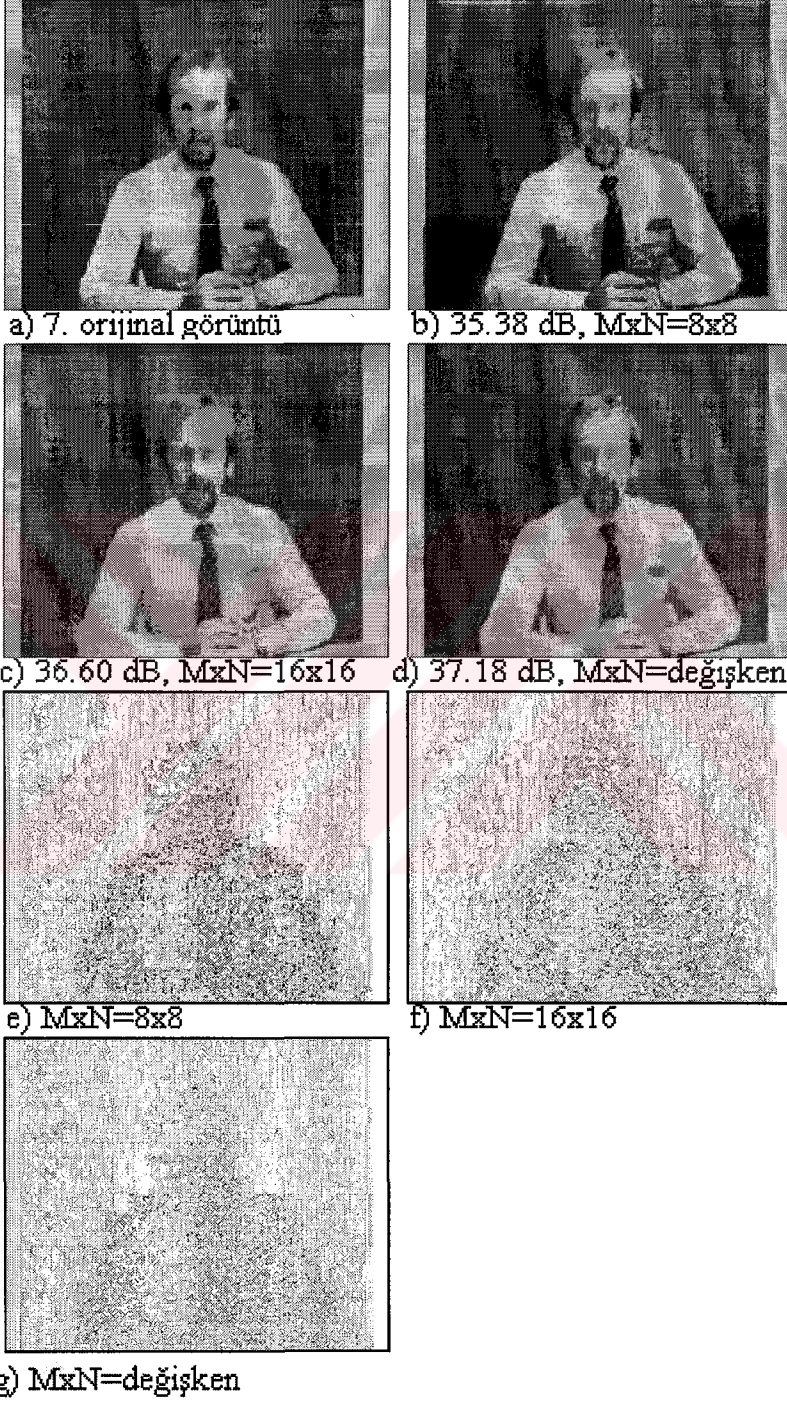
Dört ayrı şekilde de önerilen yöntemin daha iyi sonuç verdiği görülmektedir. Görüntüler bilgisayar yazıcısıyla kağıta aktarılmıştır. Görüntülerin gerçek kalitesi bu nedenle tam olarak görülememektedir. Şekil 54'de b) görüntüsünün S/G oranı 36 dB, c) görüntüsünün S/G oranı 38.34 dB, d) görüntüsünün S/G oranı 38.91 dB'dir. Şekil 55'de b) görüntüsünün S/G oranı 37.95 dB, c) görüntüsünün S/G oranı 39.11 dB, d) görüntüsünün S/G oranı 39.81 dB'dir. Şekil 56'da b) görüntüsünün S/G oranı 35.38 dB, c) görüntüsünün S/G oranı 36.60 dB, d) görüntüsünün S/G oranı 37.18 dB'dir. Şekil 57'de b) görüntüsünün S/G oranı 33.72 dB, c) görüntüsünün S/G oranı 33.39 dB, d) görüntüsünün S/G oranı 34.00 dB'dir.



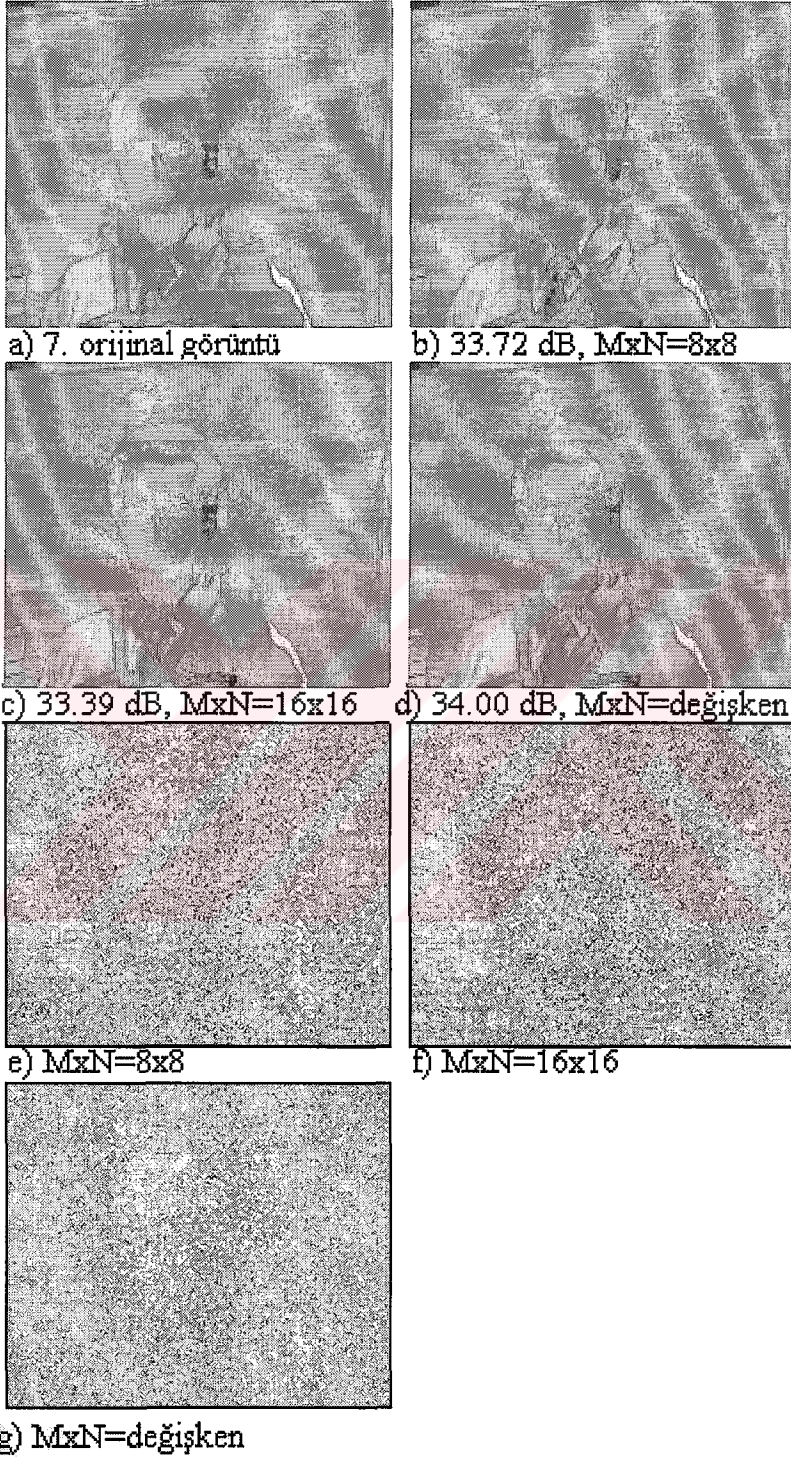
Şekil 54. Alıcıda kodu çözülen görüntüler ve hata görüntüleri. a): Orijinal görüntü. b)-d): Alıcıda kodu çözülen görüntülerin kullanılan blok boyutlarına göre sinyal/gürültü oranları. e)-g): 10 kat kuvvetlendirilmiş hata görüntüleri.



Şekil 55. Alıcıda kodu çözülen görüntüler ve hata görüntüleri. a): Orijinal görüntü. b)-d): Alıcıda kodu çözülen görüntülerin kullanılan blok boyutlarına göre sinyal/gürültü oranları. e)-g): 10 kat kuvvetlendirilmiş hata görüntüleri.



Şekil 56. Alıcıda kodu çözülen görüntüler ve hata görüntüleri. a): Orijinal görüntü. b)-d): Alıcıda kodu çözülen görüntülerin kullanılan blok boyutlarına göre sinyal/gürültü oranları. e)-g): 10 kat kuvvetlendirilmiş hata görüntüleri.



Şekil 57. Alıcıda kodu çözülen görüntüler ve hata görüntüleri. a): Orijinal görüntü. b)-d): Alıcıda kodu çözülen görüntülerin kullanılan blok boyutlarına göre sinyal/gürültü oranları. e)-g): 10 kat kuvvetlendirilmiş hata görüntüleri.

4. İRDELEME VE TARTIŞMA

Türevsel ve konum özyineli teknikler, cisimlerin hareketlerinde düzensiz değişimler olduğu zamanlarda, hareketleri blok karşılaştırma tekniklerinden daha kötü izlemektedirler. Bunun nedeni, daha önce elde edilen hareket vektörlerinin bir sonraki işlemde başlangıç değeri olarak alınmasıdır. İki görüntü çerçevesi arasında cisimlerin hareket yönleri ve hızları ani olarak değiştiğinde, başlangıç değerleri yanlış seçilmiş olmaktadır. Bu, hataların üstüste yığılmasına neden olur. Konum özyineli hareket kestirme tekniklerinde bir görüntü elemanı boyutundan daha küçük yer değiştirmeler de kestirilebildiği için karmaşık görüntülerdeki hareketler bloksal yöntemlerden daha iyi algılanır. Türevsel ve konum özyineli hareket kestirme teknikleriyle hızlı ve düzensiz hareketler içeren karmaşık görüntülerde hareket kestirimi iyi sonuç vermemektedir. Bu tekniklerin daha iyi sonuç vermesi bir ön süzgeçleme yapılarak sağlanabilir.

Faz ilişkisi tekniğinde, kullanılan blok boyutunun yarısı kadar mesafelerdeki hareketler kestirilebilmektedir. Örneğin, 16x16 boyutlu bloklar kullanıldığında ± 8 görüntü elemanı boyuna kadar olan hareketler kestirilebilmektedir. Faz ilişkisi tekniği bir görüntü elemanı boyundan daha küçük mesafeleri algılayacak duyarlıkta kullanılırsa, blok karşılaştırma tekniklerinden daha da iyi sonuç vereceği açıktır. Fakat bu durumda işlem sayısı da artar. Faz ilişkisi tekniğinde, görüntüdeki dar bir tayf bölgesini içeren parlaklık değişimleri ölçmelere etki etmez. Yerdeğiştirme vektörleri, türevsel veya blok karşılaştırma tekniklerinden daha doğru olarak kestirilebilir. Bu tekniğin bir sakıncası, dönmeler ve hızlı ölçeklemeler (fast zooms) bir çok aday vektör ortaya çıkarır ve bu durumda ilişki yüzeyindeki tepeler birbirinden ayırtedilemeyebilir.

Sıralı aramalı blok karşılaştırma ve faz ilişkisi teknikleri diğerlerine göre daha fazla işlem gerektirmektedirler. Özyineli ve türevsel hareket kestirim yöntemlerinde işlem sayısı az görülmesine rağmen, bir görüntü elemanı boyundan daha küçük duyarlıkta hareket kestirilmesi istendiğinde yapılan tekrar sayıları artacağı için işlem miktarı tekrar sayısı ile orantılı olarak bir kaç kat artar.

Önerilen yöntemde işlem hızı 0.0023 kat artmaktadır. Ancak blok boyutları daha geniş sınırlarda değiştirilirse bu oran artar. Buna karşılık veri azaltma performansı da artar.

Hareket kestirme tekniklerinden iyi sonuç alınabilmesi Bölüm 2' de anlatılan varsayımların geçerli olmasıyla ilişkilidir. Ancak, görüntüde sadece yeknesak bir öteleme hareketi olması çoğu zaman geçerli değildir. Örneğin, cisimlerin dönme hareketleri

yapması, cisim boyutlarının büyümesi veya küçülmesi, hareket eden cisimin arka planında yeni görüntülerin belirmesi veya bu cisimin görüntüsünün başka bir cismin görüntüsünün üzerine gelmesi, farklı hızlarda hareket eden çok sayıda cisimlerin olması, cisimlerin hızlarında ve hareket yönlerinde ani değişiklikler olması, cisimlerin parlaklıklarının değişmesi gibi olaylar kestirimin doğruluğunu ve performansını olumsuz etkiler.

Hızlı blok karşılaştırma tekniklerinde, yanlış minimum noktasına yakınsama olduğu zamanlarda veri hızı azaltımı yine de gerçekleşmektedir. Ancak bu durumda alıcıya iletilen hareket vektörü gerçek hareketi temsil etmediği için, eğer alıcıda aradeğerleme işlemleri yapılacaksa yanlış değerler elde edilecektir.

Bu çalışmada renk ve ses bilgisinin kodlaması yapılmamıştır. Ses için 16 kbit/saniyelik veri hızı [5], [11], ayrılmıştır. İki renk fark sinyali ise parlaklık bileşeninin dörtte biri kadar hızda örneklenmektedirler. Bu nedenle renk bilgisi için, parlaklık bileşeni için gereken veri hızının yarısını kullanmak yeterlidir. Renk bileşenleri için hareket kestirme işlemine gerek yoktur [1], [22], [57]. Parlaklık bileşeni için kullanılan hareket vektörü renk bileşeni için de kullanılır. Çalışmada kanal kapasitesinin 32 kbit/saniyelik bölümü renk ve ses bilgisi için ayrılmıştır.

5. SONUÇLAR

Bu çalışmada sayısal video sinyallerinin düşük veri hızlarında iletilmesi için kodlama yöntemleri genel olarak incelenmiş ve hareket kestirimi ve dönüşüm kodlamasını birlikte kullanan melez kodlama yöntemlerinin iyi sonuç verdiği görülmüştür.

Hareket kestirme teknikleri blok karşılaştırma, türevsel, konum özyineli ve faz ilişkisi teknikleri olarak gruplandırılarak detaylı olarak karşılaştırmalı bir şekilde incelenmişlerdir. Yapılan deneyler ve incelemeler sonucunda video telefon sinyallerinin bir ISDN kanalından (64 kbit/sn) iletilmesini amaçlayan ve yeni bir hareket kestirim tekniği kullanan bir kodlama yöntemi önerilmiştir.

Deneylerde QCIF formatında uygun beş çeşit görüntü dizisi üç türe ayrılarak incelenmiştir. Birinci tür görüntü dizisi bilgisayar programı ile oluşturulan sentetik görüntü dizisidir. Bilgisayar benzetimi için kullanılan sentetik görüntülerde değişik boyutlarda ve geometrik şekillerde sentetik cisim görüntüleri değişik yönlerde ve değişik hızlarda hareket ettirilmiştir. Önceden bilinen bu hız ve yön bilgileri yardımıyla hareket kestirme yöntemlerinin çalışmaları test edilmiştir. İkinci türdeki görüntü dizileri üç adet tanınmış standart görüntü dizisidir ("Claire", "Miss America", ve "Trevor"). Üçüncü tür görüntü dizisi ise, bölüm laboratuvarındaki video kamerayla elde edilen bir görüntü dizisidir.

Görüntülere değişik oranlarda Gauss gürültüsü eklenerek hareket kestirme yöntemlerinin gürültüye karşı duyarlılıkları da incelenmiştir.

Blok karşılaştırmada karşılaştırma ölçütleri, karşılaştırma şekilleri ve uygun blok boyutları araştırılmıştır. Küçük blok boyutu ve büyük arama uzaklığı kullanılması durumunda en iyi sonuçların elde edildiği gözlenmiştir. Ancak bu durumda kestirim hatasının entropisi azalırken, hareket vektörlerinin entropisinin arttığı gözlenmiştir. Sabit blok büyüklüğü kullanılması durumunda en uygun blok boyutlarının 8x8 olduğu sonucuna varılmıştır. Daha sonra hızlı blok karşılaştırma teknikleri ve türevsel hareket kestirme teknikleri gürültü eklenmiş görüntüler için incelenmiş ve en iyi sonucu sıralı arama tekniği kullanan blok karşılaştırmaların verdiği görülmüştür. Hızlı arama tekniği kullanan blok karşılaştırmaların sıralı arama tekniği kullanan blok karşılaştırmaya yakın performans gösterdiği, buna karşılık yaklaşık on kat daha az işlem gerektirdiği görülmüştür. Sıralı aramalı blok karşılaştırma ve faz ilişkisi teknikleri diğerlerine göre daha fazla işlem gerektirmektedirler. Hızlı arama tekniklerinden üç adımlı yönlü arama tekniği, kolay uygulanabilir olması ve performansının iyi olması nedeniyle tercih edilmiştir.

Önerilen yeni hareket kestirme yönteminde, kestirim hatasının ve hareket vektörleri entropisinin düşük seviyede olması sağlanmaktadır. Bu yöntemde, kullanılacak olan bütün blok tipleri önceden tanımlanmaktadır. Tanımlanan bütün blok tipleri bir liste halinde küçük bir bellek biriminde saklanmakta ve kodlama sırasında ilgili blok tipleri uyarlamalı olarak seçilmektedir. Alıcıda da vericidekiyle aynı olan bir liste bulunmaktadır. Vericide herhangi bir anda belirlenen blok tipinin adresi alıcıya gönderilmektedir. Alıcıdaki liste taranarak blok çeşidi öğrenilmektedir. Bu yöntemde, listenin boyutlarının küçük olması halinde fazla bir işlem hızı gerekmemektedir.

Veri azaltma performansı iki ayrı şekilde hesaplanmaktadır. İlk olarak, hareket kestirme sonunda YÇF sinyalinin ve hareket vektörlerinin entropisi teorik olarak hesaplanmaktadır. İkinci olarak YÇF sinyalinin ayrık kosinüs dönüşümünde ortaya çıkan katsayıların ve hareket vektörlerinin istatistiksel kodlanmasında ortaya çıkan veri hızı deneysel olarak incelenmiştir. Fazla hareket içeriği olan görüntü dizileri için YÇF entropisinin ve hareket vektörleri entropisinin arttığı görülmüştür. İncelenen bütün görüntü dizileri için blok boyutları büyük seçildiğinde YÇF entropisinin arttığı, hareket vektörleri entropisinin azaldığı görülmüştür.

Bu çalışmada incelenen yeni yöntem diğerlerinden daha iyi performans göstermiştir. Veri hızı, en iyi sonuç veren sabit blok büyüklüğü kullanan yöntemlere göre ortalama %1 ile %6 arasında azalmıştır. En son bölümde, önerilen yöntemin sabit iletim hızında alıcı tarafta elde edilen görüntü kalitesi bakımından diğer yöntemlerle karşılaştırmalı olarak incelemesi yapılmıştır. Bu inceleme, kod çözülen görüntü ile orijinal görüntü arasındaki parlaklık fark değerlerinin değişintisi (hata değişintisi) ve kod çözülen görüntüler için sinyal/gürültü oranının hesaplanması olarak iki aşamada yapılmıştır. Ayrıca kod çözülen görüntülerden bazıları için görsel sonuçlar verilmiştir. Önerilen yöntem için hata değişinti değerleri dört ayrı görüntü dizisi için de daha az çıkmıştır. 64 kbit/saniyelik sabit iletim kanalı kullanılması durumunda ise alıcıdaki görüntü kalitesinde 0 ile 1 dB kadar iyileşmeler elde edilmiştir. Bu iyileşme görsel olarak da kanıtlanmıştır. İşlem hızında ise 0.0023 kat gibi düşük oranda bir artma olmuştur. Bu oran gözardı edilebilecek kadar azdır.

Elde edilen deneysel sonuçlar sabit boyutlu bloksal yöntemlerle elde edilenlerle birlikte tablolar, grafikler ve fotoğraflar yardımıyla gösterilmiştir. Blok boyutlarının değişken seçilmesiyle, sabit bir iletim hızında görüntü kalitesinin daha iyi olduğu veya aynı görüntü kalitesinin daha düşük veri hızlarında elde edildiği gözlenmiştir. Hareket vektörlerinin entropisi ise 8x8 blok boyutu kullanıldığı duruma göre ortalama üç kat azalmıştır.

6. ÖNERİLER

Değişken blok büyüklüğü kullanan hareket kestirme yönteminde, blok sınırlarının değişim aralığını daha büyük tutarak daha iyi sonuçlar elde edilebilir. Ancak bu durumda ortaya çıkacak farklı bölge sayısı üstel olarak artacağı için kullanılacak listenin boyutları büyür ve işlem hızı da artar. Bundan sonraki çalışmalarda bu tür bir inceleme yapılarak işlem hızındaki artma ile elde edilen yararlar arasında bir denge aranabilir.

Bir başka yenilik olarak, saniyedeki görüntü sayısı hareket içeriğine göre belirlenip veri azaltımına katkı getirilebilir. Böylece sadece hareket dengelemesi yapılamayan görüntü çerçevelerini alıcıya iletmek yeterli olabilir.

İyi hareket kestirimi yapıp yapılmadığının test edilmesi de veri hızı azaltımına katkı getirebilir. Bunun için, örneğin blok karşılaştırmada kullanılan hata ifadesi (89) veya uzamsal zamansal kısıtlama yönteminde kullanılan hata ifadesi (48) bazı parlaklık genliği eşikleri ile karşılaştırılır. Hatanın eşiği aştığı bölgelerde güvenilir olmayan hareket kestirimi olduğu belirtilir ve hareket kestirme işlemi buralarda yapılmaz. Böylece gereksiz yere alıcı tarafa hareket vektörleri bilgileri iletilmez.

Günümüzde VLSI teknolojisindeki hızlı gelişmeler sayesinde düşük maliyetli ve yüksek yoğunluklu hızlı VLSI aygıtları ile yüksek hızda paralel/boruhattı (pipeline) işaret işleme pratik ve düşük maliyetli olabilmektedir. Bu nedenle çok işlem gerektiren ve iyi sonuç veren hareket kestirme teknikleri geliştirilen hızlı hesap yöntemleri ve uygun VLSI yapılarıyla desteklenerek gerçek zamanda çalışılabilir.

7. KAYNAKLAR

- [1] Jain, A. K., Fundamentals of Digital Image Processing, First Edition, Prentice - Hall International, Inc., New Jersey, 1989.
- [2] CCITT Recommendation H.261 Video Codec For Audiovisual Services at Px64 kbit/s, Geneva, 1990.
- [3] Dennis, T. ve Ghanbari, M., Data Compression, Image Processing, Ed. Pearson, D., First Edition, McGraw - Hill Book Company, London, 1991.
- [4] Evcı, C. C. ve Waldburger, H., Low Bit - Rate Videophone Codec for ISDN Applications, Image Communications Journal (EURASIP), 1, 10 (1988) 1-51.
- [5] Gharavi, H., Low Bit - Rate Video Transmission for ISDN Application, IEEE Trans. on Circuits and Systems (CAS), 35, 2 (1988) 258- 261.
- [6] Onural, L. ve Çetin, A. E., Sayısal Görüntü ve Video Kodlama Metodları, COST211 ve EUREKA-625 Projeleri Tanıtım Semineri, Temmuz 1992, Bilkent Üniversitesi, Seminer Kitabı, Cilt I, 1-81
- [7] Pereira, F. ve Masera, L., Bitrate Optimization for Fixed Picture Transmissions With ATM Adapted CCITT H.261 Algorithm, Communication Control and Signal Processing, 1, 6 (1990) 1483-1489.
- [8] Sankur, B., Sayısal İmgeler, TMMOB Elektrik Mühendisliği Dergisi, 36, 383 (1991) 287-298.
- [9] Arazaki, S., Image Data Compression by DCT Adaptive Run-Length Coding, IEEE Trans. on Consumer Electronics, 37, 4 (1991) 860-866.
- [10] Yu, Y. B. ve Constantinides, A. G., Variable Block Size and Position Transform Coding, Signal Processing: Theories and Applications, 2, 3 (1988) 1011-1013.
- [11] Bock, G., Bewegtbild-Übertragung im Schmalband - ISDN (64kbit/s), Fernseh - und Kinotechnische Gesellschaft E.V. (FKTG), 2, 3 (1988) 798- 811.

- [12] Netravali, A. N. ve Limb, J. O., Picture Coding: A Review, Proceeding of the IEEE, 68, 3 (1980) 366- 404.
- [13] Türkbeyler, N. ve Constantinides, A. G., Three Dimensional Adaptive Arithmetic Coding for Videoconference, Communication, Control, and Signal Processing, 2, 3 (1990) 1466-1471.
- [14] Pearson, D., Image Processing, Ed. Pearson, D., First Edition, McGraw-Hill Book Company, London, 1991.
- [15] Fukuhava, T. ve Murakami, T., 3-D Motion Estimation of Human Head for Model - Based Image Coding , IEE Proc.-I, 140, 1 (1993) 1-14.
- [16] Wallance, G. K., The JPEG Still Picture Compression Standard, IEEE Trans. on Consumer Electronics, 38, 1 (1992) 18-34.
- [17] Le Gall, D. J., The MPEG Video Compression Algorithm A Review, Image Processing Algorithms and Techniques II (SPIE), 1452, 3 (1991) 444-457.
- [18] Wilson, E. J. ve Tonge, G. J., An Extensible Family of Standards, IBA Technical Review, 16, 3 (1982) 12-25.
- [19] Dubois, E., The Sampling and Reconstruction of Time-Varying Imagery With Application in Video Systems, Proc. of the IEEE, 73, 4 (1985) 502-521.
- [20] Parabhu, K. A. ve Netravali, A. N., Motion Compensated Composite Color Coding, IEEE Trans. on Communications (COM), 31, 2 (1983) 216-223.
- [21] Ngan, K. N., Enhancement of Image Quality for Low Bit Rate Video Coding, IEEE Trans. on Circuits and Systems (CAS), 38, 10 (1991) 1221-1225.
- [22] Wells, N. D., The Bit-rate Reduction of Television Signals for Long Distance Transmission, Digital Television, Ed. Sandbank, C. P., First Edition, John Wiley & Sons Ltd., London, 1990.
- [23] Nagel, H. H., Image Sequence Analysis: What Can We Learn From Applications?, Ed. Huang, T. S., First Edition, Springer-Verlag, New York, 1981.
- [24] Lim, J. S., Two - Dimensional Signal and Image Processing, First Edition, Prentice-Hall International, Inc., New York, 1990.

- [25] Dubois, E. ve Moncet, J.L., Encoding and Progressive Transmission of Still Pictures in NTSC Composite Format Using Transform Domain Methods, IEEE Trans. on Communications (COM), 34, 3 (1986) 310-319.
- [26] Pratt, W. K., Spatial Transform Coding of Color Images, IEEE Trans. on Communication Technology, 19, 6 (1971) 980-992.
- [27] Abbas, H. M. ve Fahmy, M. M., Neural Model for Karhunen -Loeve Transform With Application to Adaptive Image Compression, IEE Proceedings-I, 140, 2 (1993) 135-143.
- [28] Brigham, E. O. ve Morrow, R. E., The Fast Fourier Transform, IEE Spectrum, 1, 12 (1967) 63-70.
- [29] Feig, E. ve Winograd, S., Fast Algorithms For the Discrete Cosine Transform, IEEE Trans. on Signal Processing, 40, 9 (1992) 2174-2193.
- [30] Heideman, T. M., Computation of an Odd-Length DCT from a Real-Valued DFT of the Same Length, IEEE Trans. on Signal Processing, 40, 1 (1992) 54-61.
- [31] Parabhu, K. A. ve Netravali, A. N., Motion Compensated Component Color Coding, IEEE Trans. on Communications (COM), 30, 12 (1982) 2519-2525.
- [32] Sabri, S., Movement Compensated Interframe Prediction For NTSC Color TV Signals, IEEE Trans. on Communications (COM), 32, 8 (1984) 954-968.
- [33] Srinivasan, R. ve Rao, K. R., Predictive Coding Based on Efficient Motion Estimation, IEEE Trans. on Communications (COM), 33, 8 (1985) 888-895.
- [34] Matsumoto, S., Coding Performance of Motion - Compensated Interframe, Interfield, and Intrafield Adaptive Prediction Coding for Composite and Component TV Signals, SMPTE Journal, 95, 3 (1986) 542-549.
- [35] Tu, G., A New Coding Method for Low Bit - Rate Video Signals, Visual Com. and Image Processing IV (SPIE), 1199, 1 (1989) 514-521.
- [36] Bar-Ness, Y. ve Politi, S., Lossless Image Compression Techniques using High Order Arithmetic Encoding, Communication, Control, and Signal Processing, 2, 6 (1990) 1458-1466.
- [37] Lu, W. ve Gough, M. P., A Fast Adaptive Huffmann Coding Algorithm, IEEE Trans. on Communications (COM), 41, 4 (1993) 535-538.

- [38] Netravali, A. N. ve Robbins, J. D., Motion - Compensated Television Coding: Part I, Bell Syst. Tech. J., 58, 1 (1979) 631- 670.
- [39] Tekalp, A. M. ve Özkan, M. K., Görüntü Dizisi İşleme, Bursa III Bilgisayar ve Haberleşme Sempozyumu, Eylül 1990, Bursa, Bildiriler Kitabı, Cilt I, 33-43.
- [40] Vega - Riveros, J. F. ve Jabbour, K., Review of Motion Analysis Techniques, IEE Proceedings-I, 136, 6 (1989) 397 - 403.
- [41] Bergholm, F. ve Carlsson, S., A 'Theory' of Optical Flow, Graphics, Models and Image Processing (CVGIP), 53, 2 (1991) 171-188.
- [42] Horn, K. P. ve Schunck, B. G., Determining Optical Flow, Artificial Intelligence, 17, 1 (1981) 185-203.
- [43] Treitak, O. ve Pastor, L., Velocity Estimation From Image Sequences With Second Order Differential Operators, 7th Int. Conf.on Pattern Recognition,, September 1984, Montreal, Proc. Book, Cilt I, 16-19.
- [44] Nagel, H. H. ve Enkelmann, W., An Investigation of Smoothness Constraints for the Estimation of Displacement Vector Fields from Image Sequences, IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI) , 8, 5 (1986) 565-593.
- [45] Fogel, S. V., Estimation of Velocity Vector Field from ime-varying Image Sequences, Comput. Vision Graphics Image Process.: Image Understanding, 53, 3 (1991) 1-14.
- [46] Limb, J. O. ve Murphy, J. A., Estimating the Velocity of Moving Images in Television Signals, Computer Graphics and Image Processing, 4, 2 (1975) 11-27.
- [47] Cafforio, C. ve Rocca, F., Methods for Measuring Small Displacements of Television Images, IEEE Trans. on Information Theory (IT) , 22, 5 (1976) 573-579.
- [48] Lee, C. ve Nadler, M., Interframe Image Coding With 3- Dimensional Edge Detection, ICASSP' 89, December 1989, London, Proc. Book, Cilt V, 541-545.
- [49] Choquet, B ve Pele, D., Segmentation of Apparent 2D Motion Fields, ICASSP' 89, December 1989, London, Proc. Book, Cilt V, 280-284.
- [50] Biemond, J., A Pel-Recursive Wiener-Based Displacement Estimation Algorithm for Interframe Image Coding Applications, Visual Communications and Image Processing II (SPIE), 845, 1 (1987) 424 - 431.

- [51] Efstratiadis, S. N. ve Katsaggelos, A. K., A Model - Based Pel-Recursive Motion Estimation Algorithm, ICASSP' 90, December 1990, London, Proc. Book, Cilt M4.5, 1973-1976.
- [52] Huang, T. S. ve Tsai, R. Y., Image Sequence Analysis: Motion Estimation, Ed.Huang, T. S., First Edition, Springer- Verlag , New York, 1981.
- [53] Kim, R. C. ve Lee, S. U., A VLSI Architecture for a Pel-Recursive Motion Estimation Algorithm, IEEE Trans. on Circuits and Systems (CAS), 36, 10 (1989) 1291-1300.
- [54] Walker, D. R. ve Rao, K. R., Improved Pel - Recursive Motion Compensation, IEEE Trans. on Communications (COM), 32, 10 (1984) 1128-1134.
- [55] Fortier, M., Architectures for VLSI Implementation of Movement-Compensated Video Processors, IEEE Trans. on Circuits and Systems (CAS), 33, 2 (1986) 250-259.
- [56] Thomas, G. A., HDTV Bandwidth Reduction by Adaptive Subsampling and Motion-Compensation DATV Techniques, SMPTE Journal, 96, 3 (1987) 460-465.
- [57] Thomas, G., Motion and Motion Estimation, Digital Image Processing, Ed. Pearson, D., First Edition, McGraw-Hill Book Company, London, 1991.
- [58] Mahmoud, S. A., Motion Detection and Estimation of Multiple Moving Objects in an Image Sequence Using the Cosine Area Transform, IEE Proceedings-I, 138, 5 (1991) 351-356.
- [59] Stuller, J. A. ve Netravali, A. N., Transform Domain Motion Estimation, The Bell System Technical Journal, 58, 7 (1979) 1673-1717.
- [60] Mahmoud, S. A., The Use of Cosine Area Transform (CAT) in Motion Estimation, Communication control and Signal Processing, 2, 6 (1990) 1506-1512.
- [61] Girod, B., Motion - Compensating Prediction With Fractional - Pel Accuracy, IEEE Trans. on Communications, 41, 4 (1993) 04-12.
- [62] Wu, S. F. ve Fernando, G. M. X., Comparative Study of Motion Estimation Techniques, ICASSP' 89, December 1989, London, Proc. Book, Cilt V, 305-309.
- [63] Jain, J. R. ve Jain, A. K., Displacement Measurement and Its Application in Inerframe Image Coding, IEEE Transactions on Communications (COM), 29, 12 (1981) 1799-1808.

- [64] Kappagantula, S. ve Rao, K. R., Motion Compensated Interframe Image Prediction, IEEE Trans. on Communications (COM), 33, 9 (1985) 1011-1015.
- [65] Komarek, T. ve Pirsch, P., Array Architectures for Block Matching Algorithms, IEEE Trans. on Circuits and Systems (CAS), 36, 10 (1989) 1301-1308.
- [66] Reuter, T., A Modified Blockmatching Algorithm With Vector Reliability Checking and Adaptive Smoothing, (ICASSP' 89), December 1989, London, Proc. Book, Cilt V, 295- 299.
- [67] Vos, L. D. ve Stegherr, M., Parameterizable VLSI Architectures for the Full-Search Block-Matching Algorithm, IEEE Trans. on Circuits and Systems (CAS), 36, 10 (1989) 1309-1316.
- [68] Yang, K. M., A Family of VLSI Designs for the Motion Compensation Block-Matching Algorithm, IEEE Trans. on Circuits and Systems (CAS), 36, 10 (1989) 1317- 1325.
- [69] Gerken, P. ve Adolph, D., Improved Motion - Compensating Prediction With Rotational Block Matching, ICASSP' 89, December 1989, London, Proc. Book, Cilt V, 290-294.
- [70] Orchard, M. T., Predictive Motion Field Segmentation for Image Sequence Coding, ICASSP' 90, December 1990, Proc. Book, Cilt M4.6, 1977-1980.
- [71] Orchard, M., A Comparison of Techniques for estimating Block Motion in Image Sequence Coding, Visual Communications and Image Processing IV (SPIE), 1199, 1 (1989) 248-258.
- [72] Srinivasan, R. ve Rao, K. R., Motion Compensated Coder for videoconferencing IEEE Trans. on Communications (COM), 35, 3 (1987) 297-304.
- [73] Koga, T. ve Inuma, K., Motion Compensated Interframe Coding For Video Conferencing, Nat. Telecom. Conf., November 1981, New Orleans, Proc. Book, Cilt I, G5.3.1 - G5.3.5.
- [74] Ghanbari, M., The Cross-Search Algorithm for Motion Estimation, IEEE Trans. on Communications (COM), 38, 7 (1990) 950-953.
- [75] Puri, A. ve Hang, H. M., An Efficient Block Matching Algorithm For Motion Compensated Coding, IEEE ICASSP' 87, December 1987, London, Proc. Book, Cilt I, 25.4.1 - 25.4.4.

- [76] Gangal, A. ve Kayıkçıođlu, T., A New Bit Rate Reduction Technique in Video Coding Based on Motion Compensation Using Adaptive Block Matching, MELECON' 94, Mediterranean Electrotechnical Conference, April 1994, Antalya, Bildiriler Kitabı, Cilt II, 1113-1116.



8. ÖZGEÇMİŞ

11.12.1963 yılında Trabzon'un Akçaabat ilçesinde doğan Ali GANGAL, 1986 yılında Karadeniz Teknik Üniversitesi Mühendislik Mimarlık Fakültesi Elektrik-Elektronik Mühendisliği Bölümü'nden mezun oldu ve aynı yıl yüksek lisans öğrenimine başladı. 1988 yılında aynı üniversitenin Mühendislik-Mimarlık Fakültesi Elektrik-Elektronik Mühendisliği Bölümünde araştırma görevlisi olarak çalışmaya başladı. 1989 yılında "Sayısal Rekursif Süzgeçleme İle Video İşaretinde Gürültü Azaltımı" konulu yüksek lisans tezini tamamlayarak Elektronik Yüksek Mühendisi ünvanıyla mezun oldu. 1989 yılında Doktora Programına kayıt oldu. Sayısal sinyal işleme ve video kodlama konularında araştırma yapmaktadır. Evli ve bir çocuk babasıdır. İngilizce bilmektedir.